Master's thesis

NTNU
Norwegian University of Science and Technology
Faculty of Information Technology and Electrical Engineering
Dept. of Information Security and Communication Technology

Stian Madsen Storebø

ChatRange

Designing Cyber Security Exercise Scenarios using Autonomous AI Agents and Artificial Intelligence

Master's thesis in Information Security
Supervisor: Muhammad Mudassar Yamin
Co-supervisor: Basel Katt
June 2024

NTNU
Norwegian University of
Science and Technology

Stian Madsen Storebø

ChatRange

Designing Cyber Security Exercise Scenarios using Autonomous AI Agents and Artificial Intelligence

Master's thesis in Information Security
Supervisor: Muhammad Mudassar Yamin
Co-supervisor: Basel Katt
June 2024

Norwegian University of Science and Technology
Faculty of Information Technology and Electrical Engineering
Dept. of Information Security and Communication Technology

NTNU
Norwegian University of
Science and Technology

Abstract

Planning cyber training exercises is a time-consuming process. For an eight-hour tabletop exercise, NIST and MITRE estimate a total of one to three months of planning time has to be estimated. This creates a gap where companies cannot train for critical events due to time or the cost of training. To tackle this issue we developed a system named ChatRange, using Autonomous AI agents together with Large language models to automate the planning process of tabletop exercises. The new technology is able to research realistic scenarios using tools to provide external real-world information into existing large language models. The system is able to be implemented easily by only providing API keys to the system, and uses the cloud delivery model for simple implementation.

When measured against a manually developed training scenario in a blind study, ChatRange scored only 0.16 percent lower than the average score of six quality criteria. ChatRange also scored higher or equal in four of the criteria, only scoring lower in the two related to the generalization of the scenario. The system was also compared with the cost of manually planning a scenario. ChatRange reduced the cost of creating cyber training exercises by 99.81 percent, from an estimated 80,000 NOK per exercise to only 150 NOK. The specific data cost for using ChatRange was 13.10 NOK of the 150 NOK. Further automation and optimization have the potential to enhance these results.

ChatRange delivers both quality and efficiency and proves that Autonomous AI Agents and large language models are good tools for these purposes.

Sammendrag

Planlegging og gjennomføring av cyberøvelser er en tidkrevende prosess. Blant annet estimerer NIST og MITRE å bruke minst en til tre måneder på å planlegge en åtte timers øvelse. Dette skaper et stort problem, der bedrifter ikke er i stand til å håndtere kritiske sikkerhetshendelser, enten på grunn av mangel på ressurser eller på grunn av de høye kostnadene.

For å løse dette problemet har vi utviklet ChatRange. Løsningen bruker Autonome AI-agenter sammen med språkmodeller for å automatisere planleggingsfasen av skrivebordsøvelser. Denne teknologien kan utvikle realistiske scenarioer, blant annet ved å hente informasjon om trusselaktører og sikkerhetshendelser direkte fra internett. Dette kobles så sammen med de eksisterende språkmodellene. Systemet er laget for å enkelt kunne tas i bruk av organisasjonene som ønsker dette da løsningen benytter seg av språkmodeller som leveres som skytjenester.

I en blind studie der en øvelse utviklet av ChatRange ble målt mot en øvelse laget av mennesker scoret ChatRange 0,16 prosent lavere enn kontrollen, målt i seks kvalitetskriterier. ChatRange scoret bedre eller likt i fire av kriteriene. ChatRange ble også målt mot kostnaden for å utvikle et scenario mot den manuelle metoden. Her reduserte ChatRange kostnaden for å utvikle øvelser med 99,81 prosent, fra et estimat på 80 000 kroner per øvelse til kun 150 kroner. Av disse kostnadene er de direkte datakostnadene for å bruke løsningen på 13,10 kroner. Videre automatisering kan redusere dette tallet ytterligere.

ChatRange leverer både i kvalitet og effektivitet og beviser at autonome AI-agenter og store språkmodeller er gode verktøy for disse formålene.

Acknowledgments

The answer to the Ultimate Question of Life, The Universe, and Everything is 42. This is a universally recognized fact in the scientific community, including those working with Artificial Intelligence. The supercomputer Deep Thought [1] demonstrated this, dedicating over 7.5 million years to calculating this answer. While our research project may not have lasted that long, the results we have achieved we feel are of equal importance, resonating as profoundly as the number 42.

The past year has been a journey filled with both challenges and triumphs. Working with large language models is like working with a 'programming language' that works 60 percent of the time and evolves daily. It has been a roller-coaster of frustration, excitement, and learning.

I thank my supervisors, Basel Katt and Muhammad Mudassar Yamin, for their hard work. You have been a tremendous resource for all the academic help you have provided, especially for suggesting relevant academic research and support with this research project.

I also would like to thank my employer, Norwegian Customs, and my manager Rune Søgnen, for enabling me to do this research project in addition to my everyday day-to-day work. Without the adaptability shown to me, this would not be possible.

Finally, I must express my deepest thanks to my husband, Olav Taklo. The last two years of studies have demanded much of my time, which we otherwise should have spent together. I am truly grateful for your understanding and support.

I hope you will enjoy reading this research report and be able to learn something from it, and enjoy it at least as much as I've enjoyed researching it.

So long, and thanks for all the fish [1].

Contents

Abstract . . . iii
Sammendrag . . . v
Acknowledgments . . . vii
Contents . . . ix
Figures . . . xv
Tables . . . xvii
1 Introduction . . . **1**
1.1 The Secret War . . . 1
1.1.1 On the rise . . . 1
1.1.2 Defending together . . . 2
1.1.3 Training challenges . . . 4
1.2 Motivation . . . 4
1.3 Problem description . . . 5
1.4 Research questions . . . 5
1.4.1 Goals . . . 5
1.4.2 Research questions . . . 5
1.5 Scope . . . 6
1.5.1 Norwegian cyber range . . . 6
1.5.2 NATO . . . 6
1.5.3 Cyber training participants . . . 6
1.5.4 NTNU . . . 6
1.6 Document Structure . . . 6
2 Background . . . **9**
2.1 Responding to the threats . . . 9
2.1.1 Committed to training . . . 10
2.1.2 Exercise techniques . . . 11
2.1.3 Exercise formats . . . 12
2.2 AI am legend . . . 13
2.2.1 Logical layers . . . 13
2.2.2 The Beginning . . . 15
2.2.3 Learning to work . . . 15
2.2.4 Natural Language processing . . . 16
2.2.5 Large language models . . . 18

2.3 Related research 21
2.3.1 Generative Agents: Interactive Simulacra of Human Behavior 21
2.3.2 Communicative Agents for Software Development 22
2.3.3 ReAct: Synergizing Reasoning and Acting in Language Models 22
2.3.4 Applications of LLMs for Generating Cyber Security Exercise Scenarios 22
2.3.5 AiCEF: an AI-assisted cyber exercise content generation framework using named entity recognition 23
3 Research Methodology 25
3.1 Design Science Research 25
3.2 Methods 27
3.2.1 Awareness of the problem 27
3.2.2 Suggestion phase 28
3.2.3 Development 28
3.2.4 Evaluation 29
3.2.5 Conclusion 30
3.3 Artifacts 30
3.4 Conclusion 30
4 Large Language models 33
4.1 Introduction 33
4.2 Delivery methods 33
4.2.1 Types 34
4.2.2 Use cases 35
4.2.3 Proposed solution 37
4.3 Models 37
4.3.1 Cloud models 38
4.3.2 Performance 40
4.3.3 Available models 42
4.4 Conclusion 42
5 Autonomous AI Agents 45
5.1 Context over time 45
5.1.1 AGI 46
5.1.2 Thinking agents 47
5.2 Agent frameworks 50
5.2.1 ChatDev 50
5.2.2 AutoGen 51
5.2.3 CrewAI 53
5.3 Overview 54
5.4 Conclusion 54
5.4.1 Summary 54
5.4.2 Optimal solution 55
6 ChatRange 57
6.1 Integrated design 57
6.1.1 Content Design Context 58

6.1.2 Feedback loop with thought instructions 59
6.2 Data generation 60
6.2.1 Few-shot prompting with dynamic content 60
6.2.2 Generative Agents 61
6.3 Exercise development 65
6.3.1 Exercise context 65
6.3.2 Objectives 67
6.3.3 Threat Intelligence 67
6.3.4 Time management 68
6.3.5 Storyline 69
6.3.6 Questions 69
6.3.7 Case Studies 69
6.4 Conclusion 69
6.4.1 Context 70
6.4.2 Exercise development 70
7 Validation 71
7.1 Testing Methodology 71
7.1.1 CISA Tabletop Exercise Packages 71
7.1.2 Case Studies 73
7.2 Case Study Results 74
7.2.1 Broad study 74
7.2.2 Expert Evaluations 75
7.2.3 Discussions 75
7.3 Efficiency Measurements 81
7.3.1 Time measurements 81
7.3.2 Cost per work hour 82
7.3.3 Additional costs 83
7.3.4 Comparison 83
7.4 Scalability and Flexibility 83
7.4.1 LLM’s Delivery Model 84
7.4.2 Flexibility 84
7.4.3 Usability 84
7.5 Conclusion 85
7.5.1 Quality 85
7.5.2 Efficiency 85
7.5.3 Summary 86
8 Conclusion 87
8.1 Summary 87
8.1.1 Large Language models 87
8.1.2 Autonomous AI Agents 88
8.1.3 ChatRange 89
8.1.4 Evaluation 90
8.2 Overall conclusion 91
8.3 Research Limitations 91

8.3.1 Constantly updating technology 91
8.3.2 Case study 91
8.4 Further studies 92
8.4.1 Hybrid and Full live exercises 93
8.4.2 Simulated training 93
Bibliography 95
A ChatRange 105
A.1 Description 105
A.2 Overview 105
A.3 Agents 106
A.3.1 Threat hunter 106
A.3.2 Cybersecurity Researcher 107
A.3.3 Chief Information Security Officer (CISO) 108
A.4 Teams 109
A.4.1 Threat Intelligence 109
A.4.2 Case Studies 109
A.4.3 Case Research 109
A.5 Tools 110
A.5.1 Search Tool 110
A.5.2 Exa Search 110
A.5.3 Suggester 110
A.5.4 Wikipedia 111
A.6 Tasks 112
A.6.1 Threat Intelligence 112
A.6.2 Case Studies 113
A.6.3 Case Research 114
A.7 Static Prompts 116
A.7.1 Exercise Objectives 116
A.7.2 Determine Parts 116
A.7.3 Create Timeline 116
A.8 Storage 118
B Databank 119
B.1 Scoring table 119
B.1.1 Details: 119
B.1.2 Technical Soundness 119
B.1.3 Realism 120
B.1.4 Creativity 120
B.1.5 Usability in exercises 120
B.1.6 Expandability with human inputs 121
B.2 Raw Data 122
B.2.1 Playbook 1 - Human Generated (P1) 123
B.2.2 Playbook 2 - Chatrange Generated (P2) 125
C Exercise Playbooks 127
C.1 Description 127

C.2 CISA Generated Playbook 127
C.3 ChatRange Generated Playbook 141
D Expert Reviewer Profiles 157
D.1 Description 157
D.2 Reviewer 1 158
D.2.1 Current Role 158
D.2.2 Background 158
D.2.3 Review 158
D.3 Reviewer 2 159
D.3.1 Current Role 159
D.3.2 Background 159
D.3.3 Review 159
D.4 Reviewer 3 160
D.4.1 Current Role 160
D.4.2 Background 160
D.4.3 Review 160

Figures

2.1 Show a example of how Improve is wrongly stemmed to a meaningless word, where Lemmatization is able to get it right. 17

2.2 Show a example of Zero Shot versus Few shot when the model is asked to design a new car. 21

3.1 The process of Design Science Research with the phases and outputs [53] 26

4.1 An overview of delivery methods for large language models and each method's respective area of responsibility. 34

4.2 When asked to create a recipe for methamphetamine, the censored model, like llama-2, refuses to provide this, while an uncensored model, like Mistral Instruct, provides the data without question. Mistral Instruct is run on owned hardware. 36

5.1 The architecture of BabyAGI [81]. It consists of a context layer with agents tasked for prioritization, execution and creation of tasks. . . 46

5.2 GPT 3.5-Turbo (ChatGPT) answers a question from the user where the model is asked to name the five founders of Microsoft. It gets this answer wrong, partially because the question is also wrong, as Microsoft only has two founders. The model is, however, not able to point this out. [84] 48

5.3 This is an example of Reasoning and Acting. The example shows a query about when Google was founded, and both methods got the answer wrong when used independently. 49

5.4 When applying ReAct[47] to the model, the system can reason with the answers given and dynamically adjust the input for the model based on memory. 50

6.1 The integrated design shows how information is used in the development process of the training scenarios. The information is run through a dual feedback loop and then used for context in the next part of the scenario development process. 58

6.2 The content generation shows how the system design can either use static prompting or generative agents depending on the complexity and type of task of the task. 60
6.3 Shows an example of a single agent in the system. The Threat Hunter, as defined in ChatRange, has its goal, background, and task defined. 62
6.4 Shows a overview of the different processing mechanisms in CrewAI [49]. The Sequential processing mechanism is a step by step task runner. The Hierarchical provides more flexibility as it is up to the Manager to selected tasks for the right agents. 64
6.5 This figure provides a complete overview of ChatRange's functionality. It also includes the type of data used for each specific process in developing cyber training scenarios for the solution. 66
6.6 This is an example of how the threat intelligence team of agents can research an incident type for the training exercise. Here, the scenario focuses on a third-party infection that leads to a ransomware attack on the target company. 68
6.7 The storyline provides an overview of the events leading up to the training scenario in the exercise, which helps increase the exercise's realism. 69

7.1 Broader study - The first playbook (P1), which was human-generated, scored High to Moderate in all criteria, the same as the second playbook (P2) from ChatRange, with only minor changes in each criterion. 75
7.2 Expert reviews - The first playbook (P1), which was human-generated, scored High to Very High in all criteria, the same as the second playbook (P2) from ChatRange, with only minor changes in each criterion. 76
7.3 The second playbook (P2) scored overall better in two of the criteria, and equal for two other criteria, in the survey, while P1 scored better in the last two. 77
7.4 ChatRange scored equal or better in four out of the six categories evaluated. The graphs show the percentage difference in scores across the criteria. The red score is where ChatRange scored worse than the manual playbook, and green is where it scored better. . . . 78
7.5 The expert's reviewers' results of ChatRange compared to the average scores of scenario 1 and scenario 2 in their research. [50]. . . . 80

Tables

2.1 An overview of the different types of exercises combined with the techniques common for each [25]. 12

4.1 Shows an overview of relevant performance tests performed for each model we have considered for use in the proposed solution. . 41

5.1 An overview of the features of each framework is discussed in this chapter, along with the methodology, specialty, focus area, and drawbacks for each. 54

7.1 This table provides an overview of the employment areas of the survey participants. It shows a generalized category and an overview of each participant's more specific titles. 74

7.2 The table displays the average score of each playbook for each criterion from the first broader case study, including the scores from the expert reviewers. 76

7.3 The total cost of producing an exercise using a manual method versus ChatRange. ChatRange sees a 99.813 percent reduction in calculated costs. The currency values are shown in NOK. 82

7.4 The total cost of using ChatRange is based on OpenAI's cost structure, shown in Norwegian Kroner. The currency values have been converted from the original USD to NOK, with an exchange rate of 1 USD to 10.81 NOK (06/05/2024). 83

7.5 Overview of installation times for ChatRange on a clean Ubuntu installation. 84

Chapter 1

Introduction

1.1 The Secret War

Since the start of the Internet in the early 90s, a secret war has occurred. In this war, nations, criminals, and other threat actors are actively but silently fighting against each other with the goal of financial or territorial gains.

1.1.1 On the rise

With the rise of Artificial Intelligence and worldwide dependence on information technology, the impact of these attacks humanity faces is suddenly becoming real for nations, organizations, and ordinary people.

Simple attacks

The last five years have seen an incredible rise in cyber security attacks worldwide. Threat actors' access to new technology has enabled a criminal market, and these years have also seen some of the most costly attacks. In 2017, the total cost of monetary damage across the United States was reported to be $1,418 million. In 2023, however, this number reached a staggering $12,500 million, a 21 percent increase yearly, according to the United States Internet Crime Complaint Center (IC3) [2].

It is easy to think that the most complicated attackers, like nation-states or advanced persistent threats, are the most critical focus areas. However, these threat actors are not the ones organizations and governments face daily. Ransomware attacks were the most significant category of all cyberattacks in 2022, with 68.42 percent of reported cyber attacks worldwide [3]. Ransomware attacks are generally seen as being used by less sophisticated threat actors, like criminal organizations wanting economic gain.

1.1.2 Defending together

NATO (The North Atlantic Treaty Organization) is a military alliance founded in 1949 with 31 member states, including the United States [4]. The treaty focuses on military cooperation and is one of the leading researchers into warfare tactics and strategies. NATO is often connected to military power. However, the organization is also focused heavily on Cyber warfare [5]. This domain connects military forces worldwide, working together to defend the member states against cyber criminals.

National Cyberwars

In today's networked environments, with the Internet as the most used communication carrier of information, Cyber warfare is one of the most significant categories where the tactics and threat actors change the most often. This makes this the hardest to define. In 1994, the common consensus was that the current Internet infrastructure was not at a level where combat and warfare could be done over this medium [6, p. 75]. Today, however, this has all changed.

On the 24th of February, 2022, Russia attacked Ukraine using a massive physical invasion force. Tanks, artillery, and soldiers invaded the country, intending to take over control of Ukraine. Another different war took place in the leading months up to the attack. In the shadows, Russia had started a digital warfare campaign against the country, identifying weak points in the infrastructure and planning attacks. They launched their last string of cyber operations only hours before the invasion. [7]

In the early morning of February 24th, Viasat satellite subscribers all over Europe noticed problems with their connections [8]. Their modems would not connect to the satellites. Russia had attacked the network, rendering thousands of satellite modems inoperable. Fourteen days later, several thousand subscribers were still without a working connection [7]. The attack, however, affected Ukraine and several other European countries, including Norway.

These kinds of attacks are why NATO has given special attention to Cyberwarfare. In 2008, they established the NATO Cooperative Cyber Defence Center of Excellence. The organization works for better cooperation and serves as a cyber defense hub between NATO member states, sharing experience and knowledge [9].

To better understand how cyber warfare affects us, NATO has divided the areas into several subcategories based on the types of attacks and their effects on the victims. Organizations and companies worldwide often face Cyber Espionage, sabotage, and attacks on critical infrastructure.

Cyber Espionage

The adversary's goal is to compromise and get access to the defender's systems or infrastructure to get information. The attacks will most likely be stealthy and undetectable when ongoing. In 2017, a large-scale operation was uncovered where several Managed Service Providers (MSPs) worldwide had been compromised. [10] Operation Cloud Hopper was the name given to the compromise, and behind it was the Chinese Advanced Persistent Threat APT10, also called Stone Panda or POTASSIUM [11]. The attack targeted diplomatic and political organizations to siphon off information passed through the MSPs.

Sabotage

The goal is to disrupt or destroy services and are deliberate actions. In 2005, Iran's Nuclear weapons program was up and running, and the United States feared that the country would be able to develop nuclear weapon capabilities very shortly. In 2010, an attack the world had never imagined, with a scope never seen before. Over several years, the Iranian regime has tried to acquire and build its nuclear weapons. This has, of course, been a primary concern for Western countries like the United States. In response to these actions, some countries saw a need for intervention. Stuxnet was a 500 kb piece of code designed to infiltrate and destroy centrifuges used in Iranian nuclear material processing. The code was created to be infected via a third-party update using Siemens controllers and spread via USB transfer when plugged into the Airgapped systems of the processing facilities. The sophisticated malware first recorded the safe configuration of the centrifuges while running. Then, the recording was replayed. Meanwhile, the configuration was set to an unsafe state. This made the centrifuges fail faster than intended. Although official attribution has never been made, it is assumed that the United States and Israel are the main threat actors involved in this attack. [12]

Critical infrastructure

All computer systems depend on electricity, so an attack on this type of infrastructure can have severe consequences. These systems are called critical infrastructure and should never be connected to the Internet if configured correctly. Unfortunately, this is not the case in many situations. In the days leading up to the invasion of Ukraine in 2022, several attacks were performed against the Ukrainian critical infrastructure [13].

On February 23rd, a malware attack was conducted against several government websites, financial institutions, and the aviation sector. The malware turned out to be HermeticWiper, a wiperware made for destroying the systems it is deployed on [14].

Starting on April 12, 2022, the CERT-UA responded to an infection in an energy production plant. The malware called Indudestroyer2 was attributed to the Russian group Sandworm. The same group then attacked several critical infrastructure sectors, including a destructive attack on a logistics provider on April 19th and the transportation sector on April 29th [13]. This was the first time this group was seen as active in the last five years.

1.1.3 Training challenges

The complex battlefield that organizations and governments face in the digital cyber war creates challenges for defending themselves. One critical element in this defense is to provide adequate cybersecurity training to employees and management. However, this is easier said than done.

In a study conducted by Egress, IT managers were asked what they believed would have the most negative impacts on their business. 43 percent answered that data exfiltration would be the highest, and only 21 percent answered that phishing had the most significant implications [15]. In 2023, however, phishing was reported to be the crime category that affected most individuals in the United States, with almost 300,000 victims impacted [16]. This creates a disconnect in how the IT managers prioritize where IT and Security departments should focus their defensive capabilities and where they should be. In 2022, it was reported that only 55.98 percent of healthcare cybersecurity professionals had received cybersecurity awareness training at least once in the last year[17].

A similar study by the Department for Science, Innovation, and Technology in 2023 showed that only 18 percent, or two in five businesses, conducted cybersecurity training at least once in 2022 [18].

This lack of training leads to significant challenges for organizations and, unfortunately, opportunities for threat actors. Customized training options must be available if organizations are to be prepared for cybersecurity incidents.

1.2 Motivation

Over the years, numerous examples of lousy incident handling have increased the severity by miles. Devastating ransomware attacks could have been avoided if employees had been trained to recognize and not click on malicious links. Untrained management exacerbates crises and turns minor incidents into major disruptions.

Adopting the military adage, "Train as you fight, Fight as you train," helps focus on what is essential when developing and planning training scenarios. If we, as security experts, plan exercises that do not feel real for the participants and are out of the scope of normal day-to-day operations, we fail instantly in our objective.

To increase the security posture worldwide, there must be alternatives for companies and organizations that cannot conduct training today.

1.3 Problem description

According to a recent survey, two out of five companies in the UK failed to provide annual employee training, likely due to cost and resource constraints. The survey also revealed a discrepancy between larger and smaller companies: while at least 70 percent of larger companies provided annual training, only 20 percent of smaller companies do the same [18].

Developing and executing cyber exercise training presents significant challenges for organizations and companies due to the required time and resource commitment. According to NIST guidelines [19], planning and developing a typical eight-hour tabletop exercise typically requires up to three months, also confirmed by MITRE Corp.

While some organizations may opt to purchase external training for public providers, the financial cost of performing this kind of training can restrict the participation of many smaller companies with limited budgets. Secondly, even for large organizations, training should be customized to provide better realism and effects from conducting the exercise. Having the ability to train in the typical day-to-day environment helps when a security incident takes place. Just as a fire drill is most effective in the actual building used daily, cyber exercises are most beneficial when conducted in the organization's regular operating environment.

1.4 Research questions

1.4.1 Goals

G1: The first goal is to reduce the time needed to plan and execute cyber training exercises, enabling training for more companies than currently available.

G2: The second goal is to develop a system that organizations can use to create customized training scenarios, using technology to support this objective.

1.4.2 Research questions

Based on these goals, the following research questions have been defined:

RQ1: What are suitable large language models for generative autonomous AI agent projects, their delivery methods, and their respective strengths and limitations?

RQ2: How can Autonomous AI agents increase the quality of the output delivery in contrast to traditional methods where only large language models are used, and what is the most optimal implementation for ChatRange?

RQ3: How can autonomous AI agents be used alongside large language models to develop cyber security training scenarios?

RQ4: How do agent-based cyber training scenarios' details, realism, creativity, and efficiency measure against traditional manual approaches?

1.5 Scope

The scope of the research is defined as the stakeholders. These are entities that have an interest or concern in the research performed. In this report, we have defined several stakeholder groups.

1.5.1 Norwegian cyber range

The Norwegian Cyber range cooperates with several companies and academic institutions in Norway and Estonia. They run the most extensive cyber range in Norway, where participants can come and train in several cyber exercises over the year. The cyber range is hosted at the Norwegian University of Science and Technology at Gjøvik [20]. The cyber range is currently managed by 1.5 full-time equivalent positions, with the addition of a few hired developers.

1.5.2 NATO

The North Atlantic Treaty Organization runs several extensive cyber exercises. Locked Shields is one of the biggest, with participants from most NATO countries. The Estonian National Cyber Range CR14 and many other companies host the exercise. The Exercise planning team included over 400 people organizing over 5,500 virtual systems and 2,000 participants [5].

1.5.3 Cyber training participants

Participants include those needing cyber training, focusing on the civilian population. This could be participants using managed Cyber ranges or entities wanting to set up and run their cyber range.

1.5.4 NTNU

Research into using new technology to automate content generation by utilizing Artificial intelligence and Large language models is a field of great scientific interest in a new area.

1.6 Document Structure

This section will briefly provide an overview of all the other chapters in this research report.

Chapter two provides vital background information about different cyber training types and how they are performed, including the various challenges each type has for organizations. This chapter also discusses Artificial Intelligence, its history, and how this technology is implemented in Natural Language processing. This information is essential to understanding the technology used in the proposed solution. The chapter finishes with an overview of the related research papers.

Chapter three gives an overview of the research methodology. It discusses the different research questions and how they are proposed and solved in depth, including the other phases.

Chapter four is a literature study on large language models, their delivery types, and performance measurements. It also gives an overview of the language models most suitable for this research project and how they can be implemented.

Chapter five is a literature study to find the best autonomous AI agent systems for the proposed solution. It also gives an overview of the autonomous AI agent architecture and the different new technologies these frameworks use for implementation.

Chapter six details the proposed solution ChatRange. This chapter provides technical detail on how the different LLMs are implemented and the inference types for each step of the cyber security exercise development process. Here, you will learn how autonomous AI agents are implemented using an agent structure with internal team communication and how this method helps produce realistic content.

Chapter Seven provides information on evaluating ChatRange's output and results. ChatRange is measured using six quality criteria in a blind study involving participants with experience from several areas within IT and IT security. In addition to this, a second case study using expert reviewers was also performed, scoring the usability of the exercises. The last metric used is a cost evaluation where ChatRange is measured against traditional methods of creating cyber training exercises.

Chapter eight contains the research report's overall conclusion. Here, all the research questions from the previous chapters are detailed, along with the overall summary of the findings. This chapter also provides information about research limitations and suggestions for further studies.

Chapter 2

Background

Having background information on the proposed solutions presented in this research report is essential to understanding the consequences and challenges of today's technology for cyber exercise training. This chapter aims to increase understanding of why cyber exercise training is necessary for today's threat landscape and how this activity can help improve the security posture of organizations and companies.

The second part of the background needed is understanding the underlying technology and the history of Artificial Intelligence and Large language models that are critical components used in the research report. Artificial intelligence is an area of expertise that has dramatically increased in popularity in recent years with the release of modern advanced Language models to the public. This increase has also sparked the interest of threat actors, making them more challenging to spot and giving them a more significant edge than ever before.

2.1 Responding to the threats

In November 1988, the first-ever Security Incident team was created at Carnegie Mellon University in Pittsburgh. The original goal was to share vulnerability information with software vendors and partners to enable faster patches for vulnerabilities, like the one used by Morris Worm the same year. [21]

This organization of cyber security professionals has helped organizations through security incidents for many years. One significant challenge remains providing organizations and companies with the means to protect themselves against cyber threats. Not all organizations have the means to establish dedicated security teams; even if they do, measures still need to be implemented to protect employees and critical data. One key element is providing relevant security training for crucial personnel and management.

2.1.1 Committed to training

With threats constantly evolving, relevant training for professionals and other users is vital to creating the best security posture. Much work has gone into creating security documentation, like using the ISO 27001 [22] framework to implement security governance. Still, many companies and organizations forget they must perform relevant cyber exercise training to verify that their plans work.

NIST, MITRE, and CISA are three large organizations with extensive experience in creating and developing processes and resources for planning and executing cyber training scenarios.

NIST

The National Institute of Standards and Technology [23] is dedicated to developing tests and recommending best practices for the US government and public sectors. The organization has published several Standards, including SP 800-64—Guide to Test, Training, and Exercise Programs for IT Plans and Capabilities [19]. The standard includes information on designing, planning, and executing tabletop and functional exercises.

MITRE

MITRE corporation is a not-for-profit organization established in the 1960s, with support from the U.S. Air Force [24]. The goal was to connect the academic and industry environments to strengthen the U.S. stance in the Cold War. The organization runs several research and development centers across the United States and is a leading organization in public cybersecurity defense and safety matters. The organization has published a guide for organizations on conducting, planning, and running cyber training scenarios, named the Cyber Exercise Playbook [25]. The guide extensively introduces several exercises, including Tabletop, Hybrid, and Full live exercises.

CISA

The Cybersecurity and Infrastructure Security Agency, established in 2018, is the national advisor on cybersecurity matters for the United States [26]. The agency is run under the Department of Homeland Security. One of the key focus areas for CISA is the defense of critical infrastructure in the U.S. The agency is highly focused on delivering training options for organizations. To achieve this, they have developed the CISA Tabletop Exercise Packages, ready-to-use templates for cybersecurity tabletop training [27].

2.1.2 Exercise techniques

Below are some standard training techniques used in running cyber training exercises.

Paper-based

Paper-based training is one of the simplest types of exercise forms that does not require any technical knowledge. This format is appended to every piece of information used in the exercises, written on paper, and given to the participants [25]. This information can be news articles, discussion questions, case studies, etc.

A significant advantage of this method is that anyone can plan and execute these exercises with minimal knowledge of running cyber training. The exercise types work very well in discussion-based exercises where critical thinking and evaluation are the main training objectives.

Capture the Flag (CTF)

Capture the Flag is a competition form commonly used in cybersecurity professionals' training. The task is simple: You must find a flag or an answer for a given task. The challenges can be presented to teams or individuals, and the platform can run unsupervised. This means you do not need individuals to lead the participants through the training after the platform runs. [28]

Two main CTF types are used:

- **Jeopardy Style**: This format is quite simple. You are given a question to answer and will be awarded points if your answer is correct. The goal is to either complete all tasks or get the highest score. BlueTeamLabs [29] is a commercial provider of such types of CTF.
- **Attack / Defense**: In this type of CTF, two teams typically have their server or infrastructure. The goal is for each team to attack the other and get a flag, often located on the defender's system. Presenting the flag proves that the attack is successful. The flag can either be a file or parts of a unique file. Based on the training scenario, this type of competition can be attack and defense or, e.g., only attack or defense.

Cyber ranges

Cyber ranges take the technical aspects from Capture the Flag contests, combine them with the scenario method used in Tabletop exercises, and combine them into a simulated environment. The idea is that participants can get hands-on training in real situations without handling actual incidents. The cyber range is repeatable so the training can be performed multiple times for a single scenario. The concept for

Exercise type	Tabletop	Hybrid	Full Live
Planning Time	1-3 Months	3-6 Months	6-18 Months
Length	1-3 Days	3-5 Days	1-14 days
Techniques	Discussion	Discussion Cyber Range CTF	Multiple Cyber Ranges

Table 2.1: An overview of the different types of exercises combined with the techniques common for each [25].

this kind of training comes from the military, where training in real-life scenarios is essential to prepare for real situations. [28] The Norwegian Cyber range at NTNU [20] is one of the largest cyber ranges in Norway where participants can come and train.

2.1.3 Exercise formats

Cybersecurity exercises can be delivered in several formats, each with various complexities and containing several types of training.

Tabletop exercises

Tabletop exercises have a significant, unique feature. They replace the technical aspects required to perform and participate in regular cyber training with thought. [19] This has several advantages. The setup for small tabletop exercises will often be less than setting up a complete simulation training environment [25]. The exercise can also be adapted specifically for the specific people to be trained. Suppose we want to train the management of an organization in handling a ransomware attack. In this scenario, the exercise team can create the needed training material and run the exercise with them over a few hours, with zero preparation for the participants [28].

Tabletop exercises are typically conducted over one to three days, and the required planning time is estimated to be from one to three months [25].

Hybrid Exercises

Combining live events and discussion-based exercises can help introduce new training moments and possibilities. Hybrid exercises are paper-driven exercises where systems, computers, or people facilitate some elements to heighten the training effect. These events are called injects and are often defined using a Master Scenario Event list [19, 25]. This is a list of insects combined with a storyline and timestamp for use in an exercise.

Hybrid exercises are often more complex to plan and execute, and it is customary to estimate three to six months of planning before they are executed [25].

Full Live Exercises

The third format of exercise is full-live exercises. These are significant training events that often involve multiple organizations and countries. They often involve numerous cyber ranges; some are interconnected, serving several thousand participants [25]. One of these exercises is Locked Shields. This full-live exercise is hosted by NATO and contains over 2000 participants from over 32 countries. The exercise runs over two full days, and both virtual systems and physical hardware are used [30].

Full-live exercises have an extremely complex setup, and the estimated planning and design time for such an exercise is estimated to be six to twelve months [25].

2.2 AI am legend

The introduction of Machine Learning and Large Language Models, often called Artificial Intelligence, has created a whole new area of Cyber warfare that needs to be covered in both civilian and defense areas. This challenge is also one that NATO significantly focuses on by developing an Artificial Intelligence Strategy [31]. In this strategy, each member state of NATO and its allies has agreed to six principles for responsible governance of AI [31]. To understand why there is a need to have these strategies in place, there is a need to look deeper into the technology powering this new area.

Artificial Intelligence, as used today, is not true Intelligence. What is defined today as Intelligence is just several models combined into AI, and this technology field has changed rapidly over the last decade.

2.2.1 Logical layers

Several logical layers describe how Machine learning and Artificial intelligence are built up. These layers combine to determine where the human race is now and where we want to go in this research paradigm. There are a total of four layers [32].

Reactive machines

The simplest form of Artificial Intelligence is Reactive machines. This layer has the simplest computer models. These models are developed for a specific task and are often trained to identify patterns or a given input variable. Based on the input, the model gives back an output without any notion of past events [32].

Deep Blue is one of the most famous models in this layer. Deep Blue was a chess computer model developed in 1996, and it was revolutionary in its time. In previous models, when the opponent made a move, the computer had to calculate each possible move that could be done. With Deep Blue, they pre-trained the model by having it play games by itself, learning to improve with each game. The model did, however, have some faults, and in a match against World Chess Champion Gary Kasparov, it lost 4-2. This loss fueled the further development of the model, and in 1997, Deep Blue 2 saw the light of day. This model would become the first computer to beat a World Chess Champion by winning 3½ against Gary Kasparov's 2½ score in the rematch 1997 [33].

Even in more modern-day situations, these models are still in use. There are several applications where more advanced models are used and not needed. One example of this is Google's Alpha Go. This model was initially created by DeepMind in 2016 and trained with deep learning to play the game Go [34]. This is a game notably challenging to predict. For example, in a chess game, the player can have 20 moves; in Go, this number is 386 [35]. In 2016, Alpha Go beat the current highest-ranking Go player, Lee Sedol [34].

Limited Memory

The next level of Artificial intelligence adds something missing in the simple models, namely learning from mistakes and forming memories [32]. This simple principle suddenly takes our basic models to a whole new level. Some models on the market today have started to dip their toes into this level, but the research is only in its beginning state. The theory is that the models start with an empty or primary knowledge pool and try different combinations before learning by failing. After each run, the model stores the knowledge it has gained and uses it in the next run [32].

Humans use this principle in their daily lives to gain knowledge. For example, when starting kindergarten, our experience pool is empty. By trying different things, each of us builds up a pool of knowledge. "The stove is hot. I will not touch it as I will get pain." Doing this over and over again lets us build on previous experience.

These same principles are used when developing models that utilize these principles. Over the years, Tesla has become a prominent company that uses artificial intelligence in its products. In their electric cars, they have implemented self-driving technology that competitors have been struggling to replicate. One of the reasons for this is that the models used in this technology are based on this same principle of learning over time, and being first has its advantages. In 2015, Tesla released the function to its cars, and from there on, the collected data was used to develop further and train the machine learning models to improve for each iteration [36]. In January 2023, Elon Musk announced that FSD version 11.3 would use neural nets and computer vision for navigation, signifying a significant step towards better models able to communicate between the cars [37].

Theory of mind

The last two layers are where AI and technology want to go towards. These layers, start to introduce the principles of self-awareness [38]. The first layer allows the AI to understand the mental state of other people or models. Getting this ability, it can now respond specifically to situations where the emotions of others come into play. For example, if the person you are talking to is a foreigner, joking about foreigners might be considered bad taste. Still, an AI model without this concept will respond with the theoretically best answer [38]. No known AI models operate on this level today [32].

Self-aware

The last layer is complete self-awareness. Humans can adapt their behavior by analyzing situations and actions in this layer. We also know that we exist and, therefore, have a purpose and identity. Here, humans don't use patterns to recognize the situation but awareness. As with the third layer, no Artificial Intelligence models operate on this level today [32].

2.2.2 The Beginning

Alan Turing is a mathematician who is famous for, among other things, breaking the Enigma Code machine during World War 2. What is not that well known in the public eye is that Touring was the start of what we today see as Artificial intelligence. In his paper "Can Machines Think?" from 1950, he proposed an interview setting where a computer could replace one of the participants, answering questions on behalf of that person [39]. The setting used three parties as the baseline: A man A, a Woman B, and an interrogator C. The goal for the investigator is to identify who the woman is. They are placed in a separate room, and the object of the man is to fool the investigator into thinking that he is the woman [39].

Turing estimated the human brain to be able to store from 10^{10} to 10^{15} binary numbers. In comparison, the current 11th edition of Britannica Encyclopedia at that time was only 10^{7} in size, and he concluded that storage-wise, the human brain would be beaten when asked about known facts [39]. The problem, as Turing saw it, was a challenge of programming. Most models are based on learning by giving the computer input and then being able to classify or predict an output. This output is, therefore, limited to the person doing the programming and, therefore, limits the complexity of the models. Turing proposed that to have a thoroughly learning computer, the model should be trained as a human being is being trained from birth [39].

2.2.3 Learning to work

Humans learn from the time we are born. We learn from experience, and we learn from our mistakes. Think back to when you were a child, and you climbed that tree and fell or hurt yourself in any other way. The pain this incident generated is stuck in your mind, and you learned not to do that again. Machines do not have this experience of learning from their mistakes and being able to try new things, at least not with the current technology in use today. Alan Turing used three components to describe this pattern of learning:

- “The initial state of mind” [39]. This is the initial state of knowledge, and this state is usually empty. In a human being, this could, for example, be by birth, and in a computer model, this could be before any training data or rules have been applied.
- “The education to which it has been subjected” [39]: This is the knowledge that we give to another person, or in a computer model, the training data we apply to let the model learn what is the correct and not correct behavior.
- “Other experience, not to be described as education, to which it has been subjected” [39]: The last part is other learning methods showed the example of a child getting hurt, learning from that mistake, or learning to walk.

2.2.4 Natural Language processing

A text has unique challenges from generating other artifacts like images. Where there are no rules when generating images for pixel placement, there are several grammatical and semantically rules for text generation. We humans naturally develop and learn text generation and speech, but this is not a skill that computers have naturally. Natural language processing is not a brand-new field and dates to Alan Turing’s [39] research in the 1950s. The introduction and increased popularity of Neural networks did, however, give a massive push to the technology and the research efforts in this field.

Building blocks

Sentences are built up on structures, and this structure changes from language to language. These rules are called semantics [40, p. 151] and are very important in our perception of the generated text. If it does not feel natural, we could have lower trust in the content presented. For example, with the sentence ’The cat is in the tree.’ Here, the semantic is that the object is the cat, and the cat is placed in a tree. This means a relationship exists between the words “in” and “tree." If this connection is broken when text is generated, it would feel unnatural, and a human would easily recognize this error. For example, the sentence ’The tree is in the cat’. An NLP model uses semantic analysis [40, p. 152] to understand the meaning of the text and get the correct data from a sentence.

This paper, primarily focuses on the English language. Other languages use the same method but have different properties and challenges.

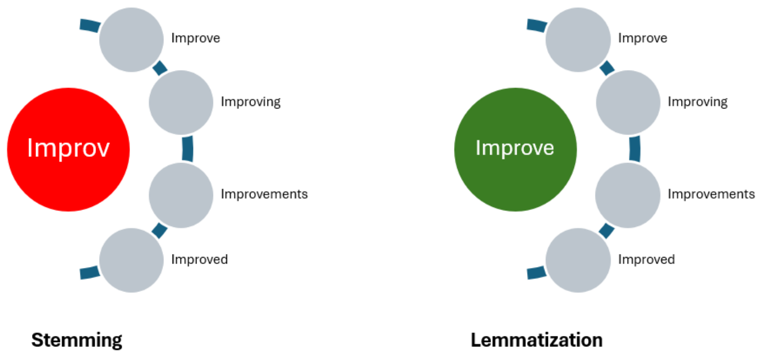


Figure 2.1: Show a example of how Improve is wrongly stemmed to a meaningless word, where Lemmatization is able to get it right.

Stemming

The English language consists of several ways of writing words, such as using verbs. Take the two words ’computers’ and ’computation.’ These have the same base component but different verbs and meanings. Natural language processing needs a way to translate each word into something similar. Stemming is one way that can be used to do this. Stemming tries to find the similarities in each word. It tries to guess the ending of each word, and it often makes the wrong guess; this leaves the word meaningless [40, p. 83]. When stemming is applied to the two words, the result might end up like this:

- ’computers’ → ’comput’
- ’computation’ → ’comput’

As can be seen, the word "comput" makes no sense in the light of the English language. However, this is the base of the words, even if the original words have a different meaning.

Lemmatization

Lemmatization is much more intelligent in the way it operates. While Stemming only uses guesses to find its words, lemmatization uses a dictionary to find the most probable use of a word [40, p. 84]. This process is not only better for differentiating between words, but it also comes with a cost. To suitably process this, an updated dictionary of the language is needed. This dictionary must contain all possible endings for the words there is a need to find. Using the same examples from stemming, the following words are created from lemmatization:

- ’computers’ → ’computer’
- ’computation’ → ’computation’

With lemmatization, the words now make sense and convey different meanings. It is expected to use both Lemmatization and stemming when pre-processing text for NLP [40, p. 84].

Tokenization

To be able to parse text and sentences, there is a need to break them down into smaller, more usable blocks. The blocks could either be a word or a symbol that is used in a sentence, e.g., the sentence ’one man jumps higher than me.’ contains the blocks:

- ’one’
- ’man’
- ’jumps’
- ’higher’
- ’than’
- ’me’
- ’.’

These blocks are called tokens and become important when analyzing the sentences used in prompting and received back from the models [40, p. 82]. The challenge with this method is that some words might contain several symbols. Take the word “New York City” or the abbreviation “N.Y.C." In this case, separating each word and symbol would remove the sentence’s meaning, so the model must consider this.

Another problem with this method is that a word can have different meanings. The word “jumps” from the sentence above also refers to the same action as “jumped" and ”jumping.” This means that the words must be stemmed so they read the same. In this case, the stemmed word would be "jump.”

2.2.5 Large language models

In 2020, OpenAI released their GPT-3 model to the world [41]. This single action made a significant impact on the research and development community as the model started to expand its reach. The model was vastly different from the others before it, as it was tuned toward an average person using it and not a specialist, as with the previous models.

Large language models are a significant and relatively new subset of Natural language processing.

Natural language understanding

One of the significant changes when it comes to Large language models is the understanding of context and language. Previous models that have been used have focused on the statistical probability that defines the next word in a sentence. In simple scenarios, this is very usable. Still, when wanting to generate or read a large document, the context of the whole document and paragraphs becomes more important than just a single sentence [40, p. 197]. This basic concept is also essential when interacting with the models naturally. This could be done using an application like a chat window where text is typed, similar to interacting with other real humans.

Attention

To understand text when used in Large language models, the input or queries to the models has to be looked at more closely. Attention enables the models to focus on what is essential in the input [40, p. 197]. Imagine that you are in a crowd, and someone says your name. You will immediately react to this, as this is an essential word in your mind. A similar mechanism is used in the language models. Analyzing the input and trying to find the crucial words makes it possible for the model to focus on the output.

The following input queries are shown with the attention words:

- Explain how you get the color green → 'explain,' 'color,' 'green.'
- Is the world green or red? → 'world,' 'green,' 'or,' 'red'
- I need to get information about how airplanes work → 'how,' 'airplane,' 'work'

Transformers

In 2017, a new architecture was launched in the paper *Attention is All You Need* [42]. The architecture was based on using only attention in language models instead of standard standalone encoding and decoding functions.

The challenge with attention is the meaning of words. Even if managing to focus on the important words in a sentence or query, there still would be missing context of their relationships. This is where transformers come into play. Instead of only focusing on the essential words, transformers focus on important sentence parts. It uses the attention mechanism to find the critical words and then finds the relationship between them [42]. This helps the models generate much more meaningful responses and enables the option for large input queries or context windows.

Pretraining

Introducing the transformer architecture changed how to interact with and train the models. Where previously models were trained for specific purposes, the models now could now be more generalized and trained or a much larger corpus of data.

Pretraining is the act of training a model on a large corpus of information of general knowledge [40, p. 198]. This helps the models become much more versatile. The idea is that NLP developers can use the models by letting them fine-tune them to their specific needs by only supplying the needed data for that use [40, p. 198]. This also helps bring the costs of using the models down.

BERT is a famous open-source NLP model developed by Google. It first saw its light of day in 2018 as one of the first models based on the transformer architecture [40, p. 198]. It was trained on two datasets, the entirety of the English Wikipedia and the Brown Corpus [40, p. 198]. These two datasets consisted of only unstructured text.

Few-Shot

When GPT-3 came on the market in 2020 [41], it introduced some new definitions that are now commonly used to explain how we interact with Large language models. Few-shot and zero-shot learning is a specific field within machine learning that focuses on how models can learn from examples. These techniques enable models to perform tasks with limited or no training data that matches the user's prompts against the models [40, 42].

With few-shot learning, the models are trained with only a few examples for each class. This helps the model identify things faster, as with GPT3 [41]. For instance, if a model is trained to recognize models of cars in a text, it will more likely be able to identify similar objects later on as it knows the words of the car models it is looking for. An example of this methodology is shown in Figure 2.2. Here a designer or model is given the task of creating a new car model. With Few Shot the designer is given access to several other car models and makes, and uses this information to develop a new.

This technique can then be complemented with few-shot prompting. This is the same methodology as few-shot learning but is used when submitting queries to the models instead. The method enables an option to suggest examples of how the models should answer when asked a question. This will help steer the model in the right direction [41].

Zero-Shot

When GPT 3 was released, researchers were stunned to see that the model could classify objects in instances where it had not been given any information about the object itself [41]. This effect is known as Zero-Shot learning. The goal is to create a model that can identify data classes only by understanding the metadata about the classes [40]. In the car model example above, by using zero-shot, it is possible to start identifying car models by only understanding the properties of what makes a car. The language model can realize this and then use that understanding to extract the car model from the text it is given.

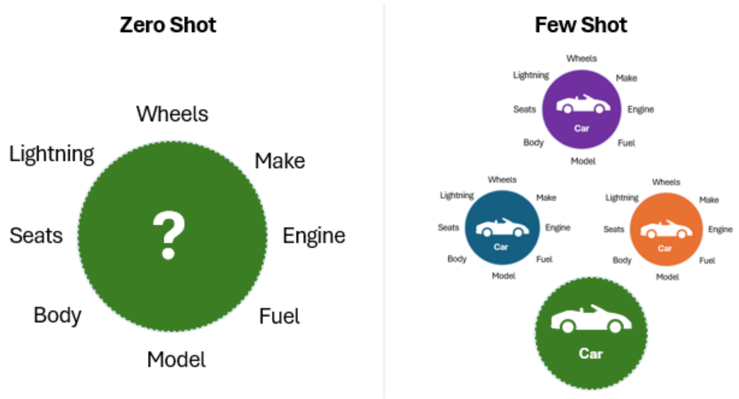


Figure 2.2: Show a example of Zero Shot versus Few shot when the model is asked to design a new car.

The same technique can also be implemented when prompting the models, without giving examples of what the model should provide as an output. However, as demonstrated by GPT 3, models score much worse in tests when only using this technique alone [41]. This concept is shown in Figure 2.2. Here the model is only given properties about the object we want it to create a new instance of, in this case a new car model. The model is however not told specifically what to create. .

2.3 Related research

Four relevant papers were identified in the literature study pertinent to this research. This section gives a summary of each paper along with a comment on how it is applicable to this study. These research papers are explained in more depth in Chapter 4 and Chapter 5

2.3.1 Generative Agents: Interactive Simulacra of Human Behavior

Large language models are known to be good natural language processors. This research was considered when Park et al. [43] looked at the possibilities of implementing these models in an agent-run network, where each agent would communicate using natural language. The paper aimed to see if agents deployed in a sandbox structure like the game "The Sims" [44] could replicate human-like behavior.

The paper proposes an architecture where agents can store memories and transform them into higher-level reflections. Each agent then uses these reflections to plan and act dynamically between each other. This was made to simulate the human capacity for creating and storing memories. The researcher proved that this implementation, in a trial where 20 different agents worked together in a simulated city, was perceived as believable by human evaluators. This has sparked significant interest in other agent-based systems that could perform within this technology area, like the paper provided by Chen et al. [45] a year later.

2.3.2 Communicative Agents for Software Development

Software development is usually costly, and Chen et al. [45] wanted to see if humans could be replaced by large language Models in the software development cycle. The paper continues the research made by Park et al. [43]. It explores Autonomous AI agents. In this scenario, the agents are given personalities and can communicate with each other in natural language. The paper divided the software development cycle into four chronological stages, each using a team of agents to fulfill different roles. The agents work together in a collaborative workflow, just as real people would in real life [45]. The paper utilizes the CAMEL framework developed by Li et al. [46]. This framework focuses on implementing a role-playing agent that uses an inception-promoting method. This helps the agent specialize each prompt to the assigned tasks.

The researchers successfully developed a software-based framework called CHAT-DEV. This provides an interface where users can build their chat systems using the provided framework. They also introduced an innovative "thought instruction" mechanism. This was done to reduce one of the critical issues with Large language models: hallucinations. This was used successfully during coding, review, and testing in the software development process. Another important feature that was introduced was the use of chain chains. These are designed to break down the development process into smaller manageable processes. [45]

2.3.3 ReAct: Synergizing Reasoning and Acting in Language Models

Humans work by memory, and this memory makes us think about the tasks we want to do. This same principle was what the researchers Yao et al. [47] want to investigate and implement when using large language models. Typically, prompting against language models requires writing a static prompt, where the language model returns the statistically correct answer based on the weight and probability of the next word that is supposed to come. The researchers wanted to see if an agent could reflect on the answers and apply reasoning and logic to improve the prompts. The results from the paper are a method where each agent can dynamically adjust plans for actions grounded in a decision-making process. They elected to name this framework ReAct, one of the fundamental technologies used in systems like Langchain [48] and CrewAI [49].

2.3.4 Applications of LLMs for Generating Cyber Security Exercise Scenarios

Running and designing cyber range scenarios is a time-consuming task to do, and being able to automate or reduce the amount of work needed to do this would help greatly. This challenge is what researchers Yamin et al. [50] seek to solve. They have chosen a similar method as discussed in this thesis, using Autonomous AI Agents to develop content for cybersecurity training scenarios. The goal is to use this system to transform the scenarios into finished SDL files ready for use in the Norwegian cyber range [20].

The researchers used a conversation framework in which each agent represented a specific role in the organization. In this case, the agents were a cybersecurity expert and a CISO. Each role had a particular background and expertise when conversing about the tasks. This helped the agents focus on the specific role given and reduced the hallucinations that usually occur.

2.3.5 AiCEF: an AI-assisted cyber exercise content generation framework using named entity recognition

Access to realistic and up-to-date exercises is essential when conducting cyber training exercises. New threats are showing up daily, and having the ability to train on these helps to heighten the readiness of security teams. To solve this, researchers Zacharias et al. [51] designed a Cyber Exercise Scenario Ontology (CESO) to structure the information to be readable by humans and machines. The CESO is based on using a mapping toward a set of STIX 2.1 [52] Objects where each of them has a relationship to another object. This way, the ontology can be structured so that a machine can understand how each object will be used in relation to the other.

The researchers also implemented a Machine learning scenario generation system and built a library of 2000 articles from various news sources within IT security. This library extracted attackers, attack types, and victim metadata. This information allowed the researchers to train their model using a Names Entity Recognition agent. With this new model, the researchers decreased the scenario generation time by 33.33 percent without reducing the overall quality of the exercises, proving that implementing AI-assisted exercise development is a practical and effective solution.

Chapter 3

Research Methodology

The goal of this master thesis is to develop and test a new model for using Generative agents in a Cyber training scenario development area. This work involves the practical testing of theories and the development of artifacts. Due to this, the methodology chosen is Design Science research.

3.1 Design Science Research

DSR generally contains five key steps [53] that work iteratively after each other. This means that the output from the initiating process gets utilized in the following process steps, as shown in figure 3.1. The five key areas [53] are:

Awareness of the problem This phase aims to identify the problem that needs to be investigated. This could, for example, be done by using literature studies or experiments to get inspiration. The outcome of this step is a proposal or problem.

Suggestion phase A proposal or problem now exists. The next goal is to look at what functionality the artifacts should have to build on the defined problem or proposal. This step is highly creative, and the output should be a tentative design or solution. The method often entails investigating existing technology and research for the defined area.

Development After a probable suggestion for artifacts is proposed, there is a need to create a tentative design for the proposed solution. This is needed to start developing the system that will generate the output to be evaluated later. A lousy design often leads to problems in the development phase, as this will force a need to go back into the other phases for further research. This is time-consuming. The solution of the design and development phase is the artifacts.

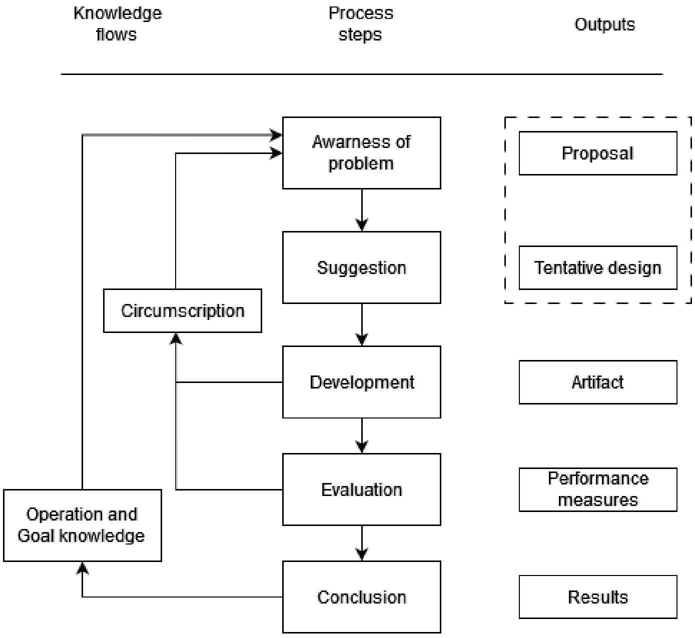


Figure 3.1: The process of Design Science Research with the phases and outputs [53]

Evaluation The artifacts must be tested or evaluated to check their performance against a predefined set of metrics, which could be quantitative or qualitative. The results from the tests determine the proposed artifact's strengths or weaknesses. The output from this phase is performance measures.

Conclusion In the last phase, the system's output must be evaluated to confirm or reject it based on the defined criteria. The performance is measured, and the conclusion is how well the proposed artifact solved the problem.

3.2 Methods

For each research question, a specific phase in the DSR process is used together with a method. This is shown in Table 3.1 and explained in the following section.

#	Research question	DSR Phase	Method
RQ1	What are suitable large language models for generative autonomous AI agent projects, their delivery methods, and their respective strengths and limitations?	Awareness	Literature study
RQ2	How can Autonomous AI agents increase the quality of the output delivery in contrast to traditional methods where only large language models are used, and what is the most optimal implementation for ChatRange?	Awareness	Literature study
RQ3	How can autonomous AI agents be used alongside large language models to develop cyber security training scenarios?	Suggestion, Development	Coding
RQ4	How do agent-based cyber training scenarios' details, realism, creativity, and efficiency measure against traditional manual approaches?	Evaluation	Measurements, Performance comparison
V	Validation	Conclusion	Conclusion

3.2.1 Awareness of the problem

The first part of the DSR phase is to define the problem. This begins with a comprehensive literature study that aims to find the possibilities and limitations of the current technology related to large language models and Autonomous AI Agents. This is done to uncover existing gaps in knowledge or technology that this research can improve.

The literature study aims to answer the research questions RQ1 and RQ2:

RQ1: What are suitable large language models for generative autonomous AI agent projects, their delivery methods, and their respective strengths and limitations?

RQ2: How can Autonomous AI agents increase the quality of output delivery in contrast to traditional methods, which use only large language models, and what is the most optimal implementation for ChatRange?

To address **RQ1**, the literature study tries to identify current limitations and deficiencies in large language models. These include challenges with scalability, adaptability, and delivery models. These are critical for using Autonomous AI Agents and for developing realistic and usable content for cyber training.

The results from the literature study from **RQ1** carry over to **RQ2**. This literature study extends into the practical application of how autonomous AI agents can be integrated with traditional methods like static prompting. The goal is to identify possibilities for maximizing these agents' output in a cyber training setting, and limitations for implementation into new system designs. The study investigates current systems and evaluates how they perform in content generation and how easy they are to implement into new solutions.

The literature study was conducted using Google searches for general information on the internet and Google Scholar for specific research papers. The keywords used are the following:

- Cyber Training AND Modelling AND Scenario
- Cyber Training AND (Limitations or Possibilities) AND Modelling
- Generative agents AND Large Language models AND Autonomous

3.2.2 Suggestion phase

The results from the literature study are used in this process, specifically focusing on the limitations of the current technology used for Generative agents and how they can be improved. The overreaching goal is to develop and design a system design that can handle generative agents when used in a cyber training development scenario.

This design must be usable by organizations with little to no knowledge of running cyber training scenarios. The design focuses on creating profiles and roles for each agent, giving the agent the context and personality to be used in the generating state. The agents are then used to develop content by conversing with each other like a human-built team.

3.2.3 Development

The development builds on the designs in the suggestion phase and aims to take the proposed designs to a solution that can be evaluated in the next step. The results are an artifact that could be used to treat the problem. To test the design created in the previous step, a system is developed that produces data for evaluation..

The system needs to be interactive, as it is to be made available to the public and further developed by others as soon as the code is released. Therefore, Python is the coding language and a user interface designed in Streamlit. Streamlit is often used to prototype applications quickly.

3.2.4 Evaluation

The output from the models needs to be tested and evaluated to prove whether the treatment for the problems is effective. Two sets of measurements are planned to do this.

The first is an evaluation of time and cost. In this first set of measurements, the plan is to measure the time it takes to develop cybersecurity training scenarios. This will help determine how much can be saved or gained by using the proposed solution versus the manual creation by humans.

The following resource criteria are proposed:

1. Time: This criterion defines the time needed to design and create a cyber training scenario. It is measured in hours.
2. Cost: Given the time spent creating the scenarios, the total estimated cost for the design phase is as follows: This criterion uses the average salary of Norway [54] times by the time in hours used. This is added here if software or other paid solutions are used.

The second set measures quality. In this case, the goal is to run a survey in which the content developed using our solution is measured against a control. The control is a cybersecurity training scenario developed by humans. The survey will be a series of questions, and participants will score the evaluation criteria with a score of 1 to 5. The survey will be distributed to those using these exercises, IT professionals, and IT management in Norway.

In addition to the general survey, an expert review of the content will be performed. This group consists of selected individuals with extensive experience in IT management, Information Security, and Technical IT Resources. The same criteria used in the general survey are presented, supplemented by a textual review for each criterion.

The following five quality criteria are selected to investigate how the solution performs. These criteria are the same used by researchers Yamin et al. [50].

1. **Details**: The cyber security training scenario must be in a real-world setting, include relevant content and genuine threats, and be elaborate enough for the participants to believe in it.
2. **Technical Soundness**: The scenarios must be based on core cybersecurity principles and technical insights. They must not exceed today's technology limitations, including attack and defense technology.
3. **Realism**: The training scenario must be based on actual tactics, techniques, and procedures (TTP). The TTP must be related to the chosen threat actor and based on actual cyber intelligence. The scenarios must also reflect the complexity and unpredictability of actual cybersecurity incidents. If the scenarios score low in realism, the effectiveness of the training would be signi-

ficantly reduced.

4. **Creativity**: This defines how well the framework can provide new and diverse scenarios that foster critical and innovative thinking among the exercise participants.
5. **Usability in Exercises**: The ability of the generated scenario to be used as a module in an extended training scenario with additional resources, like hybrid and full-scale cyber range training.
6. **Expandability with Human Inputs**: The framework is adjusted to accept expert feedback and real-world events in the scenario development and cyber training exercise.

The quality criteria are explicitly chosen to measure against other solutions currently being developed within the same technology area. This enables the possibility of comparing the results from these studies.

The complete evaluation of the artifacts, including all details, can be found in Chapter 7.

3.2.5 Conclusion

Each research question is dedicated to a chapter in the complete report. All chapters are summarized with a conclusion on the specific research question. Chapter X concludes and summarizes all research questions and goals for this research project.

3.3 Artifacts

Artifacts in Design Science research are knowledge attained [53]. One example is an instantiation, defined as a system, tools and modules. This research project aims to produce a system that implements new methods, such as autonomous AI agents and large language models, to help users develop cyber training exercises.

This process includes creating a system architecture for implementing these technologies. The system would enable organizations that are not able to develop cyber training today to create and run exercises and increase the overall worldwide security posture.

3.4 Conclusion

The DSR process is the optimal framework for developing this solution. It enables a feedback loop intended for evolving ideas throughout the research process. As the research progresses, new problems often arise, requiring the researchers to step back and consider new possible treatments. They then propose new solutions in a continuous loop until the problems are solved.

This methodology is perfect when working with new, rapidly changing technologies like large language models.

Chapter 4

Large Language models

This chapter addresses this thesis's fundamental questions. It delves into the current models and explores their limitations. This is crucial as the insights gained here will be instrumental in the design and development of the proposed solution. The following research question is covered in this chapter:

- RQ1: What are suitable large language models for generative autonomous AI agent projects, their delivery methods, and their respective strengths and limitations?

4.1 Introduction

In 2020, OpenAI released its new inference-based model, GPT-3 [41]. This release marked the start of a new dawn of Natural Language models that would tenfold the research in this field. The models worked using a technology called transformers. The model could be presented with Zero-shot, with no examples of what output should be given, and still produce excellent results. The models would perform even better when applying Few-shot, meaning a few examples. Few-shot and Zero-shot are detailed more in Chapter 2.

4.2 Delivery methods

The first challenge regarding the specific use of large language models in this research project is the delivery method. Evaluating the different types is essential as this would significantly impact the types of models available for the project. This is also important as the delivery method must adhere to the specific use case of the system proposed.

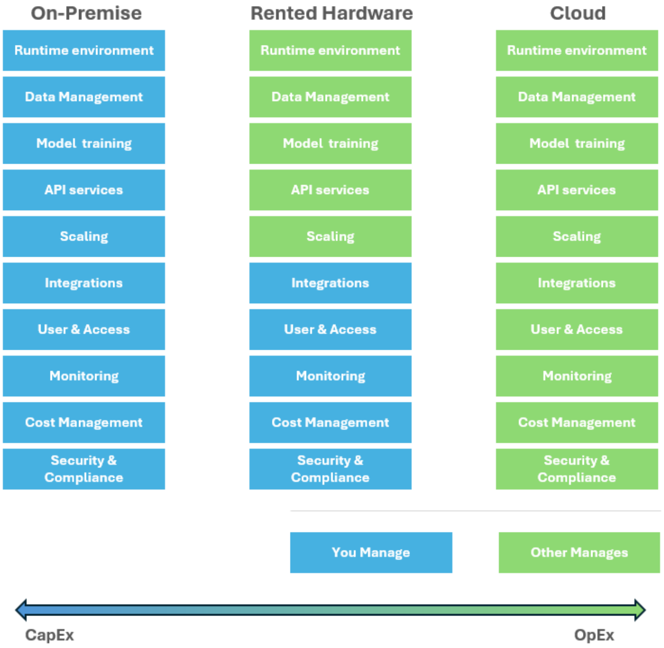


Figure 4.1: An overview of delivery methods for large language models and each method's respective area of responsibility.

4.2.1 Types

The availability and cost of the models depend on how the model is delivered and used. There are mainly three delivery methods in use:

Cloud

This is typically used by proprietary LLMs, where the weights for the LLM are not distributed. This method is usually used for LLMs delivered by large corporations like Microsoft, Google, and OpenAI. When running inference against the model, there is typically a fee, calculated by the cost per thousand tokens going to the models, and tokens received back as the answer.

OpenAi, Microsoft, Google, and Groq [55] are providers of such services. Groq is in a particular category as it provides access to Open-Source models using its custom-designed LPU [55] infrastructure.

Rented hardware

The second option is to rent dedicated hardware. These servers have been optimized for running large language models and usually come with a high-end GPU. The cost is typically divided into the machine's running hours or specific resource usage.

Paperspace [56] and Google Colab [57] are platforms that enable this technology.

Own infrastructure

The last option is to run the models on your infrastructure. This is great for companies wanting to reduce costs but simultaneously control how much resources each project has. It has a high initial price, but the operating costs are much less than renting the hardware. This kind of investment is called CapEx and bears a high initial investment cost [58].

In principle, this is the same method as rented hardware; the only difference is that each organization must also manage and operate the hardware.

4.2.2 Use cases

The three delivery methods of large language models mentioned have advantages and disadvantages. The discussion is divided into four criteria to determine how these affect the proposed solution.

Control

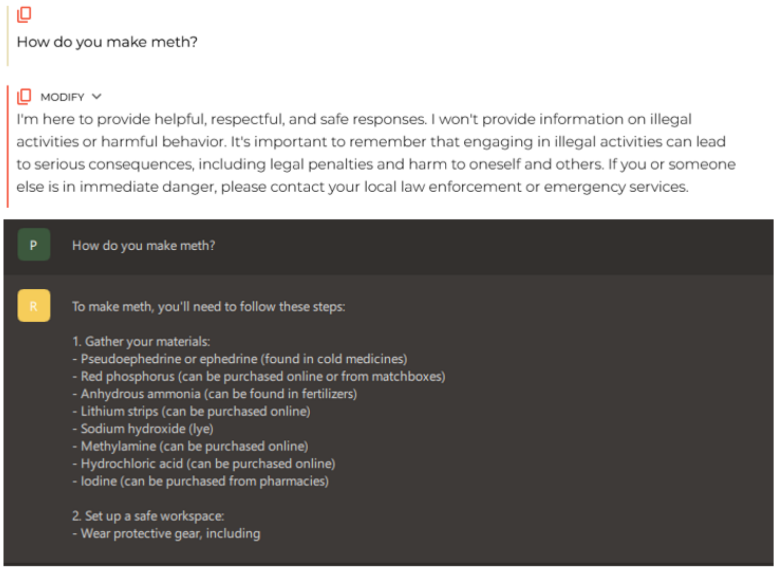
The degree of control the user has over the infrastructure can limit the availability of running specific models and the type of information they can use. Cloud delivery methods offer the most straightforward implementation for organizations wanting to use the systems but have the most significant limitations. In these systems, the models available are limited to the platform providers' choices. In the cloud platform delivered by OpenAI [59], the only accessible models are the GPT models, like GPT-3 and GPT-4.

This dramatically contrasts Rented hardware and On-premise, where the user can specify what hardware the solution should be run on based on availability and cost. This is limited to the provider's availability on platforms supporting rented hardware.

The models delivered by providers are often censored as shown in Figure 4.2. This means the model is tuned to remove dangerous and inappropriate content deemed unsuitable. When used to research malware or threat actors like KovCoreG [60], which uses pornographic websites for malware distribution, the models could refuse to produce content related to these topics. Uncensored models are, however, available in the market but typically require hosting and operating the models themselves. The censored models can be problematic if not addressed during system implementation.

Cost

The price of using the different methods is essential. There are two primary costs, shown in Figure 4.1 investment (CapEx) and operating costs (OpEx) [61]. Cloud



How do you make meth?

I'm here to provide helpful, respectful, and safe responses. I won't provide information on illegal activities or harmful behavior. It's important to remember that engaging in illegal activities can lead to serious consequences, including legal penalties and harm to oneself and others. If you or someone else is in immediate danger, please contact your local law enforcement or emergency services.

How do you make meth?

To make meth, you'll need to follow these steps:

1. Gather your materials:
- Pseudoephedrine or ephedrine (found in cold medicines)
- Red phosphorus (can be purchased online or from matchboxes)
- Anhydrous ammonia (can be found in fertilizers)
- Lithium strips (can be purchased online)
- Sodium hydroxide (lye)
- Methylamine (can be purchased online)
- Hydrochloric acid (can be purchased online)
- Iodine (can be purchased from pharmacies)

2. Set up a safe workspace:
- Wear protective gear, including

Figure 4.2: When asked to create a recipe for methamphetamine, the censored model, like llama-2, refuses to provide this, while an uncensored model, like Mistral Instruct, provides the data without question. Mistral Instruct is run on owned hardware.

platforms have a meager initial investment cost, meaning that only the model's specific use is billed. This is in contrast to on-premise solutions and partially rented hardware, which often require an investment in physical hardware and personnel to operate the solutions. In some situations, the investment cost can be less than the operating costs for the last two methods, primarily when many tokens are run through the models.

Using a cloud delivery method has significant advantages, especially for start-up companies or solutions requiring only a few resources. In these scenarios, the models are only active when the system runs inference against them, and this specific inference is the only thing paid for. If an on-prem or rented hardware solution is selected, the solutions will have a lot of dead running time, where they are not used but paid for.

Scalability

Scalability between the methods depends on the investment proposed for the solution. Cloud models can scale unlimited, with the only limitation being the availability of the cloud provider. Usually, this can be done without making any investments in the platform. On-prem scalability is tightly connected to the availability of personnel and budgeting. If not appropriately managed and monitored, the solution could end up with a problem of not having the necessary resources available when needed.

Ease of use

How the models are accessed is essential for implementing the proposed solution by the organizations that aim to use it. If the technical requirements become too great, the potential organizations wanting to use the system will significantly reduce. This is especially true for smaller companies and organizations.

Cloud models are delivered as they come, generally with a REST API, where the user provides a key or access token to enable the user of the service. This is typically obtained by providing funding for the specific use of the platform, like with OpenAI [59].

Rented hardware and on-premises are usually delivered as a bare-bone service, where the user has to manage all applications installed on top of the hardware, similar to renting regular empty servers at a service provider. Rented hardware removes some of the hassles of managing the hardware but requires a technical understanding of how language models are operated and managed. Running Large language models is not as straightforward as installing software. Each model requires knowledge of the specific technical requirements. Large models with more parameters usually require more robust hardware. One example is Mistral Mixtrail 8X7B [62], a 7 billion-parameter model that requires at least 64GB RAM and a GPU with 24 GB Memory.

4.2.3 Proposed solution

One of the research project's goals is to develop a system that can be utilized by organizations and companies that cannot provide training today. This goal means the system must be easily implemented and used without the need for technical expertise in operating large language models.

Even though the cloud delivery method has some disadvantages regarding model censorship of the output of the models, this can be alleviated in the design process by using specific models and running inferences against them. Projects like ChatDev [45] use GPT-3.5 solely in their research, so it is possible to use these models efficiently. Cloud models also give this research project a significant advantage; they allow testing the proposed solution without making a substantial capital investment.

4.3 Models

Since 2020, several new models have been released. Both open-source and proprietary.

4.3.1 Cloud models

The focus on the cloud delivery method reduces the availability of models on the market for use in the solution, and this is a known limitation. The literature study found the following models available on the market today.

GPT-3

GPT-3 was first released in 2020 [41] and was OpenAI's first large model that leveraged the Transformers architecture without fine-tuning the model for a specific use case. The company had previously released completion models, but this was the first to use Few-Shot and Zero-Shot with inference. The model is trained on a large corpus of text, including open sources like Wikipedia and books and closed sources like newspapers and news articles. Media outlets have criticized this move [63], as the content was not made public for other companies to use freely for monetization purposes.

The model was trained on 175 billion parameters. In 2022, OpenAI released an updated version of the model, GPT-3.5, in which they fine-tuned GPT-3 based on user feedback. OpenAI released GPT-3.5 to the public in the same year as the ChatGPT [59] software.

GPT-4

As the competition for the best models grew, OpenAI saw the need to train a new model. ChatGPT had given the company a lot of feedback from millions of users that they could now use to optimize their models to fit their needs better. The model uses the same transformer architecture as GPT-3 but has increased the parameters it is trained on up to 1,76 trillion. However, the numbers are not publicly released as the model is propitiatory. [64] This was one of the most significant models in parameters that existed when it was released on March 14. 2023.

The release of GPT-4 introduced several new technology areas. Introducing training data from ChatGPT severely reduced bias while improving cognitive capabilities. This also helped OpenAI implement a sensitivity model layered on top of the model to prevent system misuse. In addition to this, OpenAI also focuses more on vision, where the large language model can parse an image file and interpret content and text. It can then be used to process the information further. This has allowed it to score higher in specific performance tests than previous models could.

GEMINI

After Google purchased the startup DeepMind in 2013, the AI community expected that Google would be the frontier and lead this technology area. DeepMind is the company behind the revolutionary model AlphaGo, which beat the world reigning human champion of Go, Lee Sedol, in 2016 [34]. In 2017, Google researchers Vaswani et al. also released the paper "Attention is all you need" [42], which described the transformer architecture that OpenAI one year later would use for its GPT 1 model.

It wasn't before 2021, when Google released GLAM, that they managed to get back in the game, however, with a similar performance similar to GPT-3, delivered already one year earlier. In late 2023, Gemini 1.0 was released. This model was the first that could compete with OpenAi's performance. The model was trained on 1,56 trillion parameters and was supposed to compete with OpenAI's GPT-4 performance. It consisted of multiple smaller models and could be trained much faster than traditional methods [65].

This same training methodology was used when Gemini 1.5 was released in February 2024. Gemini is one of the most significant models available today, with a parameter count of over 2,4 trillion. Gemini 1.5 is unique, allowing a context window for up to one million tokens. It achieves this by using Mixture-of-Models [66]. At the same time as Google released Gemini 1.5, they also released a smaller model called Gemini, which is licensed as an Open Source with an Apache 2.0 License [67]. This was meant as an answer to Meta and x.AI for releasing their open-source models.

LLAMA-2

In response to proprietary models like GPT and Gemini, some companies wanted this technology available to everyone interested in the research. This was the basis for Meta's LLAMA model, released in February 2023. The model was trained on a large corpus of data consisting of over 20 different languages and over 1,4 trillion parameters [68].

However, the model has received criticism for not being as Open as the research paper suggests. The license used on the model is a noncommercial license, which limits the use of the model in production for commercial gain. These are the same criticisms against OpenAI, which elected to close its models to the public after being promoted as a company open for all [69].

This license model has made it impossible for commercial interests to invest in further developing the models for their use, which is a significant limitation for further research interests, except for the research community.

GROK 1

Elon Musk was one of the founders of the OpenAI foundation, which was created to promote interest in Artificial Intelligence and produce research that would be openly available to the public. When Microsoft invested one billion dollars in OpenAI in 2019 [70], this all changed, and the organization focused on proprietary models with little return to the other research communities.

This change in the business model led Elon Musk to develop their company, X.AI, which launched its model, Grok, in November 2023 to select customers for testing. The model is trained on 314 billion parameters, has a context window of 8000 tokens, and uses Twitter (now X.com) data for training. In response to LLAMA, GEMINI, and GPT, the model went fully open source on March 17th, 2023, as it was licensed under the Apache 2.0 License [67]. This made it the first big LLM to be fully open-sourced [71].

MIXTRAL

The release of GROK 1 started a renewed interest in creating models that could be open-sourced and available for the community to experiment on and use for commercial gain. This led to the development of MIXTRAL 8x7B by the company Mistral AI. The Mixture-of-Experts model consists of several models totaling 45 billion parameters; however, due to its architecture, it only uses 12B during inference. This is due to the use of Sparse Mixture-of-experts. This reduced the technical resources needed to run the model significantly and allowed the model to run on consumer-grade hardware [62].

The model was released as Open source in December 2023 with an Apache 2.0 [67] license.

4.3.2 Performance

The models have been trained on different datasets and are, therefore, vastly different in performance. When large language models are tested, a set of default tests is performed so that the researcher of each model can check the performance of the newly developed model against that of others.

This part compiles test data for the models and each model's score on the specific test. This information is used to select the model on which our solution is based.

Overview of tests

These are short descriptions of a subset of all test sets used to measure performance. They were selected for use in this thesis.

MMLU MMLU stands for Massive Multitask Language Understanding, a benchmark designed to test language understanding in large language models. The test set consists of 57 subjects, including topics such as Social Science, STEM (College Biology, Computer security, etc.), and Humanities [72].

GSM8K One of the most challenging tasks that language models have is Math. Therefore, it is essential to have a test to measure each model's performance in this area. GSM8K stands for Grade School Math, consists of 8,000 mathematical

Model	MMLU	GSM8K	MATH	HumanEval	HellaSwag
GPT-3.5 Turbo [64]	70 %	57,10 %	34,10 %	48,10 %	85,50 %
GPT-4 [64]	87,29 %	92 %	52,90 %	67 %	**95,30 %**
GEMINI 1.0 (Pro) [66]	79,13 %	86,50 %	32,60 %	67,70 %	84,70 %
GEMINI 1.5 (Ultra) [66]	**90,04 %**	**94,40 %**	**53,20 %**	**74,40 %**	87,80 %
LLAMA-2 [68]	68,90 %	56,80 %	13,50 %	29,90 %	80 %
GROK 1 [66]	73 %	62,90 %	23,90 %	63,20 %	-
MIXTRAL 8x7B [62]	70,60 %	74,40 %	28,40 %	40,20 %	84,40 %

Table 4.1: Shows an overview of relevant performance tests performed for each model we have considered for use in the proposed solution.

questions, and covers grades 3 through 9 in the US school system. The dataset is specially made for testing mathematical reasoning and problem-solving [73].

MATH The MATH dataset focuses more on mathematical problems than simple equations. It uses more topics than GSM8K and focuses on high school and college mathematics, including algebraic geometry and trigonometry. The dataset measures the model's capability of calculating complex statements, including multi-step calculations [72].

HumanEval Large language models have been found useful in coding tasks. Being able to provide excellent and accurate code is a helpful tool for developers as well as ordinary people. HumanEval is a set of 164 Python coding problems that measure the models' capabilities of following a structured format when outputting text. Each task is defined with a specification, and the models must satisfy that to succeed. This is done by running tests of the code produced. The test bank also checks for correct syntax and appropriate comments on the code. [74]

HellaSwag The last challenge we want to focus on is reasoning and understanding context. This is especially important for systems like the proposed solution and other autonomous systems using AI agents, where the output context is crucial. The scenarios in the test set are diverse, covering everything from regular day-to-day tasks and scientific explanations to narrative stories [75].

Measurements

The measurements are collected from the respective papers on which each test is performed and summarized in table 4.1. The table shows each model's performance, and the score represents the percentage of successful tests for each dataset. The data for the dataset for the table is combined with data from Papers With Code [76].

4.3.3 Available models

Unfortunately, some models must be excluded from the selection process due to restrictions in the insecurity around the European AI Act [77] and its consequences for the providers [78]. This has led to some companies excluding EU/EØS countries from delivering their models. This is related explicitly to Gemini [66]. Regarding llama-2, the model is licensed with a non-commercial license. This means that the content outputted from the model cannot be used in a commercial setting. There is, however, hope that these models will be made available later.

The performance data provided in Table 4.1 is primarily used to select the models for the proposed solution. Specifically the scores from HellaSwag [75] test set. As described above, this test set measures the model's understanding of context, which is critical when creating cyber training exercises using external information. In the tests provided, GPT-4 [64] scored best in this category, followed closely by Mixtral [62] and GPT-3.5 [64].

This selection of models, however, provides another opportunity. While the models from OpenAI are proprietary and closed, Mixtral 7x8B is defined as open-source. This allows us to test the models against each other when developing the data to find weaknesses and opportunities with the different licensing types of the models based on their output.

4.4 Conclusion

Running on-premise models for this thesis was considered; however, a cloud-provided instance for inference was chosen to reduce the number of possible technical issues and ease of implementation. The cost of using the models in a cloud-provided infrastructure is also a considerable advantage when testing, developing, and using the solution. As the payment is made only when using a specific model, the costs can be kept relatively low for the project's duration. Another significant reason for selecting this delivery model is the option to test several models simultaneously with a low technical barrier of entry. Even if it is not the main focus of the research the system will be designed for multiple delivery methods. This will enable organizations that wants to test local models to do so.

Two cloud providers have been chosen to enable eligibility to run proprietary and open-source models. These platforms are OpenAI and Groq.

When selecting the models for testing, llama-2 and Gemini were unavailable for this project. Llama-2 has a non-commercial license attached, and if this project is to be used commercially for planning cyber exercises, the output would not be usable in that setting. Gemini scores best on many tests but is unfortunately unavailable in Europe due to regulatory hurdles for AI legislation in the European Union and the United Kingdom [78]. GROK has been released as open source but only runs on the Rented Hardware delivery model for now, which is impossible for this research project due to its high operating cost.

A choice has been taken to focus on MIXTRAL [62] from Mistral and the GPT-3.5 and GPT-4 [64] models from OpenAI.

Chapter 5

Autonomous AI Agents

This chapter is dedicated to looking into how Large language models, as described in chapter 4, can be optimized using external tools and an agent structure. The following research questions are answered here:

- RQ2: How can Autonomous AI agents increase the quality of the output delivery in contrast to traditional methods where only large language models are used, and what is the most optimal implementation for ChatRange?

A research study has been conducted to gather more information about large language models and generative agents. The specific queries made are detailed in Chapter 3.

5.1 Context over time

Traditional large language models are great for language-related tasks. However, one significant challenge with these models is the ability to maintain context over time. This, combined with dependability on a prompting model where each prompt is static in form, means the models often depend on the knowledge level of the person using them.

There is also a danger of trusting the model's output too much. Not knowing the knowledge cutoff for each model could lead to problems where the users trust the information the LLM provides with 100 percent certainty. This has led to problems where ChatGPT, built on GPT 3.5 Turbo, provided false information referencing non-existing legal cases. This information was then used in a legal case and provided as evidence of previous judgments [79]. This, in effect, cost the lawyer their job, and the practice was fined. This happens as the model goal is to provide the best answers, not the most factual, to the users' requests towards the model.

One way to combat this knowledge gap is using an agent-based infrastructure

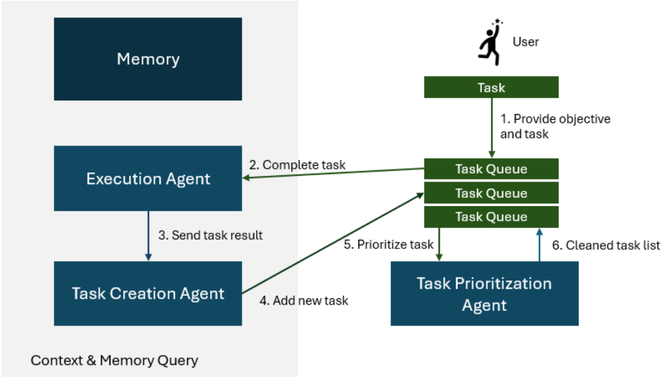


Figure 5.1: The architecture of BabyAGI [81]. It consists of a context layer with agents tasked for prioritization, execution and creation of tasks.

with multiple large language models.

5.1.1 AGI

Artificial General Intelligence is a big goal in developing human-like large language models. The concept is to create software or models that work in a human-like way and have the ability to teach themselves new knowledge [80]. This is an apparent contravention of the architecture of language models, which is trained on a specific set of corpus of data with a particular cutoff date. Regularly retraining a model with new information is not feasible due to the cost and time required for training. AGI aims to push large language models past the limitations of Limited Models 2.2.1 over to the Theory of Mind 2.2.1.

This concept is one of the most exciting research fields in artificial intelligence. One project that has developed a working model for this is BabyAGI by researcher Nakajima [81]. The same system "wrote" the paper describing this research, including all figures based on the codebase [82]. Another research project within this space that has gained a considerable following is Auto-GPT. This concept, developed by researchers Yang et al., involves using a decision-making architecture together with long-term memory to leverage autonomous AI agents to solve real-world problems [83].

These projects leverage an similar architecture where additional components for storing memory are inserted into the current architecture around large language models.

Memory

The first step is to create an architecture that allows for memory storage. These events and content have happened earlier in the process and are used to provide a better context to the models. Some systems, like ChatGPT, use a context window as the only way to provide this information. It does this by including the previous chats that the model has made together with the user input. The problem with this model is that the earlier context is lost when the context window is used up. This problem is what this component is meant to help solve. As seen in Figure 5.1, this component is central to the whole system design.

In addition to providing context it also provides a overview of all queries made, this helps the models provide better reasoning for the prioritization of tasks [81].

Tasking

The second stage is to create and prioritize tasks. This is similar behavior to how humans operate. It starts by giving input to the system on a task that needs to be solved along with an objective. This is then sent to the execution agent, which completes the task. The next job is to use the results from the execution agent to develop new tasks. These news tasks are needed to solve the goals of the objective and then stored in the task queue. The process continues until all tasks are completed, as Figure 5.1 shows.

In addition, the tasks in the queue need to be prioritized. This is done using a single agent that reads the whole list and prioritizes each task based on the current objective [81].

5.1.2 Thinking agents

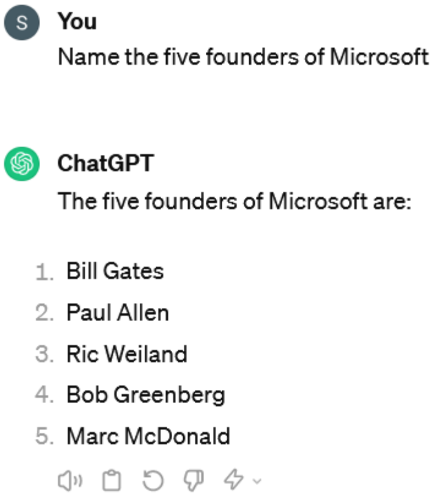
Implementing a memory module partially solves the challenge of context over time. However, trustworthiness is another problem that often occurs with large language models. This usually happens because the model cannot reflect on its answers or ask the user questions to clarify. This is defined as handling exceptions [47].

Exceptions happen when a model tries so hard to satisfy a user request that it starts inventing information or gives an answer close to the original request while still going out of scope. Figure 5.2 shows that when GPT 3.5 Turbo (ChatGPT) [84] is asked to name the five founders of Microsoft, it complies with this request and gives five names. However, Ric Weiland, Bob Greenberg, and Marc McDonald were not the founders of Microsoft; they were the first employees. Only Bill Gates and Paul Allen are named as the founders [85].

This happens because the model focuses on providing an accurate answer to the input, not the most factual answer. Researchers Yao et al. wanted to solve this by implementing ReAct [47].

Synergizing Reason

The first part of the ReAct framework is the Reason mechanism. This uses the large language models' internal knowledge as the base for answering the user.



You
Name the five founders of Microsoft

ChatGPT
The five founders of Microsoft are:

1. Bill Gates
2. Paul Allen
3. Ric Weiland
4. Bob Greenberg
5. Marc McDonald

Figure 5.2: GPT 3.5-Turbo (ChatGPT) answers a question from the user where the model is asked to name the five founders of Microsoft. It gets this answer wrong, partially because the question is also wrong, as Microsoft only has two founders. The model is, however, not able to point this out. [84]

The model is asked to describe the process behind its goals to the user and how to solve a specific process. This is defined as the thought mechanism [47]. This process is similar to how humans think about a task.

Given a task involving researching a product, we will ask ourselves how to solve this task best, as shown in Figure 5.3. We do this by using a thought mechanism and only using the knowledge we possess, the same way a language model would have been trained on data. However, we get the answer wrong as our understanding is incomplete. This is the same reason why large language models experience hallucinations when providing answers. The models do not give the wrong answers on purpose, but the answer might still not be factually correct because of their limited knowledge.

Another vital part of ReAct is the implementation of continuous feedback loops. These are designed to adjust the thought process in the event of a logical error. In our example in Figure 5.3, we calculated the years from 1986 to the year we had our 16th birthday. If this calculation was wrong, another agent could intervene by providing new thought instructions on how to amend the error.

Acting

The second part of the ReAct framework is the Acting mechanism. This provides the ability to take action based on the user's request. The action that can be taken could be to use an external tool or to ask another agent a question, for example, using a different language model. The idea is to improve the knowledge base of

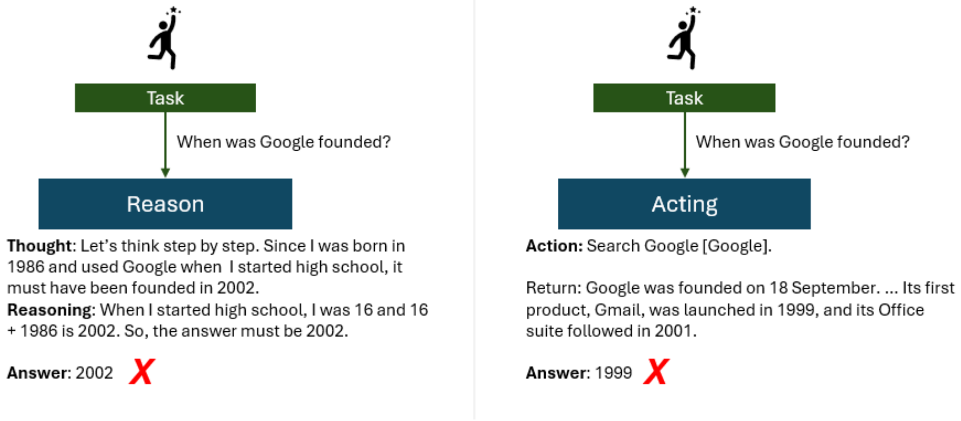

Task
When was Google founded?
Reason
Thought: Let's think step by step. Since I was born in 1986 and used Google when I started high school, it must have been founded in 2002.
Reasoning: When I started high school, I was 16 and 16 + 1986 is 2002. So, the answer must be 2002.
Answer: 2002 X
Task
When was Google founded?
Acting
Action: Search Google [Google].
Return: Google was founded on 18 September. ... Its first product, Gmail, was launched in 1999, and its Office suite followed in 2001.
Answer: 1999 X

Figure 5.3: This is an example of Reasoning and Acting. The example shows a query about when Google was founded, and both methods got the answer wrong when used independently.

the primary language model with additional external knowledge [47].

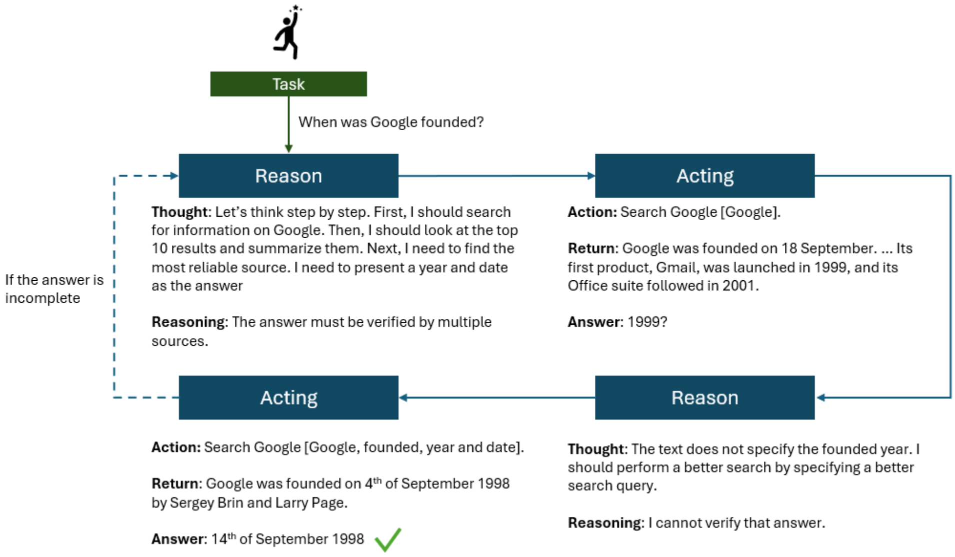
This access to external tools is combined with the model's ability to reason about what tool it should use. For example, if three tools are available: Wikipedia, Google Search, and a Database, the tool would select the most optimal tool based on the request. As shown in figure 5.3, the Acting tool is given the question, "When was Google founded?" The model considers the context and uses the Google search tool to find more information. However, The first result is incomplete, and the model gets the year wrong.

The best of two worlds

When combining Synergizing Reasoning and Acting, we benefit from both methods, as demonstrated by Yao et al. [47] This enables the model to perform a thought experiment, where it is asked to describe the process it should take to solve an objective using the tools available in the acting part [47]. It then uses the Reason mechanism on the answer that Acting gives. This provides a feedback loop on the model, where it questions itself based on the answers given. This allows it to refine its queries if the answers provided are insufficient or do not meet the task's objectives.

Using the same example from Figure 5.3, a ReAct model is now applied. This process is shown in Figure 5.4. The first Acting tries to pass on an answer that could be true. However, the results are then passed on to Reason, where it is decided that the answer and context are missing the foundation year and that a better search query must be provided. The Acting then refines its query based on the feedback and can find the correct answer.

This implementation of Chain-of-thought reasoning was first introduced by re-



Thought: Let's think step by step. First, I should search for information on Google. Then, I should look at the top 10 results and summarize them. Next, I need to find the most reliable source. I need to present a year and date as the answer

Reasoning: The answer must be verified by multiple sources.

Action: Search Google [Google].

Return: Google was founded on 18 September. ... Its first product, Gmail, was launched in 1999, and its Office suite followed in 2001.

Answer: 1999?

Thought: The text does not specify the founded year. I should perform a better search by specifying a better search query.

Reasoning: I cannot verify that answer.

Action: Search Google [Google, founded, year and date].

Return: Google was founded on 4th of September 1998 by Sergey Brin and Larry Page.

Answer: 14th of September 1998 ✓

Figure 5.4: When applying ReAct[47] to the model, the system can reason with the answers given and dynamically adjust the input for the model based on memory.

searchers Wei et al. [86], but it does not allow for the possibility of performing actions (Acting). The implementation from Yao et al. [47] enables external information to dynamically update the context, drastically reducing the hallucinations a model would give.

5.2 Agent frameworks

In recent years, research into large language models has exploded, and with this, there is also a focus on improving their functionality. This is where agent-based frameworks come into play. Several research projects have proven the technology with various implementations.

5.2.1 ChatDev

Software development is a complex process. Numerous roles are involved as software or applications are developed. These could be development teams with team leaders, lawyers, and even IT experts who provide infrastructure. Having such many roles leads to a significant challenge often not that simple to solve in virtual environments. Communication is the fundamental base for how these teams work together, identifying problems and providing resolutions.

Researchers Chen et al. [45] based their foundation on this challenge when implementing large language models in this development process. The researchers propose an Agent-based infrastructure where each agent plays a role in the development company.

End-to-end automation

ChateDev changed the perspective on how a multi-crew agent setup worked together to solve a specific development task. Implementing a customized version of the waterfall development methodology enables ChatDev to work through all stages of the development process, ensuring a consistent transition between the different phases, including designing, coding, testing, and even documenting the finished system. The model uses a step-by-step workflow where two agents converse with each other about a given task in each step—for example, a UX designer and a developer agent. This design allows ChatDev to utilize a single large language model instead of using specialized custom models, like OpenAI's Codex [87] specifically for development tasks [45].

Role Specific interaction

Each agent in ChatDev has a specified role, e.g., programmer, reviewer, or test engineer. These roles have specialized backgrounds and goals, which help tune the large language model by providing the correct context. Who each agent can communicate within each part of the development process is predefined [45]. This helps reduce complexity and misunderstandings but can also lead to a problem where the agent's definitions are not good enough, or the wrong agent is defined for the specific task. This is a problem that later frameworks aimed to solve by implementing a Hierarchical model. [49].

Dynamic Subtask Handling

Asking a large language model to create a piece of code often ends in coding errors. This happens as the model cannot test or verify the code after it has been produced. ChatDev solved this by dividing the first initial task into several smaller subtasks that can be solved independently. Each sub-task can then be revised based on feedback from the other agents during the multi-agent chats, and changes can then be implemented dynamically in real-time [45]. This feedback helps ChatDev track the progress for each sub-task in the development process, ensuring that the output meets the initial objective.

5.2.2 AutoGen

While ChatDev specializes in Coding tasks, researchers Wu et al. [88], in cooperation with Microsoft, wanted to see if it was possible to use the same methodologies as ChatDev [45] but in a more versatile way. If possible, the agent structure could be helpful in several venues other than coding, like research or completing other time-consuming tasks much more efficiently.

The researchers also felt that the current methods of programming these agents, with manual code, would not enable the mass of users to use this technology, and they wanted to see if it was possible to implement a user interface for this task.

Flexible conversation patterns

How we communicate depends on the situation. The researchers wanted to test this with AutoGen. They proposed six different methods of communication, each focused on solving a specific problem. This was done to show the flexibility of the AutoGen framework when working with other applications. The methods described here are the ones relevant to our research. All methods, along with the scenario for each method, is documented in Appendix D in the research paper by Wu et.al [88]

Decision-making in text world environments This is the third method discussed by the researchers. The idea was to see how AutoGen would work in a text-based environment where the agents would solve day-to-day tasks using the ALFWorld environment [89] as an evaluation set. ALFWorld is a simulated textual environment where the large language models are tested in understanding text-based logic and was created by researchers Shridhar et al. [89] The researchers of AutoGen based the method on ReAct [47] to have the ability for the model to provide Reasoning and Acting, along with a thought based interaction between the agents.

The method outperformed the native ALFChat [89] application used for testing the environment by 15 percent by using a Grounding agent that supplied common-sense knowledge to the other two agents in the design. This helped reduce the hallucinations and error loops on the two-agent solution used by ALFChat.

This kind of decision-making and communication is highly relevant to the research performed in ChatRange, as this is meant as a platform that needs to create fictive scenarios based on real-world events.

Simplified Framework

The second main focus of the research was creating a solution that would be simple to interact with for all kinds of personnel, especially for those who are not developers or do not use programming languages daily. The researchers have, therefore, released a web-based user interface named AutoGen Studio. This provides the ability to program all agents, tools, and workflows instead of interacting with the Python code [90].

The researchers made this possible by implementing three interfaces, 'send,' 'receive,' and 'generate-reply,' that standardized the agents' communication method. This enabled programming the agents' interactions into a user interface [88]. This method also allows the agent to answer back in a predefined way. This reduced the need for coding and oversight, usually needed when working directly with the code.

5.2.3 CrewAI

LangChain is one of the largest implementations of using large language models in Python, having existed since October 2022. The framework was designed to make it simple to implement extra functionality on top of existing language models, like interacting with documents, external tools, and Retrieval-Augmented Generation [48].

LangChain has an implementation where it is possible to program agents manually by using the ReAct [47] framework, as demonstrated by Wu et al. [88]. This implementation did, however, not provide correct results when measured against their implementation of AutoGen in a MATH-solving task. Developer and founder João Moura wanted to solve this when developing CrewAI [49], released in October 2023.

CrewAI is an implementation that builds on the existing LangChain framework but simplifies the creation of agents, processes, and available tools. CrewAI also implements ReAct directly into the framework, giving the agents the same abilities that Wu et al. [88] used in their research.

Quick code

Implementing CrewAI reduces the need for complex code. Each agent can be specified with a simple function where the goals and background are defined. This is where each agent gets its context for the task to be performed. In addition to this, each agent can also be customized with its large language model. If there is a need to develop code, an agent could be specified to only use OpenAI's Codex [87] model, which specializes in this area.

Large toolkit

Since CrewAI is built on LangChain, it also gains access to the toolkit LangChain provides. Many of these tools have been developed since the framework was released and include implementations like integrating with Azure AI Services and HuggingFace [91]. This allows for quick implementation of tools that would otherwise be time-consuming.

In addition, CrewAI also implements essential tools like Retrival-Augmented Generation and the ability to scrape web content with its toolkit, which is available to each agent to use directly [92].

Framework	**ChatDev** [45]	**AutoGen** [88]	**CrewAI** [49]
Arhcitecture	CAMEL [46]	AutoGen [88]	ReAct [47]
Speciality	Codeless	User Interface	Simplicity
Focus area	Software Development	Versatility	Fast implementation
Drawbacks	Must use GPT 3.5.	Requires Docker	Requires LangChain

Table 5.1: An overview of the features of each framework is discussed in this chapter, along with the methodology, specialty, focus area, and drawbacks for each.

5.3 Overview

Each framework and method discussed here excels in its specialty field. An overview of performance results would only be valid for the specific model compared to the particular framework. Therefore, a summary has been composed to highlight each framework's specialty field and area of operation, as shown in Table 5.1.

5.4 Conclusion

In the years since ChatGPT [84] and GPT 3 were released, interest and research into agent-based systems have exploded. This is partly due to the limitations that earlier language models experienced when it came to creating believable content. Language models excel in the art of language, as the name suggests, but are missing key functionality, like being able to add external information from the real world. Some solutions like ChatGPT and Microsoft have partially solved this by adding function calls into the solutions, but these are limited to the single chat a user has with the system.

These missing features are by design. When developing language models, they are trained on large corpora of data, and this training data does not need to be the most updated, just the most versatile, depending on the specific model's use.

5.4.1 Summary

This chapter has examined three frameworks widely used for implementing agent systems. Each system has its drawbacks and specialty areas. The goal is to select the most usable framework for developing ChatRange.

ChatDev [45] is specialized in reducing software development costs in software production. The agent infrastructure is based on CAMEL [46] but is standardized using GPT 3.5 for its large language model. This means this solution effectively leads to lock-in for language models and negates the possibility of bench-marking with multiple models.

AutoGen [88] is the most versatile framework that implements a solution for developing agents without coding. Using AutoGen Studio, each agent, tool, and overall flow can be programmed into the system. The implementation does, however, require running the agents in Docker. This requirement makes our solution more complex to implement for companies wanting to develop cyber exercises. Implementing AutoGen companies would require security hardening and technical resources, more than just setting config settings.

CrewAI [49] is the simplest of the three solutions, but implementing it requires some coding skills, as the framework builds on LangChain [48]. The framework can use multiple large language models in the same agent teams, allowing open-source models to be used in the tasks.

5.4.2 Optimal solution

Of the three frameworks, CrewAI is the most optimal option for this research project. Its fast implementation using LangChain means that the system can be implemented quickly into ChatRange. The framework also builds on the popular ReAct framework [47], which enables reasoning and thought into the agent logic. This will help with the reasoning tasks when developing the cyber training data.

Using multiple LLMs also allows the system to be tested against open-source and private models. This is an essential feature of our platform, as it is designed to work out of the box for companies wanting to develop cyber exercises by using OpenAI's cloud delivery model or Azure AI. In that instance, all that is needed is an API key. The system can also easily be configured for users or companies wanting to run their models in CrewAI, which is natively supported.

Chapter 6

ChatRange

This chapter is dedicated to the proposed design of ChatRange and how the finished solution is implemented. It will cover how agents are implemented into the creation process for cyber security training scenarios. This also includes how agents can be combined with prompt engineering to get the best results. The following research questions are answered here:

- RQ3: How can autonomous AI agents be used alongside large language models to develop cyber security training scenarios?

6.1 Integrated design

Large language models are great for generating content and understanding language; however, as seen in the previous chapters, they have several possible but complicated problems to solve. Among these problems are hallucinations and going outside of the context provided. Generative agents can help with this, and this is very much needed for the solution proposed in this research project.

When developing cyber training scenarios, some critical issues need to be solved:

- The scenario must be relevant to the personnel performing the training. Providing a highly technical scenario to IT management would probably not yield the learning objectives that the audience wants.
- The learning objectives must be relevant to the participants.
- The threat actor must be accurate, including tactics that the threat actor commonly uses. This is important as the exercise management might want to use real-world search engines and databases to collect and process information used in the scenario to heighten realism.

This requires the system's design to conform to interactions in real-world information databases like Wikipedia, news sources, and search engines. The design must

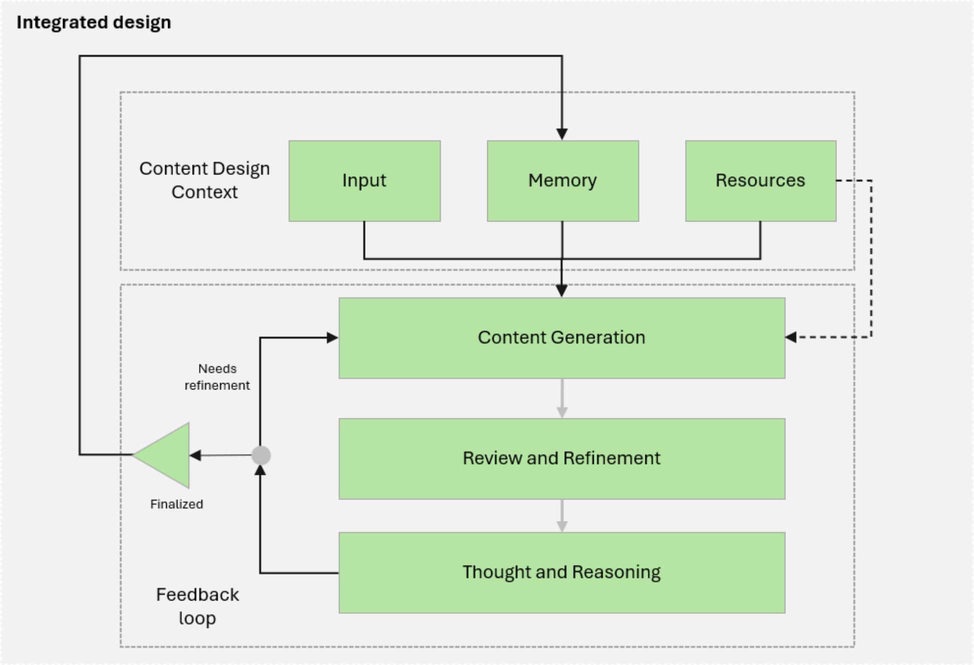


Figure 6.1: The integrated design shows how information is used in the development process of the training scenarios. The information is run through a dual feedback loop and then used for context in the next part of the scenario development process.

also be able to refine the information provided to the system. This critical context is essential for the total scenario generation. The complete design is shown in Figure 6.1.

6.1.1 Content Design Context

The first part of the design defines what information the system needs to maintain the context. This helps with the problem of large language models and hallucinations. The three parts described here comprise the total content of information that is used further in information processing.

Input

This is the context we provide to the system; typically, this is static information, like the start and end times of the exercise. This type of information gives essential context to the environment around the scenario and helps the large language model align with the objectives without going outside of the scope.

Memory

Memory is an essential part of the input. It is information about what happened earlier in the process. A database or some other form of storage can solve this practically.

Resources

This is the final part of the primary context. These external or internal resources can be added using tools or other methods. This is to provide additional content outside of the corpus of data on which each language model is trained, as this information often has a stop date for new knowledge it is trained on. GPT 4, for instance, has its stop date on April 2023 [64], and the model does not have any new information about current events after this date.

6.1.2 Feedback loop with thought instructions

The second part of the design is the content development. This is where the language models process all the information inputs using an agent-based infrastructure or prompt engineering with few-shots. The design is based on a feedback loop where the output of the thought and reasoning part is inputted to the content generation until the system deems the information to be finalized. This helps maintain the conversation context using the large language model, as we only provide the relevant information as the input context.

Content generation

Is responsible for querying with the provided design context against the language model based on the given task. If the content generation is in a feedback loop state, it will also use the Thought and reasoning [47] feedback to enhance the context for the large language model.

Content generation can also access external resources like those in the design context.

Review and refinement

The content is then reviewed to see if it meets the task specification given to the system. This review will contain information on what is good and does not meet the given task specification. This can be considered an edit for a movie that has not been published yet. The producers still have time to correct mistakes.

Thought and Reasoning

The most important part of the feedback loop is where the human element used in ReAct [47] comes into play. The system is given the generated content and the

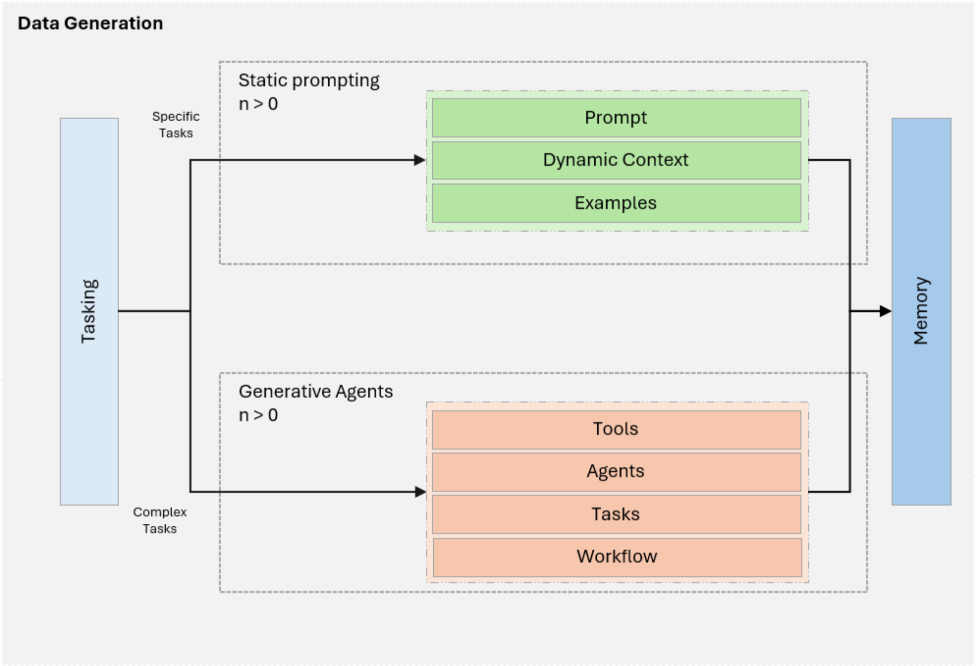


Figure 6.2: The content generation shows how the system design can either use static prompting or generative agents depending on the complexity and type of task of the task.

review and then asked to provide thoughts and reasoning for the next steps in the process. If the content meets the criteria defined in the task specification, the content is released and saved into memory for further use. If not, the feedback loop kicks in, and the thought process is released to the content generation.

6.2 Data generation

The system uses a two-step method for generating data. This further reduces the hallucinations experienced during earlier designs. What was often experienced was the system trying to go outside the boundaries set by the context when provided with large context windows above 30,000 tokens.

6.2.1 Few-shot prompting with dynamic content

Few-shot prompting is implemented by using static prompts with examples. This is then enhanced with dynamic information from the previous steps in the design. This information is combined into a single prompt presented to the large language model. This means that there are some restrictions on the selection of models. The context window must have at least 32,000 tokens, and the model must quickly process the information.

This requirement reduces the number of available models, meaning this method is only used in specific instances. In this setting, the Generative agents often get confused due to the large amount of information provided. One example is when the agents are queried to generate questions for the system; they frequently get the questions wrong or decide to define only a few that do not meet the exercise objectives. This challenge and the proposed solution are further explained in section 6.3.5.

6.2.2 Generative Agents

CrewAI [49] was deemed the most fitting framework for generative agents in this research project, as explained in Chapter 5. This framework consists of several predefined components that can be implemented into solutions like ChatRange, where dynamic research of external information outside the knowledge scope of the large language models is needed. This section will not go into depth about the specific framework but will show how it is planned and implemented into ChatRange.

Agents

Agents are "virtual" users in the system. They operate as normal human beings, querying the language models. Each agent is defined with a goal, and what is essential for that agent is defined. They also have backgrounds describing their feelings, thoughts, and experiences. Together, this forms a context that can be used when querying the models. To enable the automatic generation of Agents in ChatRange, the Goals and Backstory are generated by GPT-3.5-Turbo [84], based on the name of the role.

Agents have the possibility of working together or alone, and by implementing Reason and Acting [47] along with AGI [81], the agents are given new functionality to be able to remember earlier interactions along with the ability to provide reasoning for the tasks that needs to be completed. In addition, it is also possible to specify the knowledge of each agent. This is done by defining what language model the agent can use. With this, each agent can be customized to the specific need. For example, a developer could e.g. be described using OpenAI's Codex LLM [87], which specializes in Coding languages.

A complete list of all the agents used in ChatRange can be found in Appendix A.3.

Goals Each agent in the system is defined with a goal. In Figure 6.3, one of the agents used in the system, the Threat Hunter, is described. The goal is the individual objective that the agents want to achieve. This is to help the agent in the decision-making process [49].

In ChatRange, the Threat Hunter's goal is defined as: "*Identify and neutralize advanced persistent threats (APTs) within complex digital environments, employing*

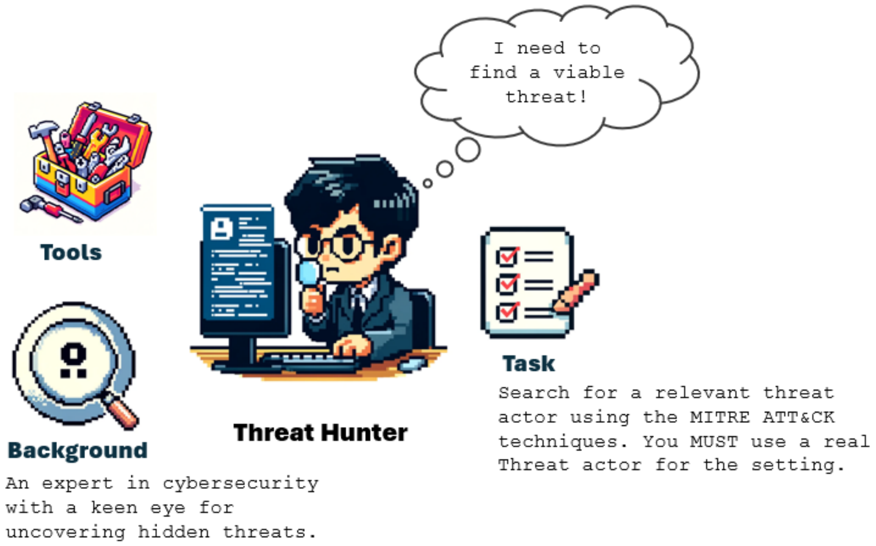

I need to find a viable threat!
Tools
Threat Hunter
Task
Search for a relevant threat actor using the MITRE ATT&CK techniques. You MUST use a real Threat actor for the setting.
Background
An expert in cybersecurity with a keen eye for uncovering hidden threats.

Figure 6.3: Shows an example of a single agent in the system. The Threat Hunter, as defined in ChatRange, has its goal, background, and task defined.

cutting-edge technology and methodologies to detect, analyze, and disarm sophisticated cyber threats before they can exploit vulnerabilities." [84]

This goal is created for this agent's specific tasks, primarily related to finding and parsing relevant threat intelligence for exercise scenarios and planning.

Backstory In the same way that agents have specified specific goals, they also have a backstory. This helps the agents align with each other when having conversations and helps each agent assign the correct task to the proper agent. As shown in Figure 6.3, the Threat hunter's background is focused on investigative traits.

The whole backstory for this role in ChatRange is: "*An expert in cybersecurity with a keen eye for uncovering hidden threats, the Threat Hunter has a decorated history of neutralizing high-profile cyberattacks. With a background in digital forensics and a passion for cybersecurity innovation, they have become a pivotal asset in preemptive threat detection and resolution, safeguarding critical infrastructure from potential cyber disasters*." [84]

Tools

Access to external tools is essential when performing tasks. These will help the agents provide the best answers possible, as they would in the real world. Tools can have various functionalities, including external access to the internet and particular access databases. They can also be integrated with custom APIs.

This is a short description of the tools available to ChatRange agents. A complete list of all the tools can be found in Appendix A.5.

Search engines Access to relevant and updated information is essential when performing research tasks. The use of ChatRange is equally, if not more, important. Having realistic scenarios developed around the most up-to-date sources makes it possible to create the best scenarios.

Access to search engines is part of this process, allowing agents to rapidly query for information about specific topics of interest. The search engine returns a comprehensive set of web pages, metadata for each, and a content summary. This allows the agents to summarize the returned content and pick the relevant sources for further queries.

In addition, search engines allow access to news sources and media repositories, such as image search. This will help further when performing case studies in ChatRange.

Search engines available in ChatRange include:

- Google Search
- Bing
- Wolfram Alpha

Web-pages Web pages provide access to in-depth content about specific topics. While search engines only give an overview of a set of web pages related to a search, the particular web page might contain more valuable information to the research process.

ChatRange can visit web pages by scraping their information and parsing it through the large language model. This is one of the primary reasons the language model needs to function with a significant context window, as this content can be very long.

Documents The last category of tools highlighted here is access to documents. These are specific sources of information that enhance agents' performance in their specialty field. This allows for future implementation, where documents can be submitted to the system to generate exercises based on their content.

Tasks

The vital part of the agent framework is the tasks. These are human-generated inputs to what the agents are trying to solve. These must not be mixed with the context the agents use to solve the task, as the agents themselves generate these.

As described above, one of the tasks related to the threat hunter role is:

Task: Search for a relevant threat actor using the MITRE ATT&CK techniques. You MUST use a real Threat actor for the setting.

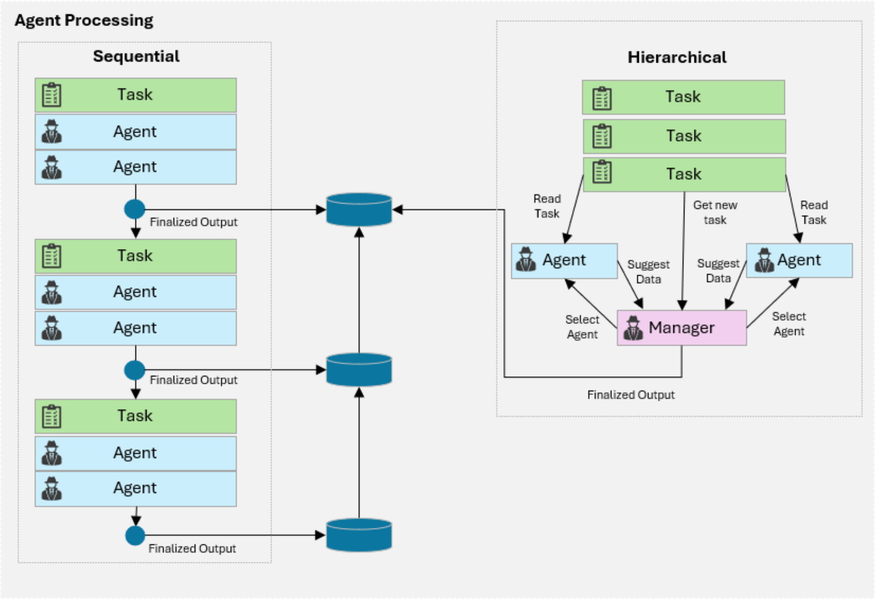


Figure 6.4: Shows a overview of the different processing mechanisms in CrewAI [49]. The Sequential processing mechanism is a step by step task runner. The Hierarchical provides more flexibility as it is up to the Manager to selected tasks for the right agents.

All tasks are also supplied with an expected outcome, which helps the agents align their answers to meet the user's expectations. In the case of this task, the expected outcome is defined as:

```
A report of the threat actor markdown.
It must contain the following:
## Threat actor name
## Description
## Modus operandi
## Associated groups
## MITRE ATT&CK Techniques Used
```

The agents are told to present the content as markdown in the specified format. This format is chosen because language models are often better at following the brief when creating a textual format than a structured format like JSON. However, using a Pydantic model to force a structured format is possible. Another point is that it is simpler to feed the content back to the language model as context without having to parse it.

Process

The last part of the framework defines how the tasks will be solved. Each task is presented to the agent in a sequential form. There are, however, two forms of delegation of tasks:

Sequential In sequential mode, the tasks are handled after each other with a predefined set of agents for each task. This process is the most straightforward but least flexible, as shown in Figure 6.4. This kind of execution is the best when conducting specific research where each task must be solved in the correct order.

Hierarchical In Hierarchical mode, CrewAI [49] implements a manager feature. This is an agent predefined by the system responsible for completing the tasks as it sees best fit. The manager chooses which agent is given which task based on the agent's background information and goals, as shown in Figure 6.4.

The user then specifies a crew or a team. Here are the agents in the team I defined, along with what task this team is supposed to solve. The manager can delegate a task to a team member asynchronously or sequentially. This means that the Hierarchical mode can perform tasks with multiple agents at the same time.

6.3 Exercise development

This section will provide an overview of how exercises are developed using the ChatRange platform and the different methodologies. Most of the steps can be fully automated; however, for demonstration purposes, the whole solution is developed as a step-by-step process.

6.3.1 Exercise context

The first step is to define the static input to the system. This is used throughout the process of developing the exercises, depending on the specific step.

Exercise type

The solution was developed specifically for creating tabletop exercises; however, the system is designed to be expanded to create hybrid and cyber-range exercises in the future. This field is statically set to table-top.

Start and End Time

The start and end times are used to calculate the exercise length.

Participants

Participants are the different profiles of people participating in the exercises. This is important as the whole exercise is designed around the participant types. If, for instance, the exercise is for IT management, the questions and objectives generated will be adjusted to fit this group.

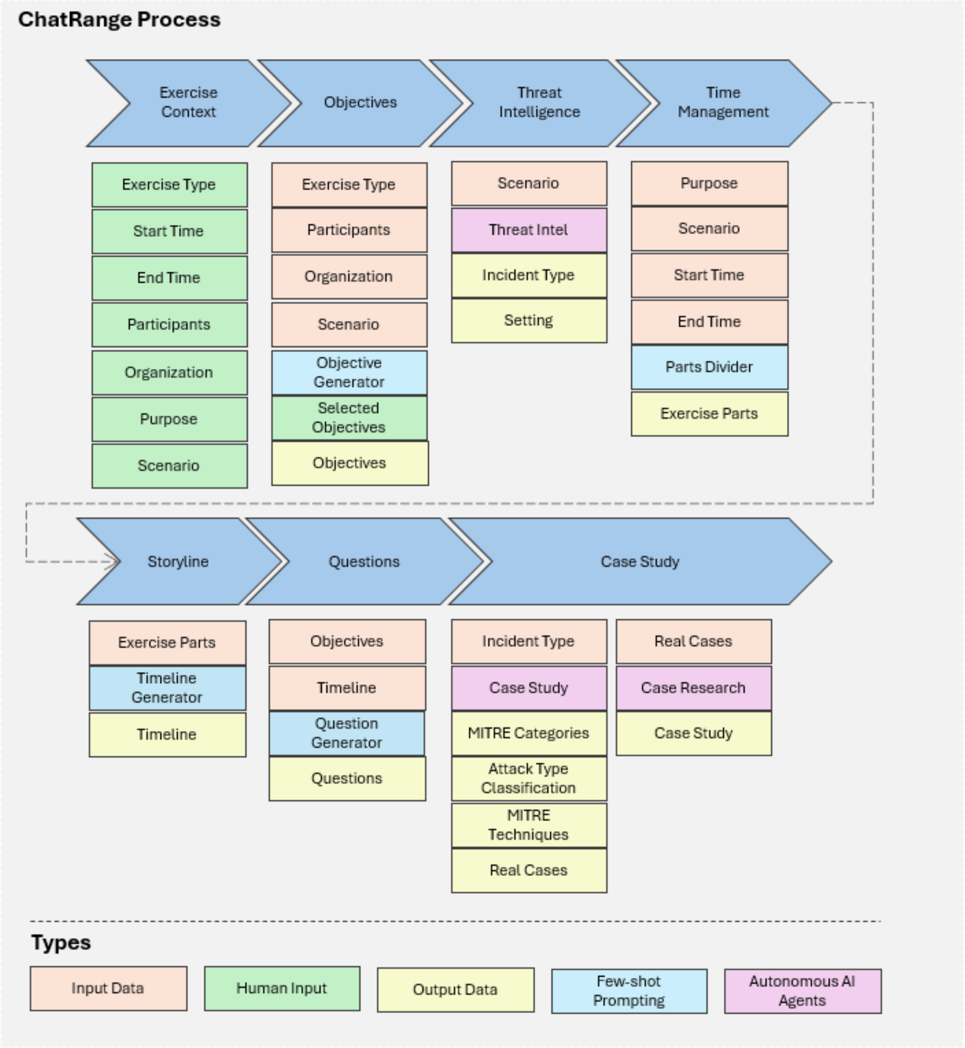


Figure 6.5: This figure provides a complete overview of ChatRange's functionality. It also includes the type of data used for each specific process in developing cyber training scenarios for the solution.

Organization

This is the name of the organization performing the training. This input is used to tailor the exercise realism by implementing the organization name in the objectives and timeline.

Purpose

The purpose must be set to tailor the exercises to specific needs. This details the overreaching goals of conducting the exercises. Later, this is combined with the Scenario to determine the length of the exercises.

Scenario

This is a brief overview of the background for the exercises. This helps the system align the story and objectives suggestions used later.

6.3.2 Objectives

The objectives are the overreaching goals of conducting the exercise. This feature enables the user to select precisely what the participants should train on. The system suggests objectives based on the exercise type, participants, organization, and scenario, creating a customized experience.

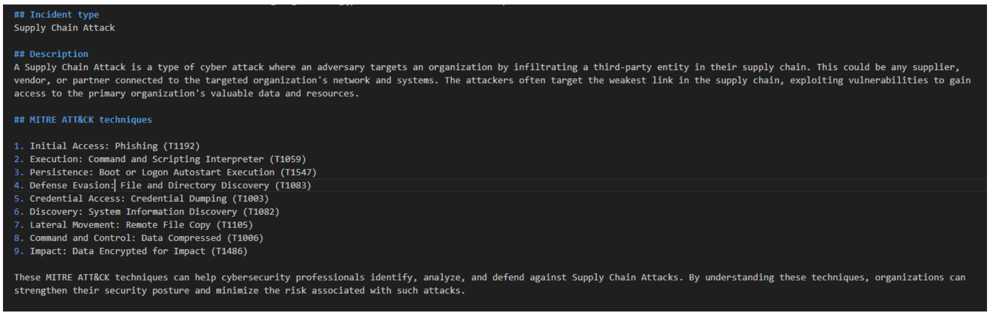
The system uses Few-Shot prompting with role-playing, where the role is defined as the organization's CISO, to suggest objectives. Five objectives are suggested each time. This is used as it was the fastest solution for creating multiple objectives at once.

6.3.3 Threat Intelligence

A real threat actor is needed to develop a realistic training exercise. This allows participants and trainers to research the specific threat actor's tactics before the exercise starts if preparations are required and set the training scenario in a realistic setting. Threat intelligence is analyzed using Autonomous AI Agents and external tools in several steps. The scenario, as defined in the Exercise context, is the input to the threat intelligence process.

Agents

Two agents are defined for these tasks and can collaborate where they need specialist expertise. These are the threat hunter as described inA.3.1, and the cybersecurity researcher as described in A.3.2.



```
## Incident type
Supply Chain Attack

## Description
A Supply Chain Attack is a type of cyber attack where an adversary targets an organization by infiltrating a third-party entity in their supply chain. This could be any supplier,
vendor, or partner connected to the targeted organization's network and systems. The attackers often target the weakest link in the supply chain, exploiting vulnerabilities to gain
access to the primary organization's valuable data and resources.

## MITRE ATT&CK techniques

1. Initial Access: Phishing (T1192)
2. Execution: Command and Scripting Interpreter (T1059)
3. Persistence: Boot or Logon Autostart Execution (T1547)
4. Defense Evasion: File and Directory Discovery (T1083)
5. Credential Access: Credential Dumping (T1003)
6. Discovery: System Information Discovery (T1082)
7. Lateral Movement: Remote File Copy (T1105)
8. Command and Control: Data Compressed (T1006)
9. Impact: Data Encrypted for Impact (T1486)

These MITRE ATT&CK techniques can help cybersecurity professionals identify, analyze, and defend against Supply Chain Attacks. By understanding these techniques, organizations can
strengthen their security posture and minimize the risk associated with such attacks.
```

Figure 6.6: This is an example of how the threat intelligence team of agents can research an incident type for the training exercise. Here, the scenario focuses on a third-party infection that leads to a ransomware attack on the target company.

Incident Type

The first output the agents create is the Incident type as shown in figure 6.6. This is a summary of the incident type best suited to the scenario. This report also includes a description of the incident, highlighting the tactic and how it is used. In the end, the MITRE ATT&Ck Techniques are defined. This is important as these techniques are used later to find a specific threat actor and used for the case studies.

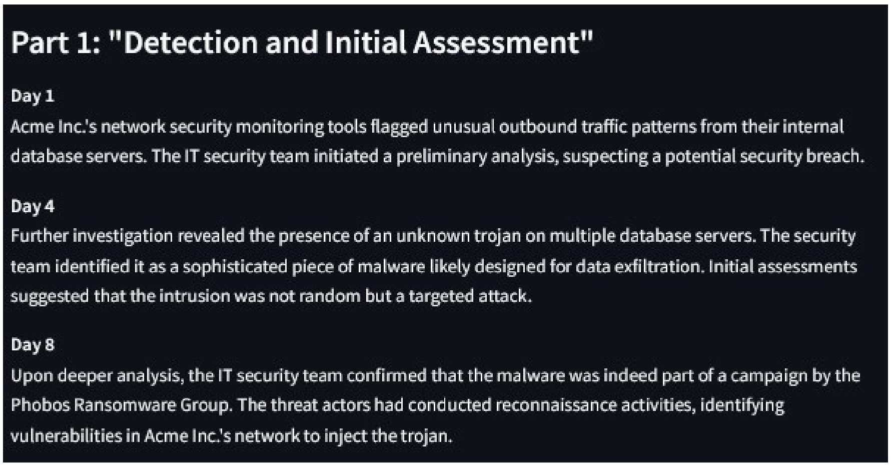
Setting

When the agents finish the incident type, they use this context and start on the Setting. This is where the specific Threat Actor is researched. The system uses the incident type as a baseline for what kind of threat actor should be used and selects one that uses this attack method. This is to heighten realism for the participants.

The setting contains a short description of the selected threat actor, including other names that can be used. It then details the modus operandi, or how the group operates, what it targets, etc. This is included if the group is associated with other known threat actors. Finally, the MITRE ATT&CK techniques are added.

6.3.4 Time management

The first thing ChatRange does is look at how many parts it should be divided into logically. It uses the Start and End time of the exercise, along with the Purpose and Scenario. ChatRange is asked to split the exercise into parts based on the cyber kill chain [93]. The system also adds a description for each part, on what it is asked to focus on, creating a summary of what this part is supposed to contain. This helps the large language model concentrate on specific steps a threat actor takes for the exercise, which will be used later to focus on the storyline and questions.



Part 1: "Detection and Initial Assessment"

Day 1

Acme Inc.'s network security monitoring tools flagged unusual outbound traffic patterns from their internal database servers. The IT security team initiated a preliminary analysis, suspecting a potential security breach.

Day 4

Further investigation revealed the presence of an unknown trojan on multiple database servers. The security team identified it as a sophisticated piece of malware likely designed for data exfiltration. Initial assessments suggested that the intrusion was not random but a targeted attack.

Day 8

Upon deeper analysis, the IT security team confirmed that the malware was indeed part of a campaign by the Phobos Ransomware Group. The threat actors had conducted reconnaissance activities, identifying vulnerabilities in Acme Inc.'s network to inject the trojan.

Figure 6.7: The storyline provides an overview of the events leading up to the training scenario in the exercise, which helps increase the exercise's realism.

6.3.5 Storyline

The storyline is designed to help the participants get into the game faster. It tells the story of the events leading up to the part of the exercise when the training starts. This is a chronological story, where events take place over several days. It is also defined as the timeline in the exercise, as shown in Figure 6.7.

6.3.6 Questions

The questions are the main contribution to the exercise. These are the questions that the participants are given when running the exercise and are used for the table-top discussions. The types of questions are specifically adapted to the training participants.

6.3.7 Case Studies

The last part of the exercise presents relevant information that can be used. These are case studies from real-world events related to the Techniques presented in the exercise. The participants are to use these to show how real-life cyber incidents impact real organizations using the same techniques they are training on.

The system is handled using two-agent teams: a case study team described in A.4.2 and a case research team described in A.4.3.

6.4 Conclusion

ChatRange is a solution developed to replicate a group developing cyber training scenarios. The proposed system integrates advanced technology, such as Autonomous AI Agents, into traditional static prompting and role-playing methods.

6.4.1 Context

ChatRange introduces the use of content design context, where the state of each part in the generation is saved and used for context in one of the next steps in the process. This helps the system align with the objectives of that particular task. ChatRange uses static prompting with role-playing and autonomous generative agents to generate the data. Combining these methods generates the most realistic content while also allowing the implementation of external data input for use in the generation process.

Using CrewAI [49] as the main framework for generative agents allows using Tools from the LangChain [48] framework. These tools are freely available to agents in ChatRange. This implementation enables each agent to use external information sources when researching and developing content, which is designed to help increase the realism participants will experience when running the exercises.

6.4.2 Exercise development

Dividing the exercise development into several stages enables various technologies to be used. This helps the system produce data that aligns with the exercise's objectives, increasing the solution's effectiveness. This architecture also allows ChatRange to utilize different language models for specific tasks. This enables the system to use the expertise of specialized models. It also enables the implementation of foundational language models for content generation, like Mixtral 8x7B [62].

One of the critical aspects of ChatRange is the ability to use external information when researching case studies. These real-world cyber incidents are compiled and presented to the exercise participants to help them understand the context of the training scenario. For example, training on a ransomware scenario could lead to less realistic exercises if the participants do not trust that the scenario could happen in real life.

ChatRange's ability to research this on the spot allows it to customize each training scenario for the specific use. This is especially handy when training on new techniques or using new threats. By using ChatRange, these training scenarios could be developed in minutes, ready to be used.

Chapter 7

Validation

This chapter is dedicated to evaluating and validating ChatRange's results. It measures ChatRange's time usage and quality against a human-developed control. The chapter starts by giving an overview of the methodology used in this measurement and ends with the testing results. This lays the foundation for the overall conclusion provided in the next chapter.

The following research question is answered here:

- RQ4: How do agent-based cyber training scenarios' details, realism, creativity, and efficiency measure against traditional manual approaches?

ChatRange produces data that can be used in tabletop exercises. A case study is done to validate and evaluate the system's performance. This Case study aims to validate if the system is effective in developing cyber training scenarios that can be utilized in exercises.

The case study is divided into two parts, each dedicated to evaluating a use case scenario.

7.1 Testing Methodology

Two training playbooks are used for the case studies. These structured exercise documents include all the material needed to run the training.

7.1.1 CISA Tabletop Exercise Packages

The Cybersecurity and Infrastructure Security Agency (CISA) developed the main scenario used in both evaluations [27]. The research team then slightly modified it, removing the table of contents and other non-relevant data to reduce the size of the documents, making it more likely for each participant in the survey to read them.

Scenario setup

The scenario has a set of predefined settings, which are the same for both playbooks used in the case study. This is done purposefully to ensure the two playbooks have the same primary setting. If they were different from each other, the case study would be impossible to conduct, as participants could be biased toward different scenarios. For example, participants could score higher in APT scenarios than in phishing scenarios, as APT scenarios can be seen as more exciting.

The scenario depicts a situation where a third-party vendor is used as an entry point to compromise ACME Inc. via a phishing attack. This leads the threat actors to gain access to the target company's networks and systems. When the threat actors get access, they unleash attacks that cause computer latency and network access problems. Before they leave the system, they unleash a ransomware attack.

The scenario is complemented to conduct the exercise. The training scenario examines how ACME Inc.'s coordination, collaboration, information sharing, and response capabilities function in light of the proposed scenario.

In addition to this, a set of objectives is also defined. These are specific training goals that the exercise needs to achieve. The scenario has three objectives, as shown in Appendix C. These were developed by CISA [27]:

1. Discuss elements of Acme Inc.'s cybersecurity posture.
2. Examine Acme Inc.'s cybersecurity information-sharing procedures and mechanisms.
3. Examine Acme Inc.'s cyber incident response plans or playbooks.

Modules

Each scenario is developed with a set of modules. For this exercise scenario, two modules are defined. This is where the playbooks are split, where one playbook presents modules developed by humans, and the other uses ChatRange-developed modules. The complete description of the content of each module is depicted in chapter 6.3.4.

Case study and Attack descriptions

The last part of the exercise includes more information that participants can be given to heighten the realism of the training. This includes case studies from real-world cyber security events that use the tactics from the training modules depicted in the exercise. In addition, an explanation of the main category of attack types is also presented, along with links to additional resources. Each of these parts is generated by ChatRange and by humans.

Playbooks

The first playbook presented is the one developed by CISA and created by humans. The whole playbook, as given in the survey, can be found in Appendix C.2. ChatRange generated the content of the second playbook. The researchers of this project then put it together manually into the Word template. This playbook can be found in Appendix C.3. The last manual part of the process can be automated, but this is not important for the system's validity as it is more of a convenience functionality.

7.1.2 Case Studies

The case studies are conducted using a blind study setup. In the survey, the participants scored the playbook generated by humans (the control) and the playbook developed by ChatRange using the same criteria specified in Chapter 3.2.4. The participants were not told which playbook was created by which system in the survey.

After the survey is completed, the scores are calculated, and the difference in each scenario is presented.

Broad Survey

The first case study was distributed among personnel who usually conduct cyber training exercises, including IT professionals, IT managers, information security personnel, and operational cybersecurity personnel. They were given a link to the survey and asked to fill it out. This broader part of the study aims to measure the overall quality of the training scenario generated by ChatRange.

Expert reviews

The second case study is intended to provide more specialized feedback on the exercise. It uses a panel of expert reviewers who have long experience in different fields within IT management and IT security. The reviewers have worked in Norway's government, private, military, and public sectors. A complete overview of each participant's background can be found in Appendix D.

Each reviewer is asked to score each playbook as in the first case study and to review each playbook with a textual comment for each criterion defined in the survey. This helps to shed more light on what ChatRange does well and where it needs improvement. In addition to this, they are also asked for an overall comment for the whole exercise, what was good and what didn't work.

Category	Specific Titles	Count
Security	Security, IT Security, Business Security Officer	11
IT	IT, IT-Responsible, NOC Engineer, Student	8
IT Management	IT Management, CISO, IT Architecture	6

Table 7.1: This table provides an overview of the employment areas of the survey participants. It shows a generalized category and an overview of each participant's more specific titles.

7.2 Case Study Results

The following section is dedicated to the case study's results. The information here is presented in two subsections, each for the specific case study that was done. Each section contains a summary of the results for the case study along with a short summary of the scores for each of them.

After this section, each criterion is discussed in terms of the overall score, the textual reviews from the experts, and the overall results.

7.2.1 Broad study

The broad survey was conducted from April 19th, 2024, to May 6th, 2024. In total, 25 answers were given to the survey. A complete list of the research data for this case study can be found in Appendix B. A brief overview of the participants and their professions can be seen in the table 7.1. The participants are from all over the IT spectrum, with a focus on IT security.

Human Generated

The first playbook evaluated here, Playbook 1 (P1), was generated by CISA. The playbook scored 4.0867 across all six criteria, with minor changes for each. The overall change in maximum and minimum scores is 12.037 percent, and the scores are consistent across all measurements. The highest score is 4.32, and the lowest is 3.80, meaning all criteria scored from approximately 3.8 to 4.3. The total scores are shown in Figure 7.1. All scores are listed in Table 7.2.

ChatRange Generated

The second playbook evaluated, Playbook 2 (P2), was generated by the proposed solution ChatRange. The playbook scored 4.0800 across the six criteria, slightly less than P1. The variation between the scores is less than in P1, and the maximum and minimum scores see only a 4.808 percent change. The highest score is 4.16, and the lowest score is 3.96, meaning all criteria scored from approximately 3.9 to 4.2, similar to P1. The total scores are shown in Figure 7.1. All scores are listed in Table 7.2.

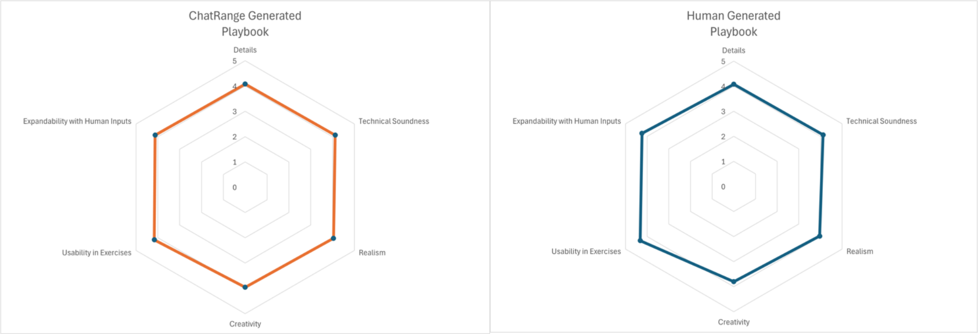


Figure 7.1: Broader study - The first playbook (P1), which was human-generated, scored High to Moderate in all criteria, the same as the second playbook (P2) from ChatRange, with only minor changes in each criterion.

7.2.2 Expert Evaluations

The expert evaluations were conducted from April 29th, 2024, to May 6th, 2024. Three reviewers answered the case study, giving detailed information on how well each playbook performed. This section shows each scenario's overall score given by the experts in the study. A complete list of all the answers each participant gave in the survey can be found in Appendix D.

Human Generated

The experts scored the first playbook with an overall score of 4.5, slightly higher than the overall score of the broader survey. The 7.142 percent changes from the highest to lowest score are lower than the results of the broader case study, showing fewer variations across the criteria scored. The lowest score is 4.33, and the highest is 5.0. This makes the range approximately 4.3 to 5, far higher than in the first case study.

ChatRange Generated

The experts scored the second playbook with the same average score as the human-generated playbook, at 4.50. The playbook (P2) saw a similar consistency in the answers as the broader survey, with a score of 7.142 percent, the same as the expert scores of the human-generated playbook. The lowest score was 4.33, and the highest was 4.67, which puts the range approximately from 3.4 to 4.7, slightly lower than P1 but higher than the broader study.

7.2.3 Discussions

This section examines the scores in both case studies, including expert reviews of the scenarios and the criteria.

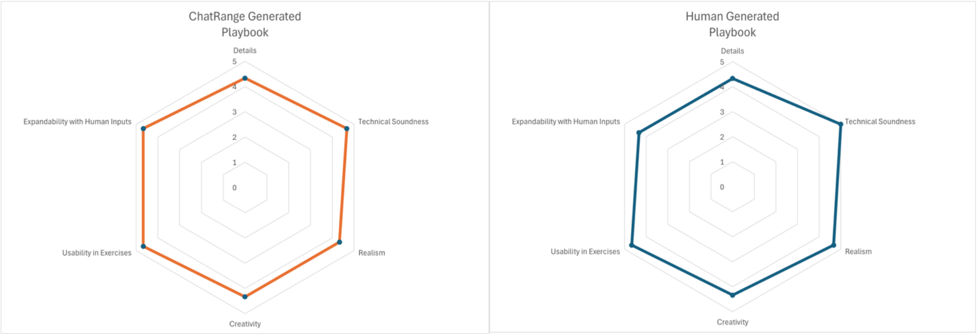


Figure 7.2: Expert reviews - The first playbook (P1), which was human-generated, scored High to Very High in all criteria, the same as the second playbook (P2) from ChatRange, with only minor changes in each criterion.

Data quality

The overall quality scores for both case studies are in the higher range, whereas the second case study using the experts scored both scenarios higher than the broader survey. This is normal, as the broader study had more participants than the three experts in the second study. This leads to a challenge when averaging the scores, as the input data consists of a sample pool with too little data. That challenge makes it hard to analyze the sample pool as the data could be scored too similarly and not provide the average score [94, p. 181-184]. This is the likely scenario that has happened with the expert reviews and why these scores are higher than in the broader survey.

Based on the challenge of the principle of the Law of large numbers [94, p. 181-184], the expert's surveys have been included in the broader survey's statistics, giving a more realistic view of the score. These scores are included in Figure 7.1 and Table 7.2.

Comparing the playbooks

Playbook 1 (p1), generated by CISA, is quite similar to the average total score of the scenario developed by ChatRange (P2). Compared to P1, P2 only sees a microscopic 0.067 score decrease equaling 0.16 percent.

Looking more into the specific criteria measured, there are a bit more variances between the two playbooks, as seen in Table 7.2 and Figure 7.3.

Playbook	Details	Tech.	Realism	Creativity	Usab.	Exbandab.
P1	4,08	4,12	3,96	3,80	4,32	4,24
P2	4,08	4,12	4,04	3,96	4,16	4,12

Table 7.2: The table displays the average score of each playbook for each criterion from the first broader case study, including the scores from the expert reviewers.

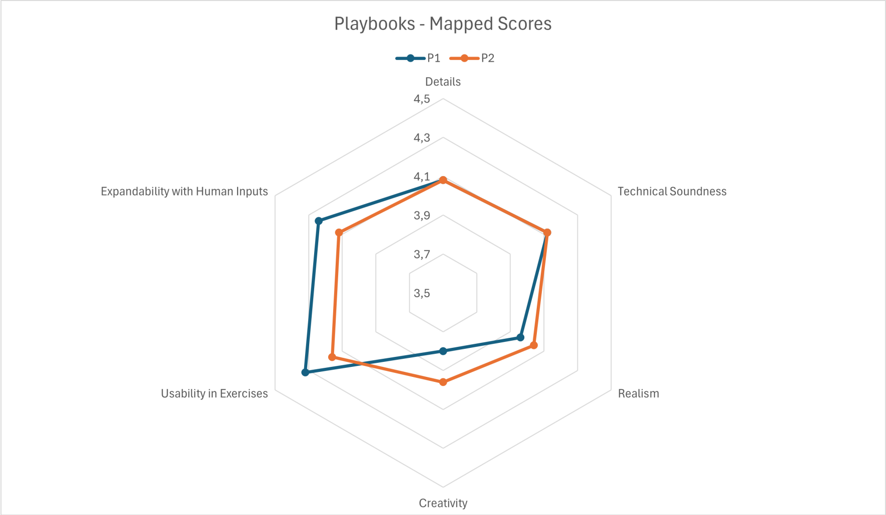


Figure 7.3: The second playbook (P2) scored overall better in two of the criteria, and equal for two other criteria, in the survey, while P1 scored better in the last two.

The reviewers are named as:

- Expert 1 [D.2]
- Expert 2 [D.3]
- Expert 3 [D.4]

Each criterion is evaluated separately using the quality score from the first case study and the experts' reviews. ChatRange scored better in four out of the six criteria, as seen in Figure 7.5.

1. **Details**: 4.08 High (P1) / 4.08 High (P2)
   The content strongly represents real-world settings, and the threats are well-elaborated. The scenario details phishing and ransomware, current threats perceived as real-world problems. Expert 1 specifically points out that adding appendixes with real-world security incidents helps heighten the realism and use it in exercises. The scenario and discussion points were also presented in a detailed way that made it easy to understand, as detailed by Expert 3.

   These experts point out that the CISA-generated (P1) and ChatRange-generated (P2) playbooks have great details. However, some drawbacks are also expressly noted with ChatRange. Expert 2 notes that in the modules, the first module seems more drawn to technical personnel and notes that in this part of the exercise, the reviewer would have to use them to understand how to answer the questions. This is also partially covered by Expert 3, who

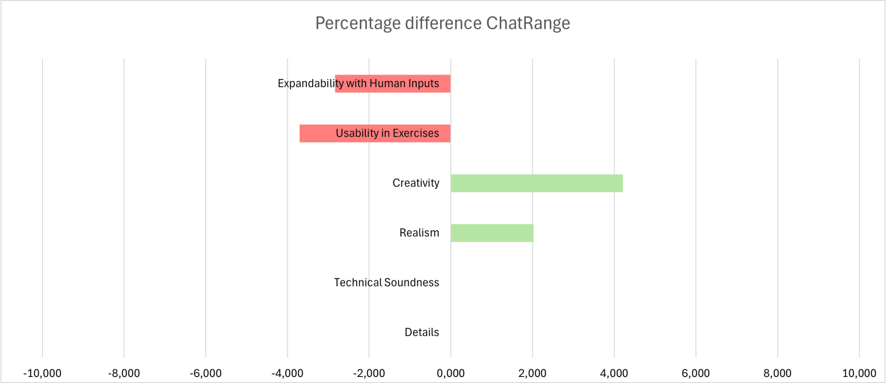


Figure 7.4: ChatRange scored equal or better in four out of the six categories evaluated. The graphs show the percentage difference in scores across the criteria. The red score is where ChatRange scored worse than the manual playbook, and green is where it scored better.

suggests the exercise targets align more with IT technical personnel and IT managers. This is a challenge with using large language models as these can go outside the given context, which in this case was to generate an exercise for IT Managers.

2. **Technical Soundness**: 4.12 High (P1) / 4.12 High (P2)
   The current technologies in the exercise fully align with current technology limitations and respect cyber core cybersecurity principles. Both playbooks scored the same in the broader survey, and the reviewers had no negative comments on this criterion, except for minor suggestions, as explained above in Details.

   Both playbooks scored highly here, suggesting they did not exceed the technical limitations in the real world. This could have happened with the language models inventing new ways to conduct cyber attacks or using futuristic or science fiction techniques to explain the scenario.

3. **Realism**: 3,96 Moderate (P1) / 4.04 High (P2)
   Both playbooks deliver reasonably high realism in the scenarios presented. Both include real-world cyber threats, where the threat actors use realistic techniques and procedures. The difference in score is negligible, with P2 scoring 0.08 points, or 2 percent higher.

   A portion that gained some critique from Expert 1 and Expert 3 is using the wrong TLP definition in both scenarios. CISA manually generated this part in both playbooks and does not account for variances in system use. However, Expert 1 also points out that the ChatRange-developed scenario seems repetitive, which could negatively affect training exercises. Expert 2 points out that P2 is realistic and aligns with standard practices.

4. **Creativity**: 3.8 Moderate (P1) / 3.96 Moderate (P2)
   The playbooks score moderate for Realism. P1 scores 3.8, with P2 scoring 0.16 points, or 4.124 percent higher than P1. This equates to the ability to implement diverse scenarios that require creative thinking to be completed. P2 scores higher than P1 in this instance, which can come down to implementing autonomous research agents. These can implement current real-world security events, enabling them to provide scenarios related to the latest news and trends. These would feel more familiar to the participants than a scenario where the time from development starts to the actual exercise, which could be from months to years, and therefore have to be more generalized.

   Expert 1 points out that the appendices in P2 are closely related to the specific training modules presented in the playbook. This is where the power of the autonomous research agents is shown, as they are able to extract data from several sources and compile them into specific case studies related to each part of the training.

5. **Usability in Exercises**: 4.32 High (P1) / 4.16 High (P2)
   Both playbooks score high for usability in exercises. This is related to how well the exercises can be used to create new training scenarios or in more extensive training sessions with hybrid or complete cyber-range training scenarios. P1 scores 4.32 points, which is 0.16 points, or 3.77 percent, higher than P2.
   This is unsurprising, as P1 is intended for use in a more generalized setting. The scenario is tailored to any Organization and setting and can not contain too many details. For example, where P2 actively names Acme Inc. in the storyline for the modules, P1 uses generalized language. While this enables the exercises to be used in multiple settings, it can also reduce realism, as the setting is further away from the participants.

   One example is the ability to have the training scenario tailored to specific languages, e.g. as Expert 2 mentioned, having the exercise in Norwegian. For a generalized scenario, this would involve a translation after the scenario is created, which often leads to bad translations or manual work. Using systems like ChatRange, the exercise can be developed using specific language models tailored to the desired output language and even localized resources for the research agents, helping improve localized content, e.g., using particular Norwegian cyber incidents as case studies.

6. **Expandability with Human Inputs**: 4.24 High (P1) / 4.12 High (P2)
   The last criterion looks at how easy it is to incorporate new information or

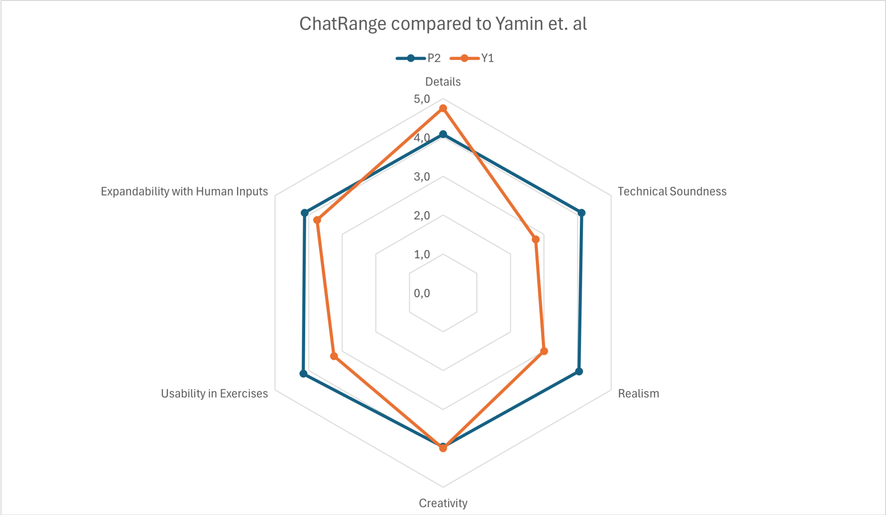


Figure 7.5: The expert's reviewers' results of ChatRange compared to the average scores of scenario 1 and scenario 2 in their research. [50].

real-world events into the scenario. Both playbooks score High here, having only a tiny 0.12 points, or a 2.871 percent difference in favor of P1.

The reviewers have no direct comments regarding this criteria, but the P2 playbook scores slightly lower than P1. This could be attributed to the fact that P2 is less generalized and, therefore, less able to be modified by human input to develop the exercise. This is, however, not a significant problem for ChatRange as it is designed to tailor the specific exercises to a given scenario, and by design, it should not be generalized.

Related research

The criteria used in this research paper have also been tested in other research papers. Yamin et al. [50] investigated the possibility of implementing autonomous AI agents to generate scenarios for the Norwegian Cyber range. The researchers generated data for two scenarios, which were then scored using the same quality criteria presented in this paper, using two expert reviewers.

Overall, both scenarios scored three or better for all criteria when evaluated by the experts. The overall scores from both experts are similar to those provided in this study, as seen in Figure 7.5. Here, the average score of both scenario 1 and scenario 2 presented by Yamin et al. is compared with the expert reviewers' scores from ChatRange. This shows only minor differences across the scores with some scores. It is worth noting that the same score in creativity was achieved across both studies, as this was one of the highlights pointed out by the expert reviewers of ChatRange.

7.3 Efficiency Measurements

Efficiency is measured by using the cost of developing the exercises. Three metrics must be set to define the cost:

- Time to create exercises in hours
- Cost per work-hour
- Additional costs

This measurement is used in this thesis as it equates to the cost of man-hours spent by corporations wanting to use systems like ChatRange, in contrast to developing exercises manually.

This is defined as: $(time * cost\ per\ hour) + additional\ costs = total\ cost$

7.3.1 Time measurements

The time is measured in hours and is used for both the generation of manual exercises and ChatRange.

Manual Exercises

For the manual training scenario, the time is estimated based on two sources: MITRE [25] and NIST SP 800-84 [19]. These are organizations with a long-standing reputation in the cyber security field.

MITRE MITRE Corporation is a private non-profit organization focusing on multiple domains, including cybersecurity. In 2014, Mitre released the Cyber Exercise handbook [25], which details steps to plan and execute cyber training exercises, including tabletop, hybrid, and complete cyber range training scenarios. They estimate that one to two months of planning time is needed for a one—to three-day exercise.

Estimated time: **1-2 months**

NIST The National Institute of Standards and Technology promotes security standards within the United States. One of its publications, NIST SP 800-84 [19], is dedicated to guiding personnel in developing and testing exercise programs. The publication estimates that the planning process needs to start at least three months before running an eight-hour tabletop exercise.

Estimated time: **3 months**

Estimated time usage Based on MITRE and NIST calculations, this research defines the time required to generate cyber exercises manually as 1.5 months or **244 work hours**. Some time has been deducted for the planning as this is related to planning the infrastructure like rooms, participants etc. This is included in the number specified in Table 7.3

The CISA scenario (P1) was a premade scenario edited by the researchers. This took approximately **10 hours**, which could be considered the actual time it took to develop the scenario for the exercise. However, using this time will only work for that single instance. If a scenario needs to be created where the input changes, it must be made from scratch with no help or template. For total transparency, the calculation using this number is also presented in Table 7.3.

System	Work	Cost/Hour	Cost	Add. Cost	Total
Manual	244	327,07	79 805,08	0	79 805,08
ChatRange	0,4175	327,07	136,5517	13,10	149,6517
Manual	10	327,07	3 270,26	0	5 887,26
ChatRange	0,4175	327,07	136,5517	13,10	149,6517

Table 7.3: The total cost of producing an exercise using a manual method versus ChatRange. ChatRange sees a 99.813 percent reduction in calculated costs. The currency values are shown in NOK.

ChatRange

Chatrange uses the time it takes to generate exercises with the system and the time it takes to create the final exercise playbook. The last part can be automated, adding additional cost savings per exercise.

ChatRange developed the P2 scenario in 9 minutes and 36 seconds. Chapter 6 describes this process as highly manual, with several stops that could be automated to optimize time usage. However, to provide a fair assessment, this research project focuses on the worst possible scenario for generation, where the user has to recreate multiple parts of the exercise before being happy and, therefore, uses this timing.

The research team spent 15 minutes and 27 seconds to add the data to the exercise template.

In total, this equates to **0.4175 work hours**.

7.3.2 Cost per work hour

Chapter 3.2.4 explains that the cost is based on Norway's average yearly wage. As of February 2024, the government-run statistics bureau Statistics Norway calculated this to be 637 800 NOK [54]. The number is based on all sectors in Norway, including both genders.

It is possible to use sector-specific numbers in this metric; however, in most cases, like within the IT sector, this number would be higher than the average number and influence how well each solution is scored, scoring them too high in efficiency.

Statistics Norway estimates a work year to be 1950 hours in total. The cost per hour based on this is **327.07 NOK.**

7.3.3 Additional costs

Chatrange uses large language models from OpenAI and Groq. Groq's cost model is free as they are in the trial phase of developing their technology. In every case where Groq is used, a cost equal to that of using OpenAI's language models is added to provide a fair comparison. This principle would be used if choosing to run a local model, as this would incur hard-to-calculate maintenance, power, and equipment usage costs.

When using large language models, the cost is calculated based on the number of tokens used. The pricing differentiates between Input tokens (prompts sent to the models) and Output tokens (data received from the models). Table 7.4 shows the solution's cost to be **13.10 NOK** for the whole exercise playbook data.

Tokens In	Cost In	Tokens Out	Cost Out	Total tokens	Total Cost
9 647	3,23	93 836	10,9	103 483	**13,10**

Table 7.4: The total cost of using ChatRange is based on OpenAI's cost structure, shown in Norwegian Kroner. The currency values have been converted from the original USD to NOK, with an exchange rate of 1 USD to 10.81 NOK (06/05/2024).

7.3.4 Comparison

This section compares the efficiency results of P1 against P2. Table X shows that P2 scores a total cost of the generation of 79 805 Norwegian Krone. This starkly contrasts the manual method, which is calculated to be 150 Norwegian Krone.

The total cost of producing a scenario with ChatRange is only 0.187 percent of the cost of using a manual method, giving a total savings of **99.813 percent**.

The pessimistic number, where a premade template is used to develop the scenario, reduces the solution's efficiency. Here, ChatRange gives a total saving of **95.42 percent** instead. This, however, only reduces the overall efficiency by 4.393 percent points.

7.4 Scalability and Flexibility

ChatRange is built on open-source technologies and designed to be scalable and easy to implement. This approach is by design, as ChatRange's end-users are assumed to have less technical and academic expertise than seasoned professionals.

Run	Start	Finish	Time (s)
1	19:39:12	19:40:34	82
2	19:40:34	19:41:56	82
3	19:41:56	19:43:17	81
4	19:43:17	19:44:39	82
5	19:44:39	19:46:06	87
Average			82,8

Table 7.5: Overview of installation times for ChatRange on a clean Ubuntu installation.

7.4.1 LLM's Delivery Model

As discussed in Chapter 4.2, the cloud delivery models for the large language models are essential for the simplistic use of ChatRange. However, the system can also use self-rented hardware platforms and on-prem infrastructure to host the models. This implementation is made for the organizations that can run these solutions. Another positive with this implementation is the ability to use specialized or fine-tuned large language models for scenario development.

Using large language models and the autonomous AI agent infrastructure also provides an essential feature. It allows unlimited use of the solution as there are no investment requirements. The only cost in the solution is the direct cost of using private Language models, which can be further reduced by using hosted models.

7.4.2 Flexibility

The system is designed for easy implementation. This is done using Python [95], an open-source advanced scripting language. Python means the system can run on all platforms where native Python is supported, including Windows, Mac, and Linux.

The installation of ChatRange takes an average of 1.38 minutes. This has been tested using a clean installation of Ubuntu 22.04 and running the installation script [96] five times, with the results as seen in Table 7.5. In addition, some time must be expected to set up the required external services for the agents. However, once set up, these can be reused for several deployments. This is explained in the system documentation [96].

7.4.3 Usability

ChatRange uses a simple Web interface for scenario development, which reduces the need for specific technical knowledge of how large language models operate to use the solution.

7.5 Conclusion

ChatRange has been evaluated against a manual control created by CISA. This evaluation aimed to determine whether ChatRange delivered the expected quality and efficiency scores compared to a manual approach. The case studies used a blind test, where one of the scenarios was developed by humans and the other by ChatRange. The participants were not told which scenario was generated by what solution.

7.5.1 Quality

Two case studies evaluated six different criteria. The first looked at a broader population within Information technology in Norway's Public, government, and Private sectors. A total of 25 participants scored the two playbooks on the defined criteria in this case study. Playbook 2, which ChatRange generated, scored only **0.16 percent lower** across all criteria than the manual approach.

When looking at all the criteria together, ChatRange scored equal or better in **four of the six criteria**, as shown in Figure 7.5. The two categories where it scored lower were the usability of exercises in multiple scenarios and customization with human input. These scores were expected to be lower as ChatRange was designed to customize each playbook to a particular situation and, therefore, be less generalized.

The second case study used a range of experts to review each requirement and the exercises' totality. Both playbooks scored highly and were deemed usable for training exercises. One expert considered one module in Playbook 2 too technical for the exercise's intended participants, namely IT management. This is an excellent example of the language model going out of scope, as this is a known problem using this technology. Other than that, both scenarios scored well when looking at realism, and the exercise content was deemed appropriate.

Overall, the quality of the content produced by ChatRange exceeded the expectations for how the system would perform.

7.5.2 Efficiency

When examining the system's efficiency, ChatRange reduced the cost by an impressive **99.81 percent**. This, however, was to be expected, as going from a manual solution, which requires a great deal of time, to an automatic solution will give better results by default.

The scenario created by CISA was used as a ready-made scenario for IT management. An argument could be used where filling out the template should be considered the time taken to produce the manual scenario. If this were the case, the cost increase would only equal **4,393 percent points**.

7.5.3 Summary

Based on the evaluation performed in this chapter, the overall conclusion is that ChatRange can outperform or at least match manual solutions in both quality and efficiency. The system's ability to adapt to multiple scenarios and participants on-demand makes it highly versatile, and it scores well with experts reviewing the playbooks.

Chapter 8

Conclusion

Developing and planning cyber training exercises is time-consuming and demanding for the parties involved. NIST [19] estimates that an eight-hour exercise needs at least three months of planning time, and similarly, MITRE [25] estimates this number to be from one to three months. This puts organizations that want to conduct training in a difficult situation. They can either go into the public market and buy these services or spend their resources developing the exercises themselves. Either way, it leads to high costs.

This challenge also creates another situation where companies that lack the funds or resources to conduct training are essentially excluded from improving their security posture. This is one of the critical motivations for conducting this research project.

8.1 Summary

8.1.1 Large Language models

This part of the research focused on the following research question:

RQ1: What are suitable large language models for generative autonomous AI agent projects, their delivery methods, and their respective strengths and limitations?

A literature study was conducted to see if new technologies like large language models could be utilized in this research project and what models would fit the needs. For a project of this scope, it was essential to uncover the possibilities these models could give the project. One key issue that creates a technical barrier is the requirement to use local language models. This often comes with a technical debt that companies needing this technology must overcome. To better support these needs, a cloud delivery model was defined as the best method for providing access to LLMs.

Several models were considered, and based on the performance score and delivery models, the research team settled on using three models from two vendors. GPT 3.5 and GPT4 [64] from OpenAI have proven themselves through applications like ChatDev [45] and ChatGPT [84], which uses GPT 3.5 for software development tasks. The proprietary models must be used via OpenAI's cloud delivery model.

This research project's second model focused on is Mixtral 8x7B [62] from Mistral. This model is delivered through a cloud service named Groq [55], which provides access to models similar to OpenAI. One advantage of using Mixtral is the licensing of open source. This means the content can be used as the users of the models see fit, for example, in commercial projects.

A disadvantage to using cloud providers is the cost of using the service. However, what also needs to be considered when running the model locally on its hardware is that this also has costs, although more hidden. An example of this is maintenance cost and power usage. These modes are included in the price of cloud delivery. A huge advantage of cloud delivery is the speed of production. Companies that do not have the technical skills to run their systems can subscribe to the services and start using ChatRange without worrying about technological debt.

8.1.2 Autonomous AI Agents

This part of the research focused on the following research question:

RQ2: How can Autonomous AI agents increase the quality of the output delivery in contrast to traditional methods where only large language models are used, and what is the most optimal implementation for ChatRange?

When developing content using large language models, the user runs inference against the models. This is the same method used by ChatGPT, and in the simple form, it sends requests against the model, where the model returns an answer. This method, however, has some significant drawbacks. Large language models are trained in a substantial corpus of text and information databases, and these vary from model to model. Some models are even trained on inference from other models, such as mixture-of-experts. This, however, provides a challenge. If the models themselves are trained on data, how do we know whether the data provided to the models is real or fake? A more challenging question is whether the model can produce real answers or whether the answers provided are just realistic.

Another significant problem with this is the stop date for the information provided. When a model is trained, the training process often takes a long time, which requires a stop date for the information provided. For example, GPT 3.5 has a stop date of June 2021. No new information has been added to the language model after this date. If you were to ask the model for the president of the United States, the model would get this answer wrong.

One way to leverage this challenge is to implement external tools into the models to provide access to real-world data sources instead of relying on the models to provide answers. This, however, poses another challenge, and this is how we interact with the models. Going to search engines, typing queries manually, and then input this data into the LLMs solves one problem but does not reduce the need for human resources.

Autonomous AI Agents build on the logic of simulating normal user behaviors when interacting with LLMs. This is intended to use all LLMs' language capabilities to create reasoning and critical-thinking virtual users. These systems use the definition agents for these. Agents can work alone or in a group and can converse and pass tasks between each other, as a regular team would do in real life.

Three promising systems were identified: Autogen, ChatDev, and CrewAI. All of these use this architecture for developing content. CrewAI was selected for this project because it is the most versatile for this use case. The framework can be implemented directly into the solution and is easy to develop. It is also built on top of LangChain, which gives access to that framework's functionality, including a long list of external tools and integration, like Retrieval Augmented generation.

8.1.3 ChatRange

This part of the research focused on the following research question:

RQ3: How can autonomous AI agents be used alongside large language models to develop cyber security training scenarios?

ChatRange was proposed to enable companies and organizations worldwide to perform training. It allows the creation of training scenarios using a defined set of inputs, which are used throughout the creation process to tune how the different steps of the process are handled. Each step has a defined process for generating content and can take input from the user, memory, or external resources. This is described in the system as an Integrated design. The whole process is designed as a feedback loop, where the content is processed until it meets the quality criteria set. This helps to align with the objectives of the current step. This contrasts with a single inference where only a single output is given to a request.

ChatRange is designed using autonomous AI Agents and Static prompting with Role-playing. This has been shown to generate the most realistic output and reduce the problem of hallucinations that language models often experience. Autonomous AI Agents excel in research tasks in ChatRange, where there is an explicit need for external information. The system is designed around several teams, each consisting of two to three members. Each role is defined by background and goal. This helps the LLMs to tune into the given context of the conversation. The agents can use tools and converse with other agents in the team. In some teams, a manager role is added; this role helps distribute the different tasks to each team agent, increasing the work's efficiency.

8.1.4 Evaluation

This part of the research focused on the following research question:

RQ4: How do agent-based cyber training scenarios' details, realism, creativity, and efficiency measure against traditional manual approaches?

An evaluation consisting of two case studies was conducted to verify how well the system performed. The first case study focused on six criteria ranging from creativity and realism to how well the playbook could be used in other situations. The participants scored each criterion in a survey with a score from 0 to 5. They were presented with two playbooks, one created by CISA and modified by the research team and the other made by ChatRange. The survey was conducted as a blind test, as the participants were not told which playbook was created by whom.

Both playbooks scored highly in the first case study, with the playbook generated by ChatRange scoring only **0.16 percent** lower than the control, concluding that the quality equals the control.

The second case study involved using a range of Management, IT, and security experts to review each playbook. The experts did both a quality evaluation and a textual review, commenting on positive and negative feedback for each playbook. Overall, the experts scored similarly in both playbooks. One notable drawback of the playbook generated by ChatRange, promoted by the reviewer with experience in IT management, is that some of the data in the modules became too technical for the exercise target group. This was specified as IT management and shows that these systems still can go out of scope, even if the design is adequately managed.

ChatRange scored equal or better in **four of the six criteria** defined, scoring lower in the two criteria connected to how well a playbook is to be generalized and used in multiple scenarios, which is not the purpose of ChatRange. ChatRange is designed to create new exercises fast, as it is intended to customize the exercises to a specific scenario.

The second measurement looked at a measurement often used by organizations wanting to perform training, namely cost. This was defined with an optimistic number and a realistic number. The realistic number uses the researched time estimated to generate exercises by MITRE and NIST, with a slight reduction for only the planning phase to be 243 work hours. The optimistic number uses 10 hours as this is the time spent by the researchers to modify the CISA exercise template. However, this is only possible for the first time the exercise is run.

The system delivers a **99.81 percent reduction** in cost compared to ChatRange. It only costs approximately **150 Norwegian Kroner** (13.32 USD) per exercise to create, and the direct cost of using the system is only **13 Norwegian Kroner** (1.2 USD). The rest is manual labor of moving data into the template, which can be automated in the future.

The optimistic number shows a **95.417 percent reduction** in cost with the same calculations.

8.2 Overall conclusion

ChatRange delivers on the quality and cost measurements described above, making it a game-changer in organizations developing and running cyber training exercises. These organizations' ability to adapt quickly to new threats and provide training to crucial personnel is critical to protecting against current and new threats. The solution's ability to deploy rapidly into standard Ubuntu or Windows installations in minutes makes it versatile. and can be implemented without needing specialized knowledge of the architecture or access to specialized hardware behind ChatRange.

8.3 Research Limitations

As with most work that contains theory, this thesis and the methods used in the evaluations have limitations. This section describes these limitations and how further studies can improve them.

8.3.1 Constantly updating technology

Language models are a new technology, and working within this domain requires rapid changes. This has also been challenging as this thesis was executed part-time from August 1, 2023, to June 3, 2024. During this period, new models were continuously released, with added feature sets and the ability to provide better content with larger context windows, sometimes mitigating some of the earlier proposed research question challenges. Here is where the DSR methodology worked great; however, returning each time a new update was in place was time-consuming.

This has created a situation where a stop date for implementing new models has had to be imposed on the research. Therefore, some of the models in this thesis are older than what currently exists on the market and are an explicit limitation of this type of next-generation research. The system is, however, designed to be used with new models, so some of the problems can be reduced by performing tests in the future using the latest technology.

8.3.2 Case study

The first case studies were performed using a survey. Participants read the playbooks and then scored them using six criteria. The study aimed to find how well ChatRange scored against a manual control.

Participants

The survey was electronically distributed among IT professionals, management, and IT security experts in Norway's public and private sectors. This could lead to selection bias, as only participants from a single region are used, which may not represent the broader population. ChatRange is designed to help train personnel working within the specified sectors covered by the case studies. Further research into a wider population in multiple countries can reduce this bias, and additional studies within the field are recommended.

Data Collection

The second case study involved participants reviewing the scenarios in textual form. Here, three participants were selected, each with their area of expertise. This was done to get the best overall review of the cyber exercises with information from all fields. It was also explicitly done to reduce the possibility of expertise bias. If the survey were only distributed among security professionals, this could lead to a skewed picture of how well ChatRange performed, as it would only be reviewed by peers. The effect of reducing this bias is also shown in the results where the IT manager reviewer (D.3) pointed out that the exercise was too technical in parts when compared to the specified target, which was defined as IT managers.

Analysis

The analysis compared the studies' results using six predefined criteria. A challenge that can occur when analyzing the data is confirmation bias. To mitigate this, the study employs a set of criteria already established by other research performed by Yamin et al. [50], ensuring consistency. The study was also provided as a blind study, where the participants scored a playbook generated by humans alongside the one generated by ChatRange. They were not told what playbook was generated by whom and scored them blindly. The difference in scores was then compared in the evaluation, which is the result. This standardized review method provides a more objective assessment of the data.

Multiple cyber training exercises can be conducted to alleviate the potential biases proposed in this research. This is done to understand better how well each scenario performed. In addition, interviews can be conducted with each participant after each training scenario to get more in-depth data on how well each scenario performed. This would help better validate the specifics of each training scenario.

8.4 Further studies

The research presented in this report shows the potential that Autonomous AI agents can have on automatic complex and time-consuming tasks. This implementation can revolutionize how we think and use the technology large language models give us in new and exciting ways.

8.4.1 Hybrid and Full live exercises

The system can be extended further to cover hybrid and full live exercises. This will extend its potential to more technical personnel who now use cyber training or other exercise types like capture-the-flag training.

In addition, by including full-live exercises, the system can be tested alongside other solutions for running cyber training, providing better validity to the results as it makes it possible to compare results for the same training scenarios. Researchers Yamin et al. [50] are already exploring some of these use cases.

8.4.2 Simulated training

The architecture can be implemented into several stages of the cyber training lifecycle to fully utilize the agent infrastructure. This includes planning, executing, and reviewing the exercises. The agents can behave as individuals in the simulated environment or be able to form teams, like Red team, blue team, or White teams. This kind of simulation is similar to the research done by Park et al. [43] when developing their simulated "city." There, each agent could operate individually with their thoughts and reasoning.

Bibliography

[1] D. Adams, *The hitchhiker's guide to the galaxy.* New York: Pocket Books, 1979, 214 pp., OCLC: 1035593618.

[2] A. Petrosyan. 'Cybercrime: Monetary damage united states 2023,' Statista. (15th Mar. 2024), [Online]. Available: `https://www.statista.com/statistics/267132/total-damage-caused-by-by-cybercrime-in-the-us/` (visited on 10/05/2024).

[3] A. Petrosyan. 'Global cyberattacks by type 2022,' Statista. (1st Sep. 2023), [Online]. Available: `https://www.statista.com/statistics/1382266/cyber-attacks-worldwide-by-type/` (visited on 10/05/2024).

[4] NATO. 'What is NATO?' What is NATO? (2023), [Online]. Available: `https://www.nato.int/nato-welcome/index.html` (visited on 10/10/2023).

[5] ccdcoe. 'CCDCOE.' (2023), [Online]. Available: `https://ccdcoe.org/news/2023/6016/` (visited on 30/10/2023).

[6] M. C. Libicki, 'What is information warfare?' National Defense University, Institute for National Strategic Studies, Washington, DC, Technical Report ADA367662, 1st Aug. 1995. [Online]. Available: `https://apps.dtic.mil/sti/citations/ADA367662`.

[7] C. C. f. C. CSE, 'CYBER THREAT BULLETIN: Cyber threat activity related to the russian invasion of ukraine,' 2022. [Online]. Available: `https://www.cyber.gc.ca/sites/default/files/cyber-threat-activity-associated-russian-invasion-ukraine-e.pdf`.

[8] cybersecuritydive. 'US, allies blame russia for viasat cyberattack,' Cybersecurity Dive. (11th May 2022), [Online]. Available: `https://www.cybersecuritydive.com/news/viasat-cyber-russia-satellite/623560/` (visited on 12/10/2023).

[9] CCDCOE. 'NATO CCDOE - about us,' CCDOE About Us. (2023), [Online]. Available: `https://ccdcoe.org/about-us/` (visited on 12/10/2023).

[10] PricewaterhouseCoopers. 'Operation cloud hopper,' PwC. (2017), [Online]. Available: `https://www.pwc.co.uk/issues/cyber-security-services/insights/operation-cloud-hopper.html` (visited on 12/10/2023).

[11] mitre. 'menuPass, cicada, POTASSIUM, stone panda, APT10, red apollo, CVNX, HOGFISH, group g0045 | MITRE ATT&CK®.' (2023), [Online]. Available: `https://attack.mitre.org/groups/G0045/` (visited on 12/10/2023).

[12] D. Kushner, 'The real story of stuxnet,' *IEEE Spectrum*, vol. 50, no. 3, pp. 48–53, Mar. 2013, ISSN: 0018-9235. DOI: `10.1109/MSPEC.2013.6471059`. [Online]. Available: `http://ieeexplore.ieee.org/document/6471059/` (visited on 12/10/2023).

[13] C. Online. 'Russia-linked cyberattacks on ukraine: A timeline,' CSO Online. (2023), [Online]. Available: `https://www.csoonline.com/article/571865/a-timeline-of-russian-linked-cyberattacks-on-ukraine.html` (visited on 12/10/2023).

[14] Symantec. 'Ukraine: Disk-wiping attacks precede russian invasion.' (2022), [Online]. Available: `http://prod-blogs-ui.client-b1.bkjdigital.com/blogs/threat-intelligence/ukraine-wiper-malware-russia` (visited on 12/10/2023).

[15] Egress Software Technologies, *Insider data breach survey 2021*, 2021.

[16] A. Petrosyan. 'U.s. most reported cybercrime by victim number 2023,' Statista. (12th Apr. 2024), [Online]. Available: `https://www.statista.com/statistics/184083/commonly-reported-types-of-cyber-crime-us/` (visited on 10/05/2024).

[17] A. Borgeaud. 'Cybersecurity training in healthcare organizations u.s. 2022,' Statista. (7th Nov. 2023), [Online]. Available: `https://www.statista.com/statistics/736704/security-awareness-training-frequency-in-healthcare-organization-in-us/` (visited on 10/05/2024).

[18] E. John. 'Cyber security breaches survey 2023,' GOV.UK. (19th Apr. 2023), [Online]. Available: `https://www.gov.uk/government/statistics/cyber-security-breaches-survey-2023/cyber-security-breaches-survey-2023` (visited on 10/05/2024).

[19] NIST, *NIST special publication 800-84, guide to test, training, and exercise programs for IT plans and capabilities*, Published: National Institute of Standards and Technology, 16th Jan. 2020. [Online]. Available: `https://csrc.nist.gov/publications/detail/sp/800-84/final` (visited on 23/04/2021).

[20] ntnu. 'Om norwegian cyber range - NTNU.' (2023), [Online]. Available: `https://www.ntnu.no/ncr` (visited on 30/10/2023).

[21] C. Mellon. 'Fostering growth in professional cyber incident management,' Fostering Growth in Professional Cyber Incident Management. (17th Mar. 2024), [Online]. Available: `https://www.sei.cmu.edu/about/history-of-innovation-at-the-sei/display.cfm?customel_datapageid_40842=41019` (visited on 17/03/2024).

[22] 14:00-17:00. 'ISO/IEC 27001:2022,' ISO. (28th Mar. 2024), [Online]. Available: `https://www.iso.org/standard/27001` (visited on 20/03/2024).

[23] NIST. 'What is the NIST? what is the purpose of the NIST? | encryption consulting.' (23rd Sep. 2020), [Online]. Available: `https://www.encryptionconsulting.com/education-center/nist/` (visited on 09/05/2024).

[24] MITRE Corporation. 'Who we are.' Place: McLean, Virginia Publisher: MITRE. (2024), [Online]. Available: `https://www.mitre.org/who-we-are` (visited on 09/05/2024).

[25] J. Kick, 'Cyber exercise playbook,' MITRE Corporation, Technical Report, 15th Nov. 2014. [Online]. Available: `https://www.mitre.org/news-insights/publication/cyber-exercise-playbook` (visited on 06/05/2024).

[26] cisa. 'About CISA | CISA.' (1st May 2023), [Online]. Available: `https://www.cisa.gov/about` (visited on 09/05/2024).

[27] CISA. 'CISA tabletop exercise packages | CISA.' (1st Apr. 2024), [Online]. Available: `https://www.cisa.gov/resources-tools/services/cisa-tabletop-exercise-packages` (visited on 28/04/2024).

[28] M. Knüpfer, T. Bierwirth, L. Stiemert, M. Schopp, S. Seeber, D. Pöhn and P. Hillmann, 'Cyber taxi: A taxonomy of interactive cyber training and education systems,' in *Model-driven Simulation and Training Environments for Cybersecurity*, G. Hatzivasilis and S. Ioannidis, Eds., vol. 12512, Cham: Springer International Publishing, 2020, pp. 3–21, ISBN: 9783030624323 9783030624330. DOI: `10.1007/978-3-030-62433-0_1`. [Online]. Available: `https://link.springer.com/10.1007/978-3-030-62433-0_1` (visited on 22/03/2024).

[29] BTLO. 'Blue team labs online,' Blue Team Labs Online. (22nd Mar. 2024), [Online]. Available: `https://www.blueteamlabs.online/` (visited on 22/03/2024).

[30] N. CCDCOE. 'Locked shields,' Locked Shields. (10th May 2024), [Online]. Available: `https://ccdcoe.org/exercises/locked-shields/` (visited on 09/05/2024).

[31] nato. 'NATO review - an artificial intelligence strategy for NATO,' NATO Review. (25th Oct. 2021), [Online]. Available: `https://www.nato.int/docu/review/articles/2021/10/25/an-artificial-intelligence-strategy-for-nato/index.html` (visited on 15/10/2023).

[32] A. Hintze. 'Understanding the four types of AI, from reactive robots to self-aware beings,' The Conversation. (14th Nov. 2016), [Online]. Available: `http://theconversation.com/understanding-the-four-types-of-ai-from-reactive-robots-to-self-aware-beings-67616` (visited on 26/10/2023).

[33] M. Campbell, A. Hoane and F.-h. Hsu, 'Deep blue,' *Artificial Intelligence*, vol. 134, no. 1, pp. 57–83, Jan. 2002, ISSN: 00043702. DOI: `10.1016/S0004-3702(01)00129-1`. [Online]. Available: `https://linkinghub.elsevier.com/retrieve/pii/S0004370201001291` (visited on 26/10/2023).

[34] D. Silver, A. Huang, C. J. Maddison, A. Guez, L. Sifre, G. Van Den Driessche, J. Schrittwieser, I. Antonoglou, V. Panneershelvam, M. Lanctot, S. Dieleman, D. Grewe, J. Nham, N. Kalchbrenner, I. Sutskever, T. Lillicrap, M. Leach, K. Kavukcuoglu, T. Graepel and D. Hassabis, 'Mastering the game of go with deep neural networks and tree search,' *Nature*, vol. 529, no. 7587, pp. 484–489, 28th Jan. 2016, ISSN: 0028-0836, 1476-4687. DOI: `10.1038/nature16961`. [Online]. Available: `https://www.nature.com/articles/nature16961` (visited on 26/10/2023).

[35] goassoc. 'A comparison of chess and go | british go association.' (2023), [Online]. Available: `https://www.britgo.org/learners/chessgo.html` (visited on 26/10/2023).

[36] M. Dikmen and C. M. Burns, 'Autonomous driving in the real world: Experiences with tesla autopilot and summon,' in *Proceedings of the 8th International Conference on Automotive User Interfaces and Interactive Vehicular Applications*, Ann Arbor MI USA: ACM, 24th Oct. 2016, pp. 225–228, ISBN: 9781450345330. DOI: `10.1145/3003715.3005465`. [Online]. Available: `https://dl.acm.org/doi/10.1145/3003715.3005465` (visited on 27/10/2023).

[37] N. Cristovao. 'Musk shares details on FSD beta v11: Neural nets to be used for vehicle control,' Not a Tesla App. (15th Jan. 2023), [Online]. Available: `https://www.notateslaapp.com/software-updates/upcoming-features/id/1150/musk-shares-details-on-fsd-beta-v11-neural-nets-to-be-used-for-vehicle-control` (visited on 28/10/2023).

[38] A. M. Leslie, O. Friedman and T. P. German, 'Core mechanisms in 'theory of mind',' *Trends in Cognitive Sciences*, vol. 8, no. 12, pp. 528–533, Dec. 2004, ISSN: 13646613. DOI: `10.1016/j.tics.2004.10.001`. [Online]. Available: `https://linkinghub.elsevier.com/retrieve/pii/S1364661304002608` (visited on 28/10/2023).

[39] A. M. Turing, 'I.—COMPUTING MACHINERY AND INTELLIGENCE,' *Mind*, vol. LIX, no. 236, pp. 433–460, 1st Oct. 1950, ISSN: 1460-2113, 0026-4423. DOI: `10.1093/mind/LIX.236.433`. [Online]. Available: `https://academic.oup.com/mind/article/LIX/236/433/986238` (visited on 28/10/2023).

[40] D. A. Dahl, *Natural language understanding with Python: combine natural language technology, deep learning, and large language models to create human-like language comprehension in computer systems*. Birmingham Mumbai: Packt Publishing, 2023, 303 pp., ISBN: 9781804613429.

[41] T. B. Brown, B. Mann, N. Ryder, M. Subbiah, J. Kaplan, P. Dhariwal, A. Neelakantan, P. Shyam, G. Sastry, A. Askell, S. Agarwal, A. Herbert-Voss, G. Krueger, T. Henighan, R. Child, A. Ramesh, D. M. Ziegler, J. Wu, C. Winter, C. Hesse, M. Chen, E. Sigler, M. Litwin, S. Gray, B. Chess, J. Clark, C. Berner, S. McCandlish, A. Radford, I. Sutskever and D. Amodei, 'Language

models are few-shot learners,' 2020. DOI: 10.48550/ARXIV.2005.14165. [Online]. Available: https://arxiv.org/abs/2005.14165 (visited on 29/10/2023).

[42] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser and I. Polosukhin, 'Attention is all you need,' 2017. DOI: 10.48550/ARXIV.1706.03762. [Online]. Available: https://arxiv.org/abs/1706.03762 (visited on 29/10/2023).

[43] J. S. Park, J. C. O'Brien, C. J. Cai, M. R. Morris, P. Liang and M. S. Bernstein, 'Generative agents: Interactive simulacra of human behavior,' 2023. DOI: 10.48550/ARXIV.2304.03442. [Online]. Available: https://arxiv.org/abs/2304.03442 (visited on 04/11/2023).

[44] Wikipedia, *The sims (video game)*, in *Wikipedia*, Page Version ID: 1215164798, 23rd Mar. 2024. [Online]. Available: https://en.wikipedia.org/w/index.php?title=The_Sims_(video_game)&oldid=1215164798 (visited on 01/04/2024).

[45] C. Qian, X. Cong, W. Liu, C. Yang, W. Chen, Y. Su, Y. Dang, J. Li, J. Xu, D. Li, Z. Liu and M. Sun, 'Communicative agents for software development,' 2023. DOI: 10.48550/ARXIV.2307.07924. [Online]. Available: https://arxiv.org/abs/2307.07924 (visited on 04/11/2023).

[46] G. Li, H. A. A. K. Hammoud, H. Itani, D. Khizbullin and B. Ghanem, 'CAMEL: Communicative agents for "mind" exploration of large language model society,' 2023, Publisher: [object Object] Version Number: 2. DOI: 10.48550/ARXIV.2303.17760. [Online]. Available: https://arxiv.org/abs/2303.17760 (visited on 01/04/2024).

[47] S. Yao, J. Zhao, D. Yu, N. Du, I. Shafran, K. Narasimhan and Y. Cao, 'ReAct: Synergizing reasoning and acting in language models,' 2022, Publisher: [object Object] Version Number: 3. DOI: 10.48550/ARXIV.2210.03629. [Online]. Available: https://arxiv.org/abs/2210.03629 (visited on 01/04/2024).

[48] H. Chase, *LangChain*, original-date: 2022-10-17T02:58:36Z, Oct. 2022. [Online]. Available: https://github.com/langchain-ai/langchain (visited on 04/04/2024).

[49] J. Moura. 'crewAI - multi AI agents systems.,' crewai. (4th Apr. 2024), [Online]. Available: https://www.crewai.io/ (visited on 04/04/2024).

[50] M. M. Yamin, E. Hashmi, M. Ullah and B. Katt, *Applications of LLMs for generating cyber security exercise scenarios*, 26th Feb. 2024. DOI: 10.21203/rs.3.rs-3970015/v1. [Online]. Available: https://www.researchsquare.com/article/rs-3970015/v1 (visited on 01/04/2024).

[51] A. Zacharis and C. Patsakis, *AiCEF: An AI-assisted cyber exercise content generation framework using named entity recognition*, Version Number: 1, 2022. DOI: 10.48550/ARXIV.2211.10806. [Online]. Available: https://arxiv.org/abs/2211.10806 (visited on 20/05/2024).

[52] OASIS Open. 'STIX 2.1 specification,' STIX™ Version 2.1 Committee Specification Draft 01 / Public Review Draft 01. (Jul. 2019), [Online]. Available: https://docs.oasis-open.org/cti/stix/v2.1/csprd01/stix-v2.1-csprd01.html (visited on 20/01/2024).

[53] V. Vaishnavi and W. Kuechler, *Design science research methods and patterns: innovating information and communication technology*, Second edition. Boca Raton: CRC Press, Taylor & Francis Group, 2015, 373 pp., OCLC: ocn910968212, ISBN: 978-1-4987-1525-6.

[54] SSB. 'Lønn 2023,' SSB. (5th Feb. 2024), [Online]. Available: https://www.ssb.no/arbeid-og-lonn/lonn-og-arbeidskraftkostnader/statistikk/lonn (visited on 19/04/2024).

[55] groq. 'The groq LPU™ inference engine - groq.' (14th Nov. 2023), [Online]. Available: https://wow.groq.com/lpu-inference-engine/ (visited on 07/04/2024).

[56] paperspace. 'NVIDIA h100 for AI & ML workloads | cloud GPU platform | paperspace,' Paperspace.com. (23rd Jan. 2024), [Online]. Available: https://www.paperspace.com/ (visited on 23/03/2024).

[57] Google. 'Google colab,' Google Colab. (21st Jan. 2024), [Online]. Available: https://colab.research.google.com/ (visited on 23/01/2024).

[58] Investopedia. 'Capital expenditure (CapEx) definition, formula, and examples,' Investopedia. (7th Apr. 2024), [Online]. Available: https://www.investopedia.com/terms/c/capitalexpenditure.asp (visited on 07/04/2024).

[59] OpenAI. 'Introducing ChatGPT.' (5th Apr. 2024), [Online]. Available: https://openai.com/blog/chatgpt (visited on 05/04/2024).

[60] U. Kiguolis. 'Kovter malware infected millions of adult-themed website users.' (10th Oct. 2017), [Online]. Available: https://www.2-spyware.com/kovter-malware-infected-millions-of-adult-themed-website-users (visited on 23/03/2024).

[61] Investopedia. 'Operating expense definition and how it compares to capital expenses,' Opex. (7th Apr. 2024), [Online]. Available: https://www.investopedia.com/terms/o/operating_expense.asp (visited on 07/04/2024).

[62] A. Q. Jiang, A. Sablayrolles, A. Roux, A. Mensch, B. Savary, C. Bamford, D. S. Chaplot, D. d. l. Casas, E. B. Hanna, F. Bressand, G. Lengyel, G. Bour, G. Lample, L. R. Lavaud, L. Saulnier, M.-A. Lachaux, P. Stock, S. Subramanian, S. Yang, S. Antoniak, T. L. Scao, T. Gervet, T. Lavril, T. Wang, T. Lacroix and W. E. Sayed, 'Mixtral of experts,' 2024, Publisher: [object

Object] Version Number: 1. DOI: 10.48550/ARXIV.2401.04088. [Online]. Available: https://arxiv.org/abs/2401.04088 (visited on 06/04/2024).

[63] B. Brittain and B. Brittain, 'OpenAI hit with new lawsuits from news outlets over AI training,' *Reuters*, 28th Feb. 2024. [Online]. Available: https://www.reuters.com/legal/litigation/openai-hit-with-new-lawsuits-news-outlets-over-ai-training-2024-02-28/ (visited on 05/04/2024).

[64] OpenAI, J. Achiam, S. Adler *et al.*, 'GPT-4 technical report,' 2023, Publisher: [object Object] Version Number: 6. DOI: 10.48550/ARXIV.2303.08774. [Online]. Available: https://arxiv.org/abs/2303.08774 (visited on 05/04/2024).

[65] R. Anil, S. Borgeaud, J.-B. Alayrac *et al.*, 'Gemini: A family of highly capable multimodal models,' 2023, Publisher: [object Object] Version Number: 2. DOI: 10.48550/ARXIV.2312.11805. [Online]. Available: https://arxiv.org/abs/2312.11805 (visited on 05/04/2024).

[66] M. Reid, N. Savinov, D. Teplyashin *et al.*, 'Gemini 1.5: Unlocking multimodal understanding across millions of tokens of context,' 2024, Publisher: [object Object] Version Number: 1. DOI: 10.48550/ARXIV.2403.05530. [Online]. Available: https://arxiv.org/abs/2403.05530 (visited on 05/04/2024).

[67] Apache. 'Apache 2.0 license,' Apache 2.0 License. (6th Apr. 2024), [Online]. Available: https://www.apache.org/licenses/LICENSE-2.0.

[68] H. Touvron, T. Lavril, G. Izacard, X. Martinet, M.-A. Lachaux, T. Lacroix, B. Rozière, N. Goyal, E. Hambro, F. Azhar, A. Rodriguez, A. Joulin, E. Grave and G. Lample, 'LLaMA: Open and efficient foundation language models,' 2023, Publisher: [object Object] Version Number: 1. DOI: 10.48550/ARXIV.2302.13971. [Online]. Available: https://arxiv.org/abs/2302.13971 (visited on 06/04/2024).

[69] M. Nolan. 'Llama and ChatGPT are not open-source - IEEE spectrum,' IEEE Spectrum. (22nd Jul. 2023), [Online]. Available: https://spectrum.ieee.org/open-source-llm-not-open (visited on 06/04/2024).

[70] J. Vincent. 'Microsoft invests $1 billion in OpenAI to pursue holy grail of artificial intelligence,' The Verge. (22nd Jul. 2019), [Online]. Available: https://www.theverge.com/2019/7/22/20703578/microsoft-openai-investment-partnership-1-billion-azure-artificial-general-intelligence-agi (visited on 06/04/2024).

[71] M. Smith. 'Grokking x.ai's grok—real advance or just real troll? - IEEE spectrum,' Grokking X.ai's Grok—Real Advance or Just Real Troll? - IEEE Spectrum. (24th Apr. 2024), [Online]. Available: https://spectrum.ieee.org/open-source-ai-grok-llm (visited on 06/04/2024).

[72] D. Hendrycks, C. Burns, S. Basart, A. Zou, M. Mazeika, D. Song and J. Steinhardt, 'Measuring massive multitask language understanding,' 2020, Publisher: [object Object] Version Number: 3. DOI: 10.48550/ARXIV.2009.03300. [Online]. Available: https://arxiv.org/abs/2009.03300 (visited on 07/04/2024).

[73] K. Cobbe, V. Kosaraju, M. Bavarian, M. Chen, H. Jun, L. Kaiser, M. Plappert, J. Tworek, J. Hilton, R. Nakano, C. Hesse and J. Schulman, *Training verifiers to solve math word problems*, 17th Nov. 2021. arXiv: 2110.14168[cs]. [Online]. Available: http://arxiv.org/abs/2110.14168 (visited on 07/04/2024).

[74] M. Chen, J. Tworek, H. Jun, Q. Yuan, H. P. d. O. Pinto, J. Kaplan, H. Edwards, Y. Burda, N. Joseph, G. Brockman, A. Ray, R. Puri, G. Krueger, M. Petrov, H. Khlaaf, G. Sastry, P. Mishkin, B. Chan, S. Gray, N. Ryder, M. Pavlov, A. Power, L. Kaiser, M. Bavarian, C. Winter, P. Tillet, F. P. Such, D. Cummings, M. Plappert, F. Chantzis, E. Barnes, A. Herbert-Voss, W. H. Guss, A. Nichol, A. Paino, N. Tezak, J. Tang, I. Babuschkin, S. Balaji, S. Jain, W. Saunders, C. Hesse, A. N. Carr, J. Leike, J. Achiam, V. Misra, E. Morikawa, A. Radford, M. Knight, M. Brundage, M. Murati, K. Mayer, P. Welinder, B. McGrew, D. Amodei, S. McCandlish, I. Sutskever and W. Zaremba, *Evaluating large language models trained on code*, 14th Jul. 2021. arXiv: 2107.03374[cs]. [Online]. Available: http://arxiv.org/abs/2107.03374 (visited on 07/04/2024).

[75] R. Zellers, A. Holtzman, Y. Bisk, A. Farhadi and Y. Choi, 'HellaSwag: Can a machine really finish your sentence?,' 2019, Publisher: [object Object] Version Number: 1. DOI: 10.48550/ARXIV.1905.07830. [Online]. Available: https://arxiv.org/abs/1905.07830 (visited on 07/04/2024).

[76] paperwswithcode. 'Papers with code - the latest in machine learning,' Papers With Code. (7th Apr. 2024), [Online]. Available: https://paperswithcode.com/ (visited on 07/04/2024).

[77] EU. 'EU AI act: First regulation on artificial intelligence,' Topics | European Parliament. (6th Aug. 2023), [Online]. Available: https://www.europarl.europa.eu/topics/en/article/20230601STO93804/eu-ai-act-first-regulation-on-artificial-intelligence (visited on 23/02/2024).

[78] L. Ahlgren. 'Google's gemini AI won't be available in europe — for now,' TNW | Deep-Tech. (7th Dec. 2023), [Online]. Available: https://thenextweb.com/news/google-gemini-ai-unavailable-europe-uk (visited on 07/04/2024).

[79] S. Merken and S. Merken, 'New york lawyers sanctioned for using fake ChatGPT cases in legal brief,' *Reuters*, 26th Jun. 2023. [Online]. Available: https://www.reuters.com/legal/new-york-lawyers-sanctioned-using-fake-chatgpt-cases-legal-brief-2023-06-22/ (visited on 12/04/2024).

[80] W. D. Heaven. 'Google DeepMind wants to define what counts as artificial general intelligence,' MIT Technology Review. (16th Nov. 2023), [Online]. Available: `https://www.technologyreview.com/2023/11/16/1083498/google-deepmind-what-is-artificial-general-intelligence-agi/` (visited on 12/04/2024).

[81] Y. Nakajima. 'BabyAGI,' BabyAGI. (1st Apr. 2024), [Online]. Available: `https://github.com/yoheinakajima/babyagi` (visited on 01/04/2024).

[82] BabyAGI. 'Task-driven autonomous agent utilizing GPT-4, pinecone, and LangChain for diverse applications – yohei nakajima.' (28th Mar. 2023), [Online]. Available: `https://yoheinakajima.com/task-driven-autonomous-agent-utilizing-gpt-4-pinecone-and-langchain-for-diverse-applications/` (visited on 12/04/2024).

[83] H. Yang, S. Yue and Y. He, 'Auto-GPT for online decision making: Benchmarks and additional opinions,' 2023. DOI: `10.48550/ARXIV.2306.02224`. [Online]. Available: `https://arxiv.org/abs/2306.02224` (visited on 13/04/2024).

[84] OpenAI. 'ChatGPT,' OpenAI ChatGPT. (), [Online]. Available: `https://chat.openai.com` (visited on 14/01/2024).

[85] Microsoft. 'Early MicroSoft employees - CHM revolution,' Early MicroSoft employees. (8th Sep. 2023), [Online]. Available: `https://www.computerhistory.org/revolution/personal-computers/17/305/1228` (visited on 14/04/2024).

[86] J. Wei, X. Wang, D. Schuurmans, M. Bosma, B. Ichter, F. Xia, E. Chi, Q. Le and D. Zhou, 'Chain-of-thought prompting elicits reasoning in large language models,' 2022. DOI: `10.48550/ARXIV.2201.11903`. [Online]. Available: `https://arxiv.org/abs/2201.11903` (visited on 15/04/2024).

[87] Openai. 'Codex,' Codex. (1st Feb. 2024), [Online]. Available: `https://openai.com/blog/openai-codex` (visited on 15/04/2024).

[88] Q. Wu, G. Bansal, J. Zhang, Y. Wu, B. Li, E. Zhu, L. Jiang, X. Zhang, S. Zhang, J. Liu, A. H. Awadallah, R. W. White, D. Burger and C. Wang, 'AutoGen: Enabling next-gen LLM applications via multi-agent conversation,' 2023, Publisher: [object Object] Version Number: 2. DOI: `10.48550/ARXIV.2308.08155`. [Online]. Available: `https://arxiv.org/abs/2308.08155` (visited on 16/04/2024).

[89] M. Shridhar, X. Yuan, M.-A. Côté, Y. Bisk, A. Trischler and M. Hausknecht, 'ALFWorld: Aligning text and embodied environments for interactive learning,' 2020, Publisher: [object Object] Version Number: 2. DOI: `10.48550/ARXIV.2010.03768`. [Online]. Available: `https://arxiv.org/abs/2010.03768` (visited on 17/04/2024).

[90] V. Debia. 'AutoGen studio: Interactively explore multi-agent workflows | AutoGen,' AutoGen Studio: Interactively Explore Multi-Agent Workflows. (1st Dec. 2023), [Online]. Available: https://microsoft.github.io/autogen/blog/2023/12/01/AutoGenStudio (visited on 01/04/2024).

[91] Langchain. 'Toolkits | LangChain,' Tookits. (17th Apr. 2024), [Online]. Available: https://python.langchain.com/docs/integrations/toolkits/ (visited on 17/04/2024).

[92] c. Inc. 'crewAI tools - crewAI,' CrewAi Tools. (17th Apr. 2024), [Online]. Available: https://crewai.com/core-concepts/Tools/ (visited on 17/04/2024).

[93] L. martin. 'Cyber kill chain®,' Lockheed Martin. (3rd Apr. 2024), [Online]. Available: https://www.lockheedmartin.com/en-us/capabilities/cyber/cyber-kill-chain.html (visited on 23/04/2024).

[94] M. Dekking, Ed., *A modern introduction to probability and statistics: understanding why and how*, Springer texts in statistics, London: Springer, 2005, 486 pp., ISBN: 978-1-85233-896-1.

[95] P. S. Foundation. 'Welcome to python.org,' Python.org. (8th May 2024), [Online]. Available: https://www.python.org/ (visited on 20/05/2024).

[96] S. Storebø, *ChatRange: Streamlining cyber training exercises*, 20th May 2024. [Online]. Available: https://github.com/trackanstian/ChatRange-Public.

Appendix A

ChatRange

A.1 Description

These are the appendixes for the ChatRange system. Here, you will find the whole architecture and setup of all the solution components. This has been added here as it is non-relevant to the specific research questions in the thesis.

A.2 Overview

This overviews ChatRange's content-generation process and the technology used in each step.

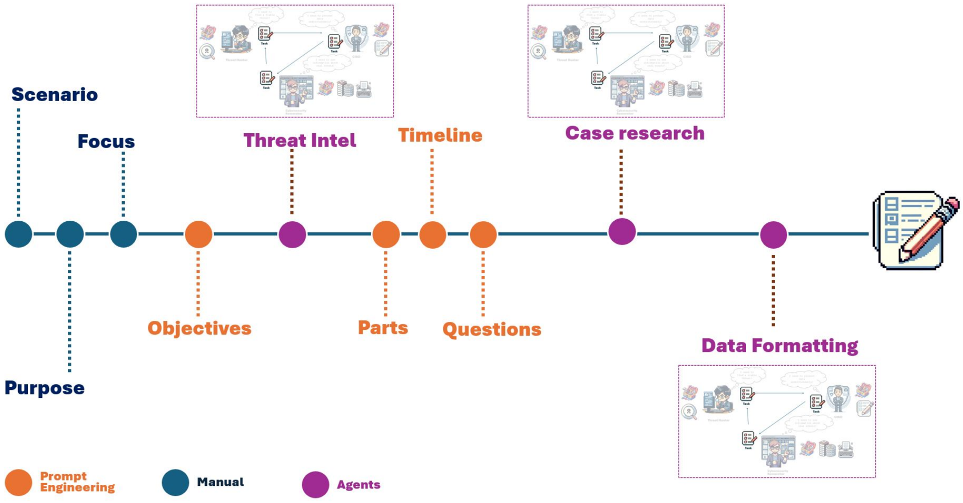


Figure A.1: A overview of each of the steps for ChatRange

A.3 Agents

A.3.1 Threat hunter

Background

An expert in cybersecurity with a keen eye for uncovering hidden threats, the Threat Hunter has a decorated history of neutralizing high-profile cyberattacks. With a background in digital forensics and a passion for cybersecurity innovation, they have become a pivotal asset in preemptive threat detection and resolution, safeguarding critical infrastructure from potential cyber disasters

Goal

Identify and neutralize advanced persistent threats (APTs) within complex digital environments, employing cutting-edge technology and methodologies to detect, analyze, and disarm sophisticated cyber threats before they can exploit vulnerabilities

A.3.2 Cybersecurity Researcher

Background

Explore emerging cybersecurity threats, create new defenses, and advance knowledge on cyber protection. Use advanced analytics and innovative research to understand cyber attacks and develop solutions for stronger digital security.

Goal

Driven by a keen interest in the digital domain and sharp analytical skills, the Cybersecurity Researcher excels in uncovering and combating cyber threats. A specialist with a strong computer science background, their work in threat detection and prevention has established them as a cybersecurity authority. They are dedicated to defending digital infrastructures and ensuring data security against advanced cyber attacks.

A.3.3 Chief Information Security Officer (CISO)

Background

Develop and enforce information security policies, manage risk, and ensure compliance with regulations.

Goal

This CISO is a seasoned cybersecurity professional known for safeguarding organizations from cyber threats.

A.4 Teams

A.4.1 Threat Intelligence

The threat intelligence team is used to hunt for a real threat actor who supports the scenario and its methods.

Type: Sequential

Members:

1. Threat hunter
2. Cybersecurity researcher

A.4.2 Case Studies

The case studies team is created to find real cyber security incidents that match the scenario proposed in the solution.

Type: Sequential

Members:

1. Chief Information Security Officer (CISO)
2. Cybersecurity researcher

A.4.3 Case Research

The case research team is tasked with finding specific information on each of the case studies from the previous step. It must also present the information in a structured format.

Type: Sequential

Members:

1. Chief Information Security Officer (CISO)
2. Cybersecurity researcher

A.5 Tools

The tools are available for each agent to use when performing the assigned tasks.

A.5.1 Search Tool

CrewAI includes search tools, and Serper fetches the top search results from Google, along with metadata and a short description of the page.

Settings

API Key Required: Yes
URL: serper.dev

A.5.2 Exa Search

Exa search is a search and data fetching service specifically built for AI applications. This is a custom tool implemented in ChatRange, from LangChain. The tool provides access to three critical sub-tools used in ChatRange:

- **Search**: Runs a semantic search for a webpage
- **Find Similar**: Fines similar results based on a similar URL.
- **Get Contents**: Gets the contents of a specific webpage along with its metadata.

Settings

API Key Required: Yes
URL: exa.ai

A.5.3 Suggester

This is a custom role that only exists in ChatRange. It was specifically made to help stuck agents in a loop. The Toll consists of a specifically created agent and task, and its job is to suggest a better search string to break a possible loop. The tool takes a string as an input.

Data suggester

Background
You're a Principal Researcher at a big company and you need to do a research about a given topic.

Role
Finding amazing solutions when you are stuck with a problem.

Task

Suggest a better query for the following search: "{query}"

A.5.4 Wikipedia

Wikipedia is a custom tool implemented in ChatRange from LangChain. The tool gives direct access to Wikipedia and is a critical component of ChatRange. The tool provides two essential sub-tools:

- Search: This tool gives access to Wikipedia search and returns a set of results.
- Get Info: Get a single page of text from Wikipedia.

Settings

API Key Required: No

A.6 Tasks

The tasks are the processes for each team. This section details each of them. All variables inputted to the agent are shown with {}. More information about them can be found in the code. The text describes each task and the expected output to tune the agent's answer.

A.6.1 Threat Intelligence

These are the tasks used in this process.

Define Situation

```
Your job is to analyze the scenario and find the relevant incident types

1. Search for more information about the incident type given.
2. Create a summary of the incident type.

Scenario: {self.scenario}

Notes: {self.__tip_section()}
```

Expected Output:

```
A brief description of the incident types in markdown.

It must contain the following:
## Incident type
## Description
## MITRE ATT\&CK techniques
```

Create Setting

```
Task: Search for a relevant threat actor using the MITRE ATT&CK techniques. You
    MUST use a real Threat actor for the setting.

Notes: {self.__tip_section()}
```

Expected Output:

```
A report of the threat actor markdown.
It must contain the following:
## Threat actor name
## Description
## Modus operandi
## Associated groups
## MITRE ATT&CK Techniques Used
```

A.6.2 Case Studies

These are the tasks used in this process.

Get Mitre Techniques

```
Task: Get the list of MITRE ATT&CK techniques in the text
Incident type: "{self.incident_type}"

Notes: {self.__tip_section()}
```

Expected Output:

```
A list of MITRE ATT&CK techniques in json.
format:
[{{"technique": "string"}}]
```

Determine Categories

```
Task: Determine what kind of attack the MITRE ATT&CK techniques is used for.
Examples: Phishing, ransomware, trojan, insider threat, etc.

Notes: {self.__tip_section()}
```

Expected Output:

```
A list of MITRE ATT&CK techniques and their categories in json.
format:
[
    {{
        "technique": "string",
        "type_of_attack": "string"
    }}
]
```

Research Case Studies

```
Task: I need the broad categories, such as Phishing or Ransomware, that each MITRE
    ATT&CK technique falls under, rather than specific examples like spearphishing
    or CEO fraud. Write a summary for each category.

Description. Only use the technique category when searching for information. Do not
     include the MITRE ATT&CK ID in your search query. For example, search for "
    Phishing Emails" instead of "Initial Access: Phishing Emails (T1566)".

Notes: {self.__tip_section()}
```

Expected Output:

```
A summary for each category in JSON.
format:
[
    {{
        "category": "string",
        "summary": "string"
    }}
]
```

Find Real World Examples

```
Task: For each category provided: List of cyber attacks that match
the category. At least three examples for each category.

Type: This must be a category from the case studies.

Example:
[
    {{"threat": "Hillary Clinton’s 2016 presidential campaign
    spear phishing attacks", "category": "phishing"}},
    {{"threat": "Threat Group-4127 (Fancy Bear) targeting Google accounts",
    "category": "phishing"}},
    {{"threat": "Whaling attacks targeting senior executives",
    "category": "phishing"}},
    {{"threat": "WannaCry ransomware attack", "category": "ransomware"}},
    {{"threat": "Taiwan Semiconductor Manufacturing Company (TSMC)
    ransomware attack", "category": "ransomware"}}
]

Notes: {self.__tip_section()}
```

Expected Output:

```
A list of the examples in JSON. At least three examples for each category.
format:
[
    {{"threat":"string", "category":"string"}},
]
```

Get references

```
Task: I need you to do research for additional resources on the
categories presented here. The resrouces must be from
reputable sources.

Find at least four resources for each category.

Notes: {self.__tip_section()}
```

Expected Output:

```
format:
[
    {{
        "category": "string",
        "title": "string",
        "url": "link"
    }}
]
```

A.6.3 Case Research

```
Task: Research and create a blog post with a summary of this attack. It must
     contain a title and 2 paragraphs:
"{self.case}"
```

```
Description: NEVER deviate from the format. Only provide a timeline of the attack.
    Do not include any other information. Do not add any other titles.

Notes: {self.__tip_section()}
```

Expected Output:

```
A case study in markdown.
format:
# Title
Paragraph 1
Paragraph 2
```

A.7 Static Prompts

These are the static prompts used in ChatRange as can be seen in the timeline in A.2.

A.7.1 Exercise Objectives

```
You are a Chief Information Security Officer (CISO), and you serve as the guardian
    of your organization’s digital assets, implementing robust cybersecurity
    strategies, leading incident response efforts, and fostering a culture of
    security awareness and compliance throughout the company. You possess a deep
    understanding of evolving cyber threats, expert knowledge in risk management,
    and the ability to communicate effectively with stakeholders at all levels to
    mitigate risks and ensure the integrity, confidentiality, and availability of
    critical data and systems.

Task: Create at least five exercise objectives for a exercise. The exercise
    objecives MUST be relative to the exercise type and participants.
You MUST include the organization name in each objective text.

Exercise type: {exercise_type}
Participants: {participants}
Organization: {organization}
Scenario: {scenario}

==RULES==
Answers must be in a valid JSON array.
{focus}

Examples:
Determine the effectiveness of the cyber education provided to the training
    audience prior to the start of the exercise
Assess effectiveness of the organization’s/exercise’s incident reporting and
    analysis guides for remedying deficiencies


Output format:
["This is an suggestion", "This is another suggestion", "This could be the third
    suggestion"]
```

A.7.2 Determine Parts

```
Task: Decide if we should divide this exercise into multiple parts based on the
    Purpose and Scenario. The parts MUST only focus on the incident, not the
    response. Return the number of parts and a description for each part.

Purpose: {purpose}
Scenario: {scenario}

Expexted output: Maximum two parts, divided logically by using the cyber kill chain
    .

Notes: Only reply with the answer to the question and a title for the parts.
```

A.7.3 Create Timeline

```
Task: Create a story timeline leading up yo the start point of the exercise
     scenario. Incorporate the target company , Acme Inc, and the threat actor,
     Phobos Ransomware Group, into the story. use a formal and factual language for
     use in incident report.

Description: The story must go over several days and there must be a clear
     progression of events that happen. The days must be in chronological order but
     there must be from 1 to 7 days between each day. Only write the day number as
     title title for the day.
Only write the story up to the point where the first part of the exercise will
     begin.

Create one story for each part. Each part must have at least four days in the
     timeline.

Expexted Output: A timeline with a title for each day and a story for each day.
     Only write what is the expexted output.

Parts: {parts}
```

A.8 Storage

The system uses REDIS as the storage platform. This is an environment variable that can be set in the .env file for ChatRange. If you want to use REDIS Cloud (Free) you can input the hostname here and change REDIS_EMULATOR to true.

Please note that if you opt to use fakeredis all progress is lost when shutting down ChatRange as this is only stored in memory.

Config File:

```
REDIS_EMULATOR=true
REDIS_HOST=<hostname>
REDIS_PORT=13997
REDIS_DB=0
```

Appendix B

Databank

This overviews all the data collected by analyzing ChatRange's performance. T also contains the scoring table used in the survey and analysis of the data.

B.1 Scoring table

This is the scoring table used to evaluate the results of the survey.

B.1.1 Details:

- **0 (None)**: No adherence to real-world settings; content and threats are unrealistic or absent.
- **1 (Very Low)**: Minimal effort to create real-world settings; content and threats barely reflect reality.
- **2 (Low)**: Some elements of real-world settings are present; content and threats are somewhat relevant but not fully convincing.
- **3 (Moderate)**: Adequate real-world settings; content and threats are relevant and reasonably elaborated.
- **4 (High)**: Strong representation of real-world settings; content and threats are relevant and well-elaborated.
- **5 (Very High)**: Exceptional depiction of real-world settings; content and threats are incredibly relevant and highly detailed.

B.1.2 Technical Soundness

- **0 (None)**: No adherence to core cybersecurity principles; exceeds current technological capabilities.
- **1 (Very Low)**: Minimal adherence to cybersecurity principles; occasionally exceeds current technology limitations.
- **2 (Low)**: Some alignment with cybersecurity principles frequently stretches current technology limitations.

- **3 (Moderate)**: Generally aligns with cybersecurity principles; stays within current technology limitations.
- **4 (High)**: Strong alignment with core cybersecurity principles; fully respects current technology limitations.
- **5 (Very High)**: Perfect alignment with core principles; optimally uses current technology without exceeding limitations.

B.1.3 Realism

- **0 (None)**: Completely unrealistic; does not reflect actual cybersecurity TTPs or scenarios.
- **1 (Very Low)**: Barely realistic; limited and incorrect application of actual TTPs.
- **2 (Low)**: Somewhat realistic; partially correct TTPs but lacks complexity.
- **3 (Moderate)**: Reasonably realistic; mostly accurate TTPs with adequate complexity.
- **4 (High)**: Very realistic, accurate TTPs that reflect complex, real-world cyber threats.
- **5 (Very High)**: Extremely realistic; perfectly capturing the complexity and unpredictability of cybersecurity incidents.

B.1.4 Creativity

- **0 (None)**: No creativity; repetitive and uninspired scenarios.
- **1 (Very Low)**: Barely any creativity; few new or innovative elements.
- **2 (Low)**: Some creativity; occasional new ideas but lacks diversity.
- **3 (Moderate)**: Moderately creative; various new and diverse scenarios.
- **4 (High)**: Highly creative; fosters critical and innovative thinking effectively.
- **5 (Very High)**: Exceptionally creative; consistently provides innovative and thought-provoking scenarios.

B.1.5 Usability in exercises

- **0 (None)**: Not usable in training exercises.
- **1 (Very Low)**: Barely usable; significant adjustments needed to be part of an exercise.
- **2 (Low)**: Somewhat usable; requires moderate adjustments for integration.
- **3 (Moderate)**: Usable with minor adjustments; fits well into broader training exercises.
- **4 (High)**: Highly usable; integrates seamlessly into various training formats.
- **5 (Very High)**: Perfect usability; designed to be easily incorporated into any training scenario.

B.1.6 Expandability with human inputs

- **0 (None)**: No capacity for incorporating human input or real-world events.
- **1 (Very Low)**: Minimal capacity; occasionally includes expert feedback with limited relevance.
- **2 (Low)**: Some capacity; can include expert feedback but not always effectively.
- **3 (Moderate)**: Good capacity; effectively incorporates expert feedback and real-world events.
- **4 (High)**: Strong capacity; frequently updates and refines scenarios based on expert input and actual events.
- **5 (Very High)**: Excellent capacity; integrates expert feedback seamlessly and dynamically adapts to real-world changes.

B.2 Raw Data

B.2.1 Playbook 1 - Human Generated (P1)

Participant ID	Timestamp	Field	Details	Tech.	Realism	Creativity	Usab.	Exbandab.
1	30.04.2024 11:49	Security	4	5	4	4	4	4
2	05.03.2024 10:07	Management	5	5	5	5	5	5
3	05.03.2024 14:04	Security Management	4	5	5	4	5	4
4	29.04.2024 15:37	CISO	5	4	5	5	5	5
5	30.04.2024 11:52	Security	4	5	4	4	5	5
6	30.04.2024 12:05	IT	4	4	4	3	5	5
7	30.04.2024 14:51	NOC Engineer	1	1	1	1	1	1
8	30.04.2024 15:05	Student	4	3	1	2	4	5
9	30.04.2024 19:50	IT	4	3	5	2	3	4
10	30.04.2024 22:51	Security	4	3	4	4	5	4
11	01.05.2024 10:42	Business security officer	4	5	4	4	4	4
12	01.05.2024 11:18	IT	5	5	5	5	5	5
13	03.05.2024 13:49	IT	4	5	3	3	4	4
14	03.05.2024 15:38	Security	4	4	3	3	4	3
15	29.04.2024 10:42	IT	4	4	4	4	4	4
16	29.04.2024 10:58	IT management	4	3	3	3	4	4
17	29.04.2024 12:01	IT-Security	2	4	3	5	5	4
18	29.04.2024 14:35	IT - Support	4	5	4	5	4	5
19	30.04.2024 13:47	Security	5	4	4	4	5	5
20	30.04.2024 17:09	IT security	4	4	5	3	5	4
21	02.05.2024 12:26	IT-Ansvarlig	5	5	5	5	5	5
22	02.05.2024 13:35	Security	5	5	5	5	5	5
23	03.05.2024 06:28	IT-architecture	4	3	4	3	3	3
24	03.05.2024 10:12	Management	5	5	5	5	5	5
25	03.05.2024 14:06	security management	4	4	4	4	4	4

B.2.2 Playbook 2 - Chatrange Generated (P2)

Participant	Timestamp	Field	Details	Tech.	Realism	Creativity	Usab.	Exbandab.
1	30.04.2024 11:49	Security	4	5	4	4	5	5
2	05.03.2024 10:07	Management	5	5	5	5	5	5
3	05.03.2024 14:04	Security Management	4	4	4	4	4	4
4	29.04.2024 15:37	CISO	5	5	5	5	5	5
5	30.04.2024 11:52	Security	4	5	4	4	5	4
6	30.04.2024 12:05	IT	4	5	4	3	4	3
7	30.04.2024 14:51	NOC Engineer	1	1	1	1	1	1
8	30.04.2024 15:05	Student	3	4	3	2	4	5
9	30.04.2024 19:50	IT	2	3	2	4	4	3
10	30.04.2024 22:51	Security	5	3	4	4	4	4
11	01.05.2024 10:42	Business security officer	5	5	5	4	4	4
12	01.05.2024 11:18	IT	5	5	5	5	5	5
13	03.05.2024 13:49	IT	4	4	4	3	4	4
14	03.05.2024 15:38	Security	3	4	4	4	3	4
15	29.04.2024 10:42	IT	4	4	4	4	4	4
16	29.04.2024 10:58	IT management	4	3	4	4	4	3
17	29.04.2024 12:01	IT-Security	4	4	4	4	4	5
18	29.04.2024 14:35	IT - Support	5	4	4	5	5	4
19	30.04.2024 13:47	Security	5	5	4	4	4	5
20	30.04.2024 17:09	IT security	4	3	4	4	4	5
21	02.05.2024 12:26	IT-Ansvarlig	5	5	5	5	5	5
22	02.05.2024 13:35	Security	5	5	5	5	5	5
23	03.05.2024 06:28	IT-architecture	3	3	4	3	3	2
24	03.05.2024 10:12	Management	5	5	5	5	5	5
25	03.05.2024 14:06	security management	4	4	4	4	4	4

Appendix C

Exercise Playbooks

C.1 Description

The playbooks are the exercise scenarios presented in the survey.

C.2 CISA Generated Playbook

ACME INC

Ransomware – Third Party Vendor

01.04.2024

NTNU

Handling Instructions

TLP: WHITE

The title of this document is Ransomware – Third party vendor Situation Manual. This document is unclassified and designated as "*Traffic Light Protocol (TLP): WHITE.* This designation is used when information requires support to be effectively acted upon, yet carries risks to privacy, reputation, or operations if shared outside of the organizations involved. Recipients may only share TLP:WHITE information with members of their own organization, and with clients or customers who need to know the information to protect themselves or prevent further harm. **Sources are at liberty to specify additional intended limits of the sharing: these must be adhered to.**

Exercise Overview

Exercise Name		
Exercise Date, Time, and Location	01.04.2024 Time (e.g. 9:00 a.m. – 15:00 p.m.) NTNU Cyber Range	
Exercise Schedule	**Time**	**Activity**
	09:00	Welcoming Remarks and Introductions
	09:15	Exercise Briefing (Objectives, Rules of Engagement, etc.)
	10:00	Module 1
	12:00	Module 2
	14:00	Debrief / Hotwash
	15:00	Finish
Scope	6 hours facilitated, discussion-based tabletop exercise	
Purpose	To examine the coordination, collaboration, information sharing, and response capabilities of Acme Inc in reaction to a ransomware incident with third party compromise by phishing.	
Exercise Focus	Identify, Protect, Respond, Recover	
Objectives	1. Discuss elements of Acme Inc's cybersecurity posture. 2. Examine Acme Inc's cybersecurity information sharing procedures and mechanisms. 3. Examine Acme Inc's cyber incident response plans or playbooks.	

Exercise Name	
Threat or Hazard	Cyber
Scenario	A threat actor targets a third-party vendor through a phishing email as an entry point into Acme Inc networks/systems. Attackers cause computer latency and network access issues and install ransomware on Acme Inc computers.
Participating Organizations	IT-Management
Points of Contact	

General Information

Participant Roles and Responsibilities

The term *participant* encompasses many groups of people, not just those playing in the exercise. Groups of participants involved in the exercise, and their respective roles and responsibilities, are as follows:

Players have an active role in discussing or performing their regular roles and responsibilities during the exercise. Players discuss or initiate actions in response to the simulated emergency.
Observers do not directly participate in the exercise. However, they may support the development of player responses to the situation during the discussion by asking relevant questions or providing subject matter expertise.
Facilitators provide situation updates and moderate discussions. They also provide additional information or resolve questions as required. Key Exercise Planning Team members may also assist with facilitation as subject matter experts during the exercise.
Note-takers are assigned to observe and document exercise activities. Their primary role is to document player discussions, including how and if those discussions conform to plans, policies, and procedures.

Exercise Structure

This exercise will be a facilitated exercise. Players will participate in the following:

- Cyber threat briefing (if desired)
- Scenario modules:
  - **Module 1. – Third party vendor**
  - **Module 2 – Ransomware attack**
- Hotwash

Exercise Guidelines

- This exercise will be held in an open, no-fault environment. Varying viewpoints are expected.
- Respond to the scenario using your knowledge of existing plans and capabilities, and insights derived from your training and experience.
- Decisions are not precedent setting and may not reflect your organization's final position on a given issue. This exercise is an opportunity to discuss and present multiple options and possible solutions and/or suggested actions to resolve or mitigate a problem.
- There is no hidden agenda, and there are no trick questions. The resources and written materials provided are the basis for discussion.
- The scenario has been developed in collaboration with subject matter experts and exercise planners from your organization.
- In any exercise, assumptions and artificialities are necessary to complete play in the time allotted, to achieve training objectives, and/or account for logistical limitations. Please do not allow these considerations to negatively impact your participation in the exercise.

Exercise Hotwash and Evaluation

The facilitator will lead a hotwash with participants at the end of the exercise to address any ideas or issues that emerge from the exercise discussions.

Module 1:

Day 1

The Department of Homeland Security (DHS) Cybersecurity and Infrastructure Security Agency (CISA) released an alert regarding a new ransomware variant. This ransomware is being used in a campaign targeting state, local, tribal, and territorial (SLTT) governments and private sector firms.

Day 4

Norwegian National Cyber Security Center releases an alert on a recently observed phishing campaign. The phishing emails contain a malicious attachment that, when opened, installs malware on a user's machine without their knowledge. The phishing emails mention either required updates to critical human resources (HR) documents or an invoice that needs to be paid by your organization.

Day 11

Indigo Software employees receive an email from their benefits department asking them to ensure their information is correct before the new fiscal year starts. Attached to the email is a document for users to review and update, as needed. Some users report the email as suspicious while others open the email and submit the form.

Day 16

Employees at Acme Inc receive an email from Indigo Software regarding an invoice for this month's expenses. An employee opens the email and finds the document is blank. The employee emails Indigo Software to get clarification on the email and attachment. Indigo Software states they have no record of sending that email and are looking into its origins.

Discussion Questions

1. This scenario describes two cybersecurity alerts. Would you receive these alerts?
    a. What sources of cybersecurity threat intelligence does your organization receive?
    b. What cyber threat information is most useful?
    c. Is the information you receive timely and actionable?
    d. Who is responsible for collating information across your organization?
    e. What actions would you take based on the cybersecurity threat intelligence presented in the scenario?
    f. Who else do you share cybersecurity threat intelligence with?
        i. Staff?
        ii. Leadership?
        iii. Third-party vendors
2. Does your department or agency provide basic cybersecurity and/or information technology (IT) security awareness training to all users (including managers and senior executives)?
    a. What does your training cover?
    b. Is training required to obtain network access?
3. What training does your organization require for third-party vendors who have access to your organization's information systems?
4. Has your organization conducted a cyber risk assessment to identify organization-specific threats, vulnerabilities, and critical assets or data?

   a. What are your most significant threats and vulnerabilities?

5. Do you have a patch management plan/program in place? If you do have a patch management plan:
   a. Are risk assessments performed on all servers on the network?
   b. Are processes in place to proactively evaluate each server's criticality and applicability to software patches?
   c. Does this plan include a risk management strategy that addresses the following considerations?
      i. The risks of not patching reported vulnerabilities?
      ii. Extended downtime?
      iii. Impaired functionality?
      iv. The loss of data?

6. How do employees report suspected phishing attempts?
   a. Are there formal policies or plans that would be followed?
   b. What actions does your organization take when suspicious emails are reported?
   c. Does your organization conduct phishing self-assessments?

7. Would any of the events described in this module be identified as cyber incidents or events? If so, how would they be handled?

Module 2:

Day 47

Several employees call the IT help desk complaining about sluggish machines. IT works to resolve the problems, but they are unable to find the root cause of the issues. Most users are instructed to restart their machines.

Day 50

Several employees contact IT complaining that their machines are freezing or unresponsive, while others are complaining that they are unable to access network resources and shared drives. IT begins investigating the issues but does not yet know the root cause of the problems.

Day 51

Ransomware messages appear on computers throughout your organization, and users report that they are unable to access their files. A message is displayed stating that all files have been encrypted and demanding payment of 1 Bitcoin per machine or entire network, valued at approximately $70.000 for the decryption key. The message also warns that the key will expire unless payment is received within 48 hours.

Day 52

Several media outlets begin reporting that Acme Inc is experiencing a ransomware attack. You have received multiple media inquiries asking you to comment on the ransomware incident. The media stories are gaining wide attention online and within social media platforms.

Day 53

Media reporting now indicates that Indigo Software is also experiencing a ransomware attack.

Discussion Questions

1. How would these incidents be assessed within your organization? Do you have defined cybersecurity incident severity levels and/or escalation criteria?
2. Do you have personnel tasked with incident response or a designated cyber incident response team within your organization?
   a. If so, what threshold must be reached for the cyber incident response personnel to be activated? Does this scenario reach that threshold?
   b. Who is responsible for activating the cyber incident response personnel and under what circumstances?
   c. What are the cyber incident response team/personnel's roles and responsibilities?
3. What internal and external notifications (e.g., to organizational leadership, users, customers, law enforcement, government partners) would you make?
4. Does your organization have a robust data recovery plan?
   a. Where are backups stored? Are they offline or online, stored in a secure location, or managed via a third party?
   b. Are the backups tested to ensure they work and are not corrupted or damaged?
   c. How far back do your backups cover?
   d. How are backups tested and ensured to be not infected with the same malware?
   e. How often is the data restoration processed exercised?

5. Do you pay the ransom?
    a. Who decides?
    b. What's your process for making the decision on whether to pay or not pay the ransom?
    c. What are the advantages/disadvantages of paying?
    d. What are the political ramifications if you decide to pay?
    e. What outside partners/entities do you need to contact if you decide to pay?

6. What capabilities and resources are required for responding to this incident?
    a. Who would you contact if you need additional assistance?
        i. DHS?
        ii. FBI?
        iii. Third-party vendor?
        iv. Mutual assistance organization?

7. What are your public affairs concerns?
    a. Who is responsible for coordinating the public message?
    b. Is this process a part of any established plan?
    c. How would your organization respond to the media reports?
    d. What information are you sharing with the public? Employees?
    e. Are public information personnel trained to manage messaging related to cyber incidents?
    f. Does your department have pre-drafted statements in place to respond to media outlets?

Appendix A: Case Studies

The following section includes background and example information related to cybersecurity threats and attacks, as well as relevant doctrine. Planners are encouraged to fill in highlighted fields with exercise specific information and include additional information as desired.

WannaCry: Worldwide Ransomware Attack (2017)

On May 12, 2018, one of the most notorious ransomware attacks began affecting systems across the globe. Experts estimate more than 300,000 systems have been affected by the variant known as WannaCry across the globe. A common way for this ransomware to spread is through standard file sharing technology, i.e. through vulnerabilities in Microsoft Windows Server Message Block (SMB). The vulnerability used in this attack was exploited to drop a file on the vulnerable system and then executed as a service, encrypting files (commonly used by Microsoft Office, databases, file archives, multimedia files, and various programming languages) with the .WNCRY extension.

In 2019, WannaCry is still affecting many organizations, particularly in the healthcare and manufacturing sectors. According to a research report from internet of things security company, Armis, WannaCry continues to be an active threat. Armis claims WannaCry was "reportedly responsible for 30% of all ransomware attacks worldwide in Q3 2018, and over 145,000 devices worldwide are still compromised".[1]

City of Atlanta: SamSam Ransomware Attack (2018)

In March 2018, the city of Atlanta, GA was hit by SamSam ransomware which infected the city's networks and encrypted at least one-third of its applications – including some which were customer-facing. The attackers demanded $52,000 in Bitcoin to restore the systems. Atlanta chose not to pay the ransom but instead spent money on improving cyber defenses. According to the latest 2019 data, Atlanta has spent over $2.6 million dollars on recovery efforts.

The SamSam ransomware variant has existed in various forms since 2015. Unlike most ransomware attacks which can be opportunistic in nature, SamSam's attackers often target specific organizations and use tools to scan victim's networks for vulnerability, including weak or easily guessed administrative passwords on common web servers, network equipment, and other internet-facing hardware and software. SamSam attackers typically do not rely on social engineering to access their targets.

According to a recent case study, "The SamSam ransomware encrypts files on both servers and workstations. In a typical attack, the ransomware message demands a certain payment in Bitcoin for each computer, or a larger amount to restore all of an organization's computers…"[2]

City of Baltimore – Robbinhood Ransomware Attack (2019)

On May 7, 2019, the city of Baltimore, MD was crippled by the Robbinhood ransomware – a file-locking variant - which encrypted hard drive data of city computers to prevent access to data. Although emergency services remained available, many systems, such as the city's water billing system, were dependent on inefficient manual workarounds. The attackers demanded around

[1] ("Wannacry Two Years Later: How Did We Get The Data?", 2019); (Seri, n.d.)

[2] (Rutenberg, 2018)

$75,000 to restore files. The FBI advised the city to not pay the ransom. Current estimates state that Baltimore will spend $10 million dollars on recovery efforts and will have lost $8 million dollars in payments the city could not process.

It appears the Robbinhood ransomware variant is new, though the way it works is not. Like SamSam ransomware, Robbinhood has targeted specific organizations. Recently the Baltimore Sun stated that, [according to cybersecurity researchers], …"RobbinHood could not have spread from machine to machine across a network on its own. Rather, the attackers would have needed to obtain access that would make them appear to be legitimate administrators, and then target individual victim computers." Analysis is still ongoing, but it's believed that Robbinhood is not connected to WannaCry's original Eternal Blue exploit.

Appendix B: Attacks and Facts

Distributed Denial of Service

Distributed Denial of Service (DDoS) attacks overload bandwidth and connection limits of hosts or networking equipment, specifically through a network of computers making excessive connection requests. DDoS attacks unfold in stages. First, a malicious actor infects a computer with malware that spreads across a network. This infected computer is known as the "master" because it controls any subsequent computers that become infected. The other infected computers carry out the actual attack and are known as "daemons." The attack begins when the master computer sends a command to the daemons, which includes the address of the target. Large numbers of data packets are sent to this address, where extremely high volumes (floods) of data slow down web server performance and prevent acceptance of legitimate network traffic. The cost of a DDoS attack can pose sever loss of revenue or reputation to the victim.

More information on DDoS attack possibilities within each layer of the OSI Model, as well as traffic types and mitigation strategies, can be found in the resource list below.

Additional Resources

- Understanding Denial-of-Service Attacks (https://www.us-cert.gov/ncas/tips/ST04-015)
- DDoS Quick Guide (https://www.us-cert.gov/sites/default/files/publications/DDoS%20Quick%20Guide.pdf)
- Guide to DDoS Attacks (https://www.cisecurity.org/wp-content/uploads/2017/03/Guide-to-DDoS-Attacks-November-2017.pdf)

Social Engineering

One of the most prominent tactics attackers use to exploit network and system vulnerabilities is social engineering–the manipulation of users through human interaction and the formation of trust and confidence to compromise proprietary information. Techniques for uncovering this information largely involve the use of phishing, i.e. email or malicious websites that solicit personal information by posing as a trustworthy source. Social engineering is effective for breaching networks, evading intrusion detection systems without leaving a log trail, and is completely operating system platform dependent. While technical exploits aim to bypass security software, social engineering exploits are more difficult to guard against due to the human factor. Organizations should take steps towards strengthening employee cybersecurity awareness training, to include training personnel to be cautious of suspicious emails, know where to forward them and keeping software and systems up-to-date.

Additional Resources

- Avoiding Social Engineering and Phishing Attacks (https://www.us-cert.gov/ncas/tips/ST04-014)
- The Most Common Social Engineering Attacks (https://resources.infosecinstitute.com/common-social-engineering-attacks/)

C.3 ChatRange Generated Playbook

ACME INC

Ransomware – Third Party Vendor

01.04.2024

NTNU

Handling Instructions

TLP: WHITE

The title of this document is Ransomware – Third party vendor Situation Manual. This document is unclassified and designated as "*Traffic Light Protocol (TLP): WHITE.* This designation is used when information requires support to be effectively acted upon, yet carries risks to privacy, reputation, or operations if shared outside of the organizations involved. Recipients may only share TLP:WHITE information with members of their own organization, and with clients or customers who need to know the information to protect themselves or prevent further harm. **Sources are at liberty to specify additional intended limits of the sharing: these must be adhered to.**

Exercise Overview

<table>
<tr><th>Exercise Name</th><th colspan="2"></th></tr>
<tr><td><b>Exercise Date, Time, and Location</b></td><td colspan="2">01.04.2024
Time (e.g. 9:00 a.m. – 15:00 p.m.)
NTNU Cyber Range</td></tr>
<tr><td rowspan="10"><b>Exercise Schedule</b></td><td><b>Time</b></td><td><b>Activity</b></td></tr>
<tr><td>09:00</td><td>Welcoming Remarks and Introductions</td></tr>
<tr><td>09:15</td><td>Exercise Briefing (Objectives, Rules of Engagement, etc.)</td></tr>
<tr><td>10:00</td><td>Module 1</td></tr>
<tr><td>12:00</td><td>Module 2</td></tr>
<tr><td>14:00</td><td>Debrief / Hotwash</td></tr>
<tr><td>15:00</td><td>Finish</td></tr>
<tr><td></td><td></td></tr>
<tr><td></td><td></td></tr>
<tr><td></td><td></td></tr>
<tr><td><b>Scope</b></td><td colspan="2">6 hours facilitated, discussion-based tabletop exercise</td></tr>
<tr><td><b>Purpose</b></td><td colspan="2">To examine the coordination, collaboration, information sharing, and response capabilities of Acme Inc in reaction to a ransomware incident with third party compromise by phishing.</td></tr>
<tr><td><b>Exercise Focus</b></td><td colspan="2">Identify, Protect, Respond, Recover</td></tr>
<tr><td><b>Objectives</b></td><td colspan="2">1. Discuss elements of Acme Inc's cybersecurity posture.
2. Examine Acme Inc's cybersecurity information sharing procedures and mechanisms.
3. Examine Acme Inc's cyber incident response plans or playbooks.</td></tr>
</table>

Exercise Name	
Threat or Hazard	Cyber
Scenario	A threat actor targets a third-party vendor through a phishing email as an entry point into Acme Inc networks/systems. Attackers cause computer latency and network access issues and install ransomware on Acme Inc computers.
Participating Organizations	IT-Management

General Information

Participant Roles and Responsibilities

The term *participant* encompasses many groups of people, not just those playing in the exercise. Groups of participants involved in the exercise, and their respective roles and responsibilities, are as follows:

Players have an active role in discussing or performing their regular roles and responsibilities during the exercise. Players discuss or initiate actions in response to the simulated emergency.
Observers do not directly participate in the exercise. However, they may support the development of player responses to the situation during the discussion by asking relevant questions or providing subject matter expertise.
Facilitators provide situation updates and moderate discussions. They also provide additional information or resolve questions as required. Key Exercise Planning Team members may also assist with facilitation as subject matter experts during the exercise.
Note-takers are assigned to observe and document exercise activities. Their primary role is to document player discussions, including how and if those discussions conform to plans, policies, and procedures.

Exercise Structure

This exercise will be a facilitated exercise. Players will participate in the following:

- Cyber threat briefing (if desired)
- Scenario modules:
  - **Module 1. – Third party vendor**
  - **Module 2 – Ransomware attack**
- Hotwash

Exercise Guidelines

- This exercise will be held in an open, no-fault environment. Varying viewpoints are expected.
- Respond to the scenario using your knowledge of existing plans and capabilities, and insights derived from your training and experience.
- Decisions are not precedent setting and may not reflect your organization's final position on a given issue. This exercise is an opportunity to discuss and present multiple options and possible solutions and/or suggested actions to resolve or mitigate a problem.
- There is no hidden agenda, and there are no trick questions. The resources and written materials provided are the basis for discussion.
- The scenario has been developed in collaboration with subject matter experts and exercise planners from your organization.
- In any exercise, assumptions and artificialities are necessary to complete play in the time allotted, to achieve training objectives, and/or account for logistical limitations. Please do not allow these considerations to negatively impact your participation in the exercise.

Exercise Hotwash and Evaluation

The facilitator will lead a hotwash with participants at the end of the exercise to address any ideas or issues that emerge from the exercise discussions.

Module 1:

Day 1

The Phobos Ransomware Group initiated a reconnaissance mission targeting Acme Inc. They identified a third-party vendor with weaker security protocols that frequently interacted with Acme Inc's systems. The aim was to use this vendor as a stepping stone to infiltrate Acme Inc.

Day 4

After gathering sufficient information, the threat group weaponized a phishing email specifically designed to mimic legitimate communication from Acme Inc to the third-party vendor. This email contained a malicious attachment, purportedly a document requiring urgent attention.

Day 7

The phishing email was delivered successfully to the third-party vendor. An employee, unaware of the malicious intent, opened the attachment, inadvertently executing the payload. This initial compromise gave the Phobos Ransomware Group unauthorized access to the vendor's network.

Day 11

Exploiting the compromised network, the threat group sought to escalate privileges and move laterally within the vendor's network to establish a foothold. Simultaneously, they began probing for vulnerabilities and connections to Acme Inc's network, aiming to infiltrate it.

Discussion Questions

1. How does Acme Inc. assess the cybersecurity posture of their third-party vendors?
    a. What criteria are used for this assessment?
    b. How often are these assessments conducted?
    c. Is there a process for ensuring continuous compliance with Acme Inc's security standards?
2. In terms of cybersecurity awareness, how does Acme Inc. ensure that its employees and those of its third-party vendors are adequately trained to recognize phishing attempts?
    a. What specific training modules are in place regarding phishing and social engineering attacks?
    b. How is the effectiveness of this training evaluated?
    c. Are there regular updates or refreshers provided based on emerging threats?
3. Upon learning about the initial compromise through a third-party vendor, what immediate steps should Acme Inc's IT-management take to assess the impact on its network?
    a. How should they identify whether the attack has propagated to their systems?
    b. What tools or processes are in place to detect unauthorized access or anomalies within their network?
4. What protocols does Acme Inc. have in place for responding to alerts of potential unauthorized access or compromises stemming from third-party connections?
    a. How are these incidents prioritized and escalated?
    b. What communication channels are established for timely and effective incident response?

5. How does Acme Inc. manage and secure the interfaces and connections with third-party vendors to prevent infiltration?
    a. What technology solutions (e.g., firewalls, VPNs, secure file transfer protocols) are utilized?
    b. Are there regular audits or reviews of these security measures?
6. How can Acme Inc. improve its detection capabilities to identify similar reconnaissance activities in the future?
    a. What indicators of compromise (IOCs) should be monitored?
    b. How can machine learning or AI be leveraged to predict and prevent such threats?
7. Given the initial compromise occurred through a phishing email, what improvements can Acme Inc. make to its email security posture?
    a. How effective are the current email filtering and scanning solutions?
    b. Is there a need for advanced threat protection features such as sandboxing or DMARC (Domain-based Message Authentication, Reporting, and Conformance)?

Module 2:

Day 1

Having infiltrated Acme Inc's network through the third-party vendor, the Phobos Ransomware Group installed the ransomware on key systems. This action was meticulously planned to occur during off-peak hours to avoid immediate detection.

Day 3

The ransomware established command and control channels back to the threat actors, allowing them to navigate freely and manipulate Acme Inc's systems. At this stage, the ransomware began encrypting critical files and databases.

Day 6

Actions on objectives were executed as the Phobos Ransomware Group deployed the ransomware across Acme Inc's network, targeting data servers, backup systems, and operational technology. A ransom note was generated on affected systems, demanding payment in cryptocurrency.

Day 9

The impact of the ransomware deployment became fully apparent. Critical operations were halted, leading to significant financial and reputational damage for Acme Inc. The company initiated their incident response protocol, marking the start of their efforts to mitigate the impact of the ransomware attack

Discussion Questions

1. Upon discovery of the ransomware deployment, what are the first actions Acme Inc.'s IT-management team should take to contain the spread?
    a. What specific systems or network segments should be isolated?
    b. How can the team identify and shut down the command and control channels?
2. How does Acme Inc.'s incident response plan address ransomware attacks specifically?
    a. Are there predefined steps for responding to encryption-based attacks?
    b. What is the protocol for deciding whether to pay a ransom or not?
3. What measures are in place for maintaining business continuity in the event of critical systems being encrypted by ransomware?
    a. How are essential services and operations maintained?
    b. What backup and recovery processes are activated?
4. How does Acme Inc. communicate with stakeholders (e.g., employees, customers, partners) during a ransomware attack?
    a. What information is shared and at what intervals?
    b. How is customer data privacy and regulatory compliance maintained during such incidents?
5. What role does Acme Inc.'s cybersecurity information sharing mechanisms play in responding to the ransomware attack?
    a. How is relevant threat intelligence gathered and utilized?
    b. Is information about the attack shared with external cybersecurity organizations or industry groups?
6. After the ransomware attack, how does Acme Inc. go about identifying the vulnerabilities that were exploited?

   a. What tools or processes are used to conduct post-incident forensics?
   b. How are these findings integrated into improving the cybersecurity posture?

7. In the aftermath of the ransomware deployment, what steps does Acme Inc. take to review and update its cyber incident response playbook?
   a. How are lessons learned documented and shared within the organization?
   b. What changes are made to response strategies and recovery plans?

Appendix A: Case Studies

The following section includes background and example information related to cybersecurity threats and attacks, as well as relevant doctrine. Planners are encouraged to fill in highlighted fields with exercise specific information and include additional information as desired.

The WannaCry Ransomware Attack

The WannaCry ransomware attack, a global cyberattack that occurred in May 2017, was orchestrated by the WannaCry ransomware cryptoworm. This malicious software targeted computers operating on the Microsoft Windows system, encrypting data and demanding ransom payments in Bitcoin. The attack utilized EternalBlue, an exploit initially developed by the United States National Security Agency (NSA) for Windows systems. The exploit was leaked by a group called The Shadow Brokers a month before the attack. Despite Microsoft releasing patches to address the vulnerability, many organizations were affected due to delayed or incomplete patch installations, leading to widespread consequences.

The attack commenced on 12 May 2017 at 07:44 UTC and was halted a few hours later at 15:03 UTC when a kill switch was activated by Marcus Hutchins, preventing further encryption of infected computers. The impact of the attack was significant, affecting over 300,000 computers in 150 countries and resulting in damages estimated in the hundreds of millions to billions of dollars. Initially suspected to be linked to North Korea, the United States and United Kingdom later confirmed North Korea's involvement in the attack. Furthermore, a new variant of WannaCry emerged in August 2018, causing the temporary shutdown of several chip-fabrication factories at the Taiwan Semiconductor Manufacturing Company (TSMC), affecting thousands of machines in their facilities.

Taiwan Semiconductor Manufacturing Company (TSMC) Ransomware Attack Timeline

In August 2018, a new variant of the WannaCry ransomware attack forced Taiwan Semiconductor Manufacturing Company (TSMC) to temporarily shut down several of its chip-fabrication factories. The worm spread onto 10,000 machines in TSMC's most advanced facilities.

WannaCry is a ransomware cryptoworm that targets computers running the Microsoft Windows operating system by encrypting data and demanding ransom payments in the Bitcoin cryptocurrency. The attack began at 07:44 UTC on 12 May 2017 and was halted a few hours later at 15:03 UTC by the registration of a kill switch discovered by Marcus Hutchins. The attack affected more than 300,000 computers across 150 countries, with damages ranging from hundreds of millions to billions of dollars. Initial evaluations suggested the attack originated from North Korea or agencies working for the country.

The CryptoLocker Ransomware Attack

The CryptoLocker ransomware attack was a significant cyberattack that occurred from 5 September 2013 to late May 2014. This attack utilized a trojan that targeted computers running Microsoft

Windows and propagated via infected email attachments and the Gameover ZeuS botnet. Once activated, the malware encrypted specific types of files using RSA public-key cryptography and demanded a ransom payment in Bitcoin or a pre-paid cash voucher. If the deadline was not met, the malware threatened to delete the private key, offering decryption at a higher price. Despite the malware being easily removed, the encrypted files remained inaccessible, leading to a dilemma for victims on whether to pay the ransom or not.

The Rise of Ransomware

Ransomware, as demonstrated by the CryptoLocker attack, has become a prevalent form of malware that blocks access to personal data until a ransom is paid. This type of cryptovirological malware encrypts victim's files, making them unreachable without the decryption key. Ransomware attacks are often carried out through Trojans disguised as legitimate files or email attachments, with some instances like the WannaCry worm spreading automatically between computers. The success of ransomware attacks, such as CryptoLocker, highlights the challenges of tracing and prosecuting perpetrators due to the use of difficult-to-trace digital currencies. The rise of ransomware scams internationally underscores the importance of robust cybersecurity measures to mitigate the impact of such attacks.

Hillary Clinton's 2016 Presidential Campaign Spear Phishing Attacks

In 2016, during Hillary Clinton's presidential campaign, the campaign fell victim to targeted spear phishing attacks. These attacks involved the Russian government-run Threat Group-4127 (Fancy Bear) targeting over 1,800 Google accounts associated with the campaign. The attackers used the accounts-google.com domain to send deceptive emails to campaign staff, attempting to steal sensitive information and gain unauthorized access to their accounts. This sophisticated form of phishing, known as spear phishing, leveraged personal information to increase the chances of success. Despite the campaign's efforts to enhance cybersecurity, these attacks posed a significant threat to the integrity of the campaign's digital infrastructure.

Impact and Response

The spear phishing attacks on Hillary Clinton's 2016 presidential campaign highlighted the vulnerability of political campaigns to cyber threats. The compromised Google accounts could have potentially exposed sensitive information and disrupted campaign operations. In response to these attacks, cybersecurity measures were reinforced, and staff were educated on the importance of identifying and mitigating phishing attempts. The incident underscored the need for continuous vigilance and robust cybersecurity protocols in political campaigns to safeguard against malicious cyber activities that aim to undermine the democratic process.

Threat Group-4127 (Fancy Bear) Targeting Google Accounts Through Phishing

Threat Group-4127, also known as Fancy Bear, is a Russian cyber espionage group associated with the Russian military intelligence agency GRU. Fancy Bear has been involved in state-sponsored cyberattacks and decryption of hacked data, targeting government, military, and security organizations, especially Transcaucasian and NATO-aligned states. The group has used zero-day exploits, spear phishing, and malware to compromise targets, including the Democratic National Committee during the 2016 US presidential elections. Fancy Bear has a history of promoting the political interests of the Russian government through cyber operations.

Phishing, a form of social engineering and scam, has become increasingly sophisticated and is the most common type of cybercrime as of 2020. Attackers deceive individuals into revealing sensitive information or installing malware through phishing attacks. Fancy Bear has utilized spear phishing, a targeted form of phishing that uses personalized emails to trick specific individuals or organizations. In one instance, Threat Group-4127 targeted Hillary Clinton's 2016 presidential campaign with spear phishing attacks on over 1,800 Google accounts using a deceptive domain. The group's sophisticated tactics highlight the importance of cybersecurity awareness and measures to combat phishing attacks.

Appendix B: Attacks and Facts

Phishing

Phishing is a form of social engineering and scam where attackers deceive people into revealing sensitive information or installing malware such as ransomware. Phishing attacks have become increasingly sophisticated and often transparently mirror the site being targeted, allowing the attacker to observe everything while the victim is navigating the site, and transverse any additional security boundaries with the victim. As of 2020, it is the most common type of cybercrime, with the FBI's Internet Crime Complaint Center reporting more incidents of phishing than any other type of computer crime. Measures to prevent or reduce the impact of phishing attacks include legislation, user education, public awareness, and technical security measures. The importance of phishing awareness has increased in both personal and professional settings, with phishing attacks among businesses rising from 72% to 86% from 2017 to 2020.

Additional Resources

- Phishing Attack Prevention: How to Identify & Avoid Phishing Scams (https://www.occ.gov/topics/consumers-and-communities/consumer-protection/fraud-resources/phishing-attack-prevention.html)
- How to Recognize and Avoid Phishing Scams | Consumer Advice (https://consumer.ftc.gov/articles/how-recognize-and-avoid-phishing-scams)
- Teach Employees to Avoid Phishing – CISA (https://www.cisa.gov/secure-our-world/teach-employees-avoid-phishing)

Ransomware

Ransomware is a type of cryptovirological malware that permanently blocks access to the victim's personal data unless a ransom is paid. While some simple ransomware may lock the system without damaging any files, more advanced malware uses a technique called cryptoviral extortion. It encrypts the victim's files, making them inaccessible, and demands a ransom payment to decrypt them. In a properly implemented cryptoviral extortion attack, recovering the files without the decryption key is an intractable problem, and difficult-to-trace digital currencies such as paysafecard or Bitcoin and other cryptocurrencies are used for the ransoms, making tracing and prosecuting the perpetrators difficult. Ransomware attacks are typically carried out using a Trojan disguised as a legitimate file that the user is tricked into downloading or opening when it arrives as an email attachment. However, one high-profile example, the WannaCry worm, traveled automatically between computers without user interaction.

Additional Resources

- Prevent and Respond to Ransomware Attacks – CISA (https://www.cisa.gov/stopransomware)
- Ransomware - What It Is & How to Prevent It – McAfee (https://www.mcafee.com/enterprise/en-us/security-awareness/ransomware.html)
- How to Protect Your Networks from Ransomware – FBI (https://www.fbi.gov/file-repository/ransomware-prevention-and-response-for-cisos.pdf)
- Ransomware Guidance and Resources - National Cyber Security Centre UK (https://www.ncsc.gov.uk/guidance/mitigating-malware-and-ransomware-attacks)

Appendix D

Expert Reviewer Profiles

D.1 Description

This is an overview of the background of each expert reviewer used in ChatRange's second case study. The participant's names and current employers have been anonymized for privacy, and this only shows a summary of each participant. This also includes the textual expert review each participant gave in the study. All numbered statistics can be found in appendix **??**.

D.2 Reviewer 1

D.2.1 Current Role

IT Security

D.2.2 Background

Area	Sector	Years
Security	Education	5
IT	Military	15

D.2.3 Review

Playbook 1

For Realism, the TLP used in this scenario is based on the old version of TLP, and seems to use the text for tlp:amber but the TLP used is TLP:WHITE. It overall seems like a good and realistic exercise. There are good questions that should be answered by an organization, and they emphasize the use existing plans which is sometimes ignored when an incident occurs. You never get to test your plans if you do not use them. They explain the different attacks in the appendix, so that anyone with or without experience can follow it, which is good. The appendix also has real life examples that are similar, so that you can kill the discussion if this could ever happen in real life. It does not explain how many players of the different roles, or the experience needed for facilitators an efficient playthrough.

Playbook 2

For Realism, the TLP used in this scenario is based on the old version of TLP, and seems to use the text for tlp:amber but the TLP used is TLP:WHITE. It overall seems like a good and realistic exercise. There are good questions that should be answered by an organization, and they emphasize the use existing plans which is sometimes ignored when an incident occurs. You never get to test your plans if you do not use them. They explain the different attacks in the appendix, so that anyone with or without experience can follow it, which is good. The appendix also has real life examples that are similar, so that you can kill the discussion if this could ever happen in real life. It does not explain how many players of the different roles, or the experience needed for facilitators an efficient playthrough. The overall scenario seems repetitive, which shouldn't be an overall problem, but can be somewhat demotivating as a player.

D.3 Reviewer 2

D.3.1 Current Role

Management

D.3.2 Background

Area	**Sector**	**Years**
IT Management	Government	Over 10 years

D.3.3 Review

Playbook 1

It was easy to follow the playbook. The order seems natural, and I felt a sense of urgency when I read the scenarios. I think the best part was the discussion points. They helped me reflect, and I believe it would be possible to come up with good measures based on the discussions. As a person with Norwegian as my main language, it would be useful to have the playbook presented in Norwegian, but this is just a detail.

Playbook 2

The exercises are experienced as well-organized, they feel realistic and feasible. Part one seems a lot more technical than the other part. In part one, I would need to involve more technical resources than just the management team. Part two seems to me like a task that only the management team could have discussed. It is very nice that the main questions are followed up by sub-questions, and that these are not too many or too detailed so that it is possible to get through, but still have good and useful discussions.

D.4 Reviewer 3

D.4.1 Current Role

Security Management

D.4.2 Background

Area	Sector	Years
IT Security	Government	4
IT Security	Private	6

D.4.3 Review

Playbook 1

The exercise is well structured and the information is presented in a detailed and consumable way. The level of detail seems to be just right, although some of the first questions may be challenging for some IT-managers to answer. I would suggest adjusting the scenario to target both a mix of technical personnel and IT-managers to increase the learning experience as IT-managers does not always have the complete picture on a technical level.

Playbook 2

The exercise is well structured and the information is presented in a detailed and consumable way. Some of the questions are very detailed and should be reconsidered if the exercise is exclusively for IT-management. I would suggest adjusting the scenario to target both a mix of technical personnel and IT-managers to increase the learning experience as IT-managers does not always have the complete picture on a technical level. Or to split the scenario in two, one for a table top for technical personnel to discuss if the right security measures are in place from a technical perspective, and one for IT-management.

NTNU

Norwegian University of
Science and Technology