PROJECT STIX

Research Report

Subjects: Cyber Incidents, the maritime sector and related technologies

Project Group

Robert Rachita (4859367)
Atanas Hristov (4785002)
Christopher Sulistiyo (4850025)
Stefan Untura (4839161)
Victor Tromp (4922972)
Line Amini Kaveh (4929284)

Client

Sietse Dijkstra
Stephen McCombey

Version 1.0 – 25/03/2022

Acknowledgements

We would like to thank the following people for their contribution to the research and preparation of this report:

Sietse Dijkstra and Stephen McCombie for providing the necessary background knowledge, as well as sharing their ideas for improving the web application. Rene Laan and Bert Meijerink for the support, resources and extensive feedback given throughout the entire project as well as the research module.

Version Management

Version Number	Date	Alterations
1.0	25.03.2022	First Draft

Remarks

Due to the lack of an overarching question, to combine all relevant research results, this document assumes a different layout than a standard research report. While all the different results are explored to provide background knowledge regarding the maritime sector, related cyber security threats and potential technologies to be used, they are dealt with in a way that the outcomes were difficult to combine to serve in answering a singular question. Thus, it was decided to maintain four main questions that are either explained directly or via multiple sub questions that build up to the conclusion.

Given the diversity in topics, this is easier to understand for the reader and the chapters surrounding the actual research are split up as needed to ensure the details regarding the research process and outlines are specific to each question.

Table of contents

Executive Summary

Background

This research was conducted to support the development of a web application that will help companies in identifying maritime cyber-threats on a global scale. This research is part of the "STIX"-web application project, which is based on an earlier project iteration, with particular focus on increased scalability and security.

Methodology

This document is comprised of several research questions, all focused on gathering qualitative data by means of desk research, that may be expanded on if required. The researchers have ensured that the information was sourced from reputable outlets and in case of the STIX methodology, pulled data directly from official documentation.

Key Findings

- Creating a web application tailored to the STIX methodology ensures accessibility for users with limited tech literacy
- STIX/TAXII methodology can be adapted/utilized to serve categorization of threats in underrepresented fields, such as the maritime sector, while maintaining standardized formatting of CTI
- ASP .NET (Core) is a clear winner when it comes to performance and checking off the requirements
- Due to the un-standardised manner in which the data from incidents is reported and due to clear performance difference, MongoDB NoSQL database will be used to store all the data required
- A maritime cyber security incident can be identified as any disruption of the ports or ships work or on-board devices caused by malware, misuse, etc.
- Maritime cyber security incidents can be caused both by digital/remote attackers and on-board personal (intentionally or unintentionally)
- The most common cyber security attack tactics are as follows: (Malware, Social Engineering, Phishing, Water holing, Port scanning, Build-in software weakness, Third party contribution, Brute force, DDoS, Spear-phishing, Subverting the supply chain)

Key Recommendations

- Study the STIX / TAXII specification documents more thoroughly (see limitations research question 3).
- Multiple formal documents such as research reports and articles should be documented and discussed.
- Interviewing multiple experts in the field of maritime cybersecurity.
- Collaborate with other research teams working on the same topic.
- Build an ASP.NET Web application tied with a NoSQL MongoDB database

1 Introduction

1.1 Background

As introduced to the team by Mr. Stephen McCombie, also referred to as *the client*, who will be in constant communication with the team receiving updates and supplying feedback and clarifications as the project takes form, the developing team must improve, redesign, and introduce a multitude of new functionalities for an existing web application. The web application must recognize, sort and store vulnerabilities/cyber-attacks directed to components of the maritime field such as boats, ports, etc.

Due to the continuous development of technology globally and the outdated technologies currently present in the maritime field, the field has become increasingly targeted and the idea of a product that may simplify the process of threat recognition, sorting and storing has come to life.

The project will be executed by students of the NHL Stenden University of Applied Sciences, which is based in the north of the Netherlands and offers a large variety of courses in both Dutch and English. The schooling has a heavy emphasis on group and practical project-based work, supported by a thorough theoretical education.

1.2 Research Reason

By doing this research, the team gains insight into the methodologies and standards needed to develop a web application that allows for easy communication and documentation of maritime security incidents. Furthermore, the scalability of existing products and areas in need of change are examined in detail, to provide insight into areas in need of change.

Additionally, the team hopes to gain insight into cybersecurity incidents, the methodology used to detect and identify the potential threats and what are the distinct types of the cyberthreat categorization. The knowledge and information gained as a result of this research, in turn will help the team develop an efficient way to classify, separate and present the distinct type of cyberthreats, which means setting up and using the proper database technology, creating a visual graphic representation on the website and enhance the overall website's security level.

To successfully create a logically structured platform for tracking and modelling cyber incidents, there needs to be a deeper understanding of the different parts at work as none of the project members responsible for the application build and design are yet sufficiently proficient in the area of expertise.

1.3 Research Goal

The aim of this research is to gain a fundamental understanding of the STIX methodology and its accompanying systems in order to develop a web application incorporating the STIX methodology for maritime cyber-threats. Furthermore, this research aims to determine which actions must take place in order to scale the web application further, possibly to a global scale.

1.4 Reading Guide

The executive summary gives a concise overview of the document's contents, starting with the background and reason for conducting the research. It also contains the key findings, giving the reader an immediate insight into what to expect.

The introduction chapter is split into several smaller parts. The first of which, the background chapter, describes the circumstances under which the research was commissioned and who is conducting it. Moving to the reasoning for the research and goal, the former gives insight into why this research is valuable and the research goal defines the intended outcome. Following that is the reading guide, which gives short descriptions of each sub chapter. Acting as a sort of glossary is the definition of terms, by giving concise descriptions of the given terminology. Lastly, the introduction includes a list of objectives that the research results are meant to achieve.

Going deeper into how the research is conducted, is the content of the methodology chapter, where the research problem is defined including its sub questions. Furthermore, said questions are listed in a table with the chosen research method, detailing whether quantitative research, qualitative research or a combination of both is used. The means of acquiring data is then further detailed, as well as how said data will then be analyzed and processed. Finally, the validity, reliability and potential shortcomings of the means of gathering data as well as the data itself are discussed.

The results chapter contains all research results presented in a factual and unbiased manner, to be further investigated and theorized upon in the following chapters.

The conclusion answers the research's main problem, based on the information procured and detailed in the preceding chapter.

In the discussion chapter, recommendations based on the conclusion and research results are made.

All sources referenced within this document are listed in the bibliography in alphabetical order and styled in the APA format. Additional sources and information referred to within the results are linked within the Appendix.

1.5 Definition of Terms

1.5.1 Research Question 1:

Framework: a tool that provides ready-made components or solutions that are customized in order to speed up development

Containerization: Containerization is the packaging together of software code with all it's necessary components like libraries, frameworks, and other dependencies so that they are isolated in their own "container."

web application: an application program that is stored on a remote server and delivered over the Internet through a browser interface

PHP: is a widely-used open source general-purpose scripting language that is especially suited for web development and can be embedded into HTML

Laravel: open source PHP framework

data visualization: the graphical representation of information and data

Node.js: an open source development platform for executing JavaScript code server-side

Asp.Net(Core): a cross-platform, high-performance, open-source framework for building modern, cloud-enabled, Internet-connected apps

command-line tool: a tool that processes commands to a computer program in the form of lines of text.

server-side scripting: a technique used in web development which involves employing scripts on a web server which produces a response customized for each user's (client's) request to the website
event-driven architecture: a software architecture and model for application design

asynchronous I/O: a form of input/output processing that permits other processing to continue before the transmission has finished cross-platform: is the practice of deliberately writing software to work on more than one platform

The Common Dialect Runtime: programming that manages the execution of programs written in any of several supported languages, allowing them to share common object-oriented classes written in any of the languages.

MVC: Model-View-Controller is a pattern in software design commonly used to implement user interfaces, data, and controlling logic

ASP.NET: open-source, server-side web-application framework designed for web development to produce dynamic web pages

Swagger: is an open source set of rules, specifications and tools for developing and describing RESTful

API: stands for application programming interface, which is a set of definitions and protocols for building and integrating application software

Unity: a theoretical language which focuses on what, instead of where, when or how

Serilog: a portable and structured logging framework to record diagnostic logs into files, console and SQL/NoSQL databases

IdentityServer04AspNetIdentity: ASP.NET Core Identity is a membership system that adds login functionality to ASP.

Nswag: toolchain for .Net, .Net Core, Web Api and other technologies/frameworks.

Callback: any reference to executable code that is passed as an argument to other code

Asynchronous programming: a form of parallel programming that allows a unit of work to run separately from the primary application thread

Thread: a single sequential flow of control within a program

thread pool: a software design pattern for achieving concurrency of execution in a computer program.

IIS server: Internet Information Services (IIS) is a flexible, general-purpose web server from Microsoft that runs on Windows systems to serve requested HTML pages or files

Kernel: a computer program at the core of a computer's operating system and generally has complete control over everything in the system

Queries: request of information, mostly from databases.

JSON: A specific data format, used by applications.

Micro services: a specialization of an implementation approach for service-oriented architectures (SOA) used to build flexible, independently deployable applications.

Cluster model: the task of grouping a set of objects in such a way that objects in the same group (called a cluster) are more similar (in some sense) to each other than to those in other groups (clusters).

load balancer: a piece of hardware (or virtual hardware) that acts like a reverse proxy to distribute network and/or application traffic across different servers

AWS: Amazon Web Services

ELB: Elastic Load Balancing (ELB) automatically distributes incoming application traffic across multiple targets and virtual appliances in one or more Availability Zones (AZs)

NGINX: open source Web server software that also performs reverse proxy, load balancing, email proxy and HTTP cache services

Istio: a service mesh—a modernized service networking layer that provides a transparent and language-independent way to flexibly and easily automate application network functions

Apache HTTP Server: a free and open-source web server that delivers web content through the internet

F#: a functional-first, general purpose, strongly typed, multi-paradigm programming language that encompasses functional, imperative, and object-oriented programming methods

1.5.2 Research Question 2

a tried-and-true technology: proven in the space, already used in large companies.

MongoDB: a technology derivative from MySQL

Express: Express is a node js web framework

React: React. js is an open-source JavaScript library that is used for building user interfaces specifically for single-page applications.

Node: Node js an open-source development platform for executing JavaScript code server-side

MySql: programming technology for database.

1.5.3 Research Question 3:

API: An application programming interface (API) is a connection between computers or between computer programs. It is a type of software interface, offering a service to other pieces of software.

API Root: The API Root resource contains links to other resources available in the API and allows users to discover them

Authentication: refers to the process of proving an assertion (in this document it refers to proving a user's identity and authorization level for server access)

Authorization: specifies the access rights to certain resources

Covid-19: The corona virus is an infectious disease, which has been a major worldwide health concern since March of 2020

CTI: In this context CTI refers to Cyber threat Intelligence

HTTPS: Hypertext Transfer Protocol Secure (HTTPS) is an extension of the Hypertext Transfer Protocol (HTTP). It is used for secure communication over a computer network and is widely used on the Internet.

JavaScript: A script language used for dynamic HTML in web browsers

JavaScript Object Notation (JSON): a compact data format for data exchange between applications

MobX: A library used for state management

Payload-agnostic: Meaning that the service is unaware of the content it is serving

Publish-subscribe model: A data transmission model where publishers can share content with subscribing clients

React: A JavaScript library that simplifies building user interfaces

State management: the management of one or more user interface objects

STIX: Structured Threat Information eXpression, is a language and serialization format used to exchange cyber threat intelligence (CTI).

TAXII: TAXII is a protocol used to exchange cyber threat intelligence (CTI) over HTTPS. TAXII enables organizations to share CTI by defining an API that aligns with common sharing models. relating to it.

TypeScript: Adds optional types, classes and modules to JavaScript and is supports tools for JavaScript applications with lots of code

STIX Core Objects: Any SDO, SCO, or SRO.

STIX Domain Objects (SDO): Higher Level Intelligence Objects that represent behaviours and constructs that threat that analyst would typically create or work with while understanding the threat landscape.

STIX Cyber-observable Objects (SCO): Objects that represent observed facts about a network or host that may be used and related to higher level intelligence to form a more complete understanding of the threat landscape.

STIX Relationship Objects (SRO): Objects that connect STIX Domain Objects together, STIX Cyber-observable Objects together, and connect STIX Domain Objects and STIX Cyber-observable Objects together to form a more complete understanding of the threat landscape.

STIX Meta Objects (SMO): A STIX Object that provides the necessary glue and associated metadata to enrich or extend STIX Core Objects to support user and system workflows.

STIX Bundle Object: An object that provides a wrapper mechanism for packaging arbitrary STIX content together.

Webpack: a module bundler, that is optimized for JavaScript files.

1.5.4 Research Question 4:

Cyberattack: any offensive manoeuvre that targets computer information systems, computer networks, infrastructure, or personal computer devices and is driven by means of obtaining personal or financially sensitive data

Cybersecurity incident: a type of situation where the IT system is impacted in a way that it can no longer operate as intended. It should be taken note that cyber incident differs from a “cyberattack” in that cyber-attack is generally considered the precursor to a cyber incident. A cyber incident is declared when a cyber-attack has actually impacted the confidentiality, integrity, or availability of an IT system

Cyber threat: the possibility of malicious attempt to damage or disrupt a computer network or system. The definition is almost similar to cyberattack, and any cyber incident can be designed as a cyberattack, the major difference is that a threat can be intentional or unintentional, whereas an attack is intentional

Cybercrime/ computer-oriented crime: any illegal and criminal activities carried out by means of technology by using computers and of the internet

Cryptovirology: a field that studies how to use cryptography to design powerful malicious software

Cybercriminal: a person/ individual who engages in criminal activities on digital system or networks by means of technology like computer or the internet with the intention of stealing sensitive company information or personal data, and generating profit

Hacker: computer programmers who use their digital skills to breach digital system or to gain unauthorized access to data. Not all hackers are necessarily bad actors; some, known as ethical hacker, use their knowledge to improve any security practices

Cracker/ Black Hats: someone who unethically exploits highly sensitive information and uses flaws to his/her own advantages

Computer virus: a type of computer program or malware that, when executed, attaches to another computer program (like document) which can spread and replicate itself by modifying other computer programs and inserting its own code after a person first runs it on their system

Malware (malicious software): software that is intentionally and specifically designed to disrupt, interfere, damage, or gain unauthorized access to a computer system, server, client, or computer network and that can be used to commit cybercrime (as by revealing passwords, PINs, and other sensitive data)

Ransomware: a type of malicious software designed to perpetually block access to a computer system or threatens to publish victim’s data unless a sum of money is paid

Trojan horse virus: a type of malicious code or software that looks legitimate but can take control of target’s computer and perform some innocuous function

Denial of Service (DoS): a type of cyber-attack in which a malicious actor aims to render a computer or other device unavailable to its intended users by interrupting the device's normal functioning.

Phishing: a type of social engineering where an attacker sends a fraudulent message designed to trick a person

Baiting: a type of social engineering attack where a scammer uses a false promise to lure a victim into a trap which may steal personal and financial information or inflict the system with malware

Spoofing attack: is a situation which a person or program successfully identifies as another by disguising and falsifying data, a communication, or identity to gain an illegitimate advantage

Social Engineering: the use of broad deception tactics to manipulate individuals into divulging confidential or personal information to be acquired that may be used for fraudulent purposes by hackers

OT (Operational Technology): is a category of hardware and software that monitors and controls how physical devices perform

IT (Information Technology): the use of technology involving the development, maintenance, and use of computer systems, software, and networks for the processing and distribution of data

IoT (Internet of Things): is the integration of people, processes and technology with connectable devices and sensors to enable remote monitoring, status, manipulation and evaluation of trends of such devices

GNNS (Global Navigation Satellite System): any satellite constellation that provides geo-spatial positioning, navigation, and timing (PNT) services on global or regional basis to many devices autonomously, allowing electronic devices with the appropriate receivers to determine their precise location on the surface of the Earth

AIS (Automatic Identification System): is an automatic tracking system that uses transceivers on ships and is used by VTS. It provides ship's position and movement to surrounding vessels via VHF, primarily for collision avoidance

IMO (International Maritime Organization): is a special agency of the UN responsible for regulating shipping

BIMCO (Baltic and International Maritime Council): one of the largest of the international shipping associations representing ship owners

ENISA (European Union Agency for Cybersecurity): is an agency contributes to EU cyber policy, enhances the trustworthiness of ICT products, services and processes with cybersecurity certification schemes, cooperates with Member States and EU bodies, and helps Europe prepare for the cyber challenges of tomorrow

MUNIN (Maritime Unmanned Navigation through Intelligence in Networks): is a collaborative research project, co-funded by the EC, and aims to develop and verify a concept for autonomous ship

NORMA (Norwegian Maritime Cyber Resilience Centre): is an organization that provides an intelligence and information-sharing service and an incident response and crisis-support service

SCADA (Supervisory Control and Data Acquisition): is a control system architecture comprising computers, networked data communications and graphical user interfaces for high-level supervision of machines and processes

SART (Search and Rescue Transponder)/ Transponder: is a self-contained, waterproof transponder intended of emergency use at sea

VTS (Vessel Traffic Services): A vessel traffic service is a marine traffic monitoring system established by harbour or port authorities, similar to air traffic control for aircraft.

VHF (Very High Frequency): Very high frequency is the ITU designation for the range of radio frequency electromagnetic waves from 30 to 300 megahertz, with corresponding wavelengths of ten meters to one meter.

Gyro compasses: It is used to find the right direction. Unlike a magnetic compass, the gyrocompass is not interfered with by an external magnetic field. It is used to determine the correct north position, which is also the Earth's axis of rotation to provide a stable directional source.

Radar: Marine vessels rely on an S-band and X-band radar system for navigation as it can detect targets and display information on the screen such as the ship's distance from the ground, any floating objects (island, rocks, iceberg, etc.), other vessels and obstacles to avoid collision. It is a rotating antenna that detects the surrounding area of the ship.

Autopilot: The bridge plan is filled with equipment and instruments used for navigation. Autopilot is considered one of the most effective bridge navigation aids as it helps the human operator steer the vessel by holding the steering in autopilot mode, allowing them to focus on the broad aspects of the operation. It is a combination of hydraulic, mechanical and electrical systems and is used to control the ship's steering system from a remote location (navigational bridge).

ARPA: An automatic radar plotting aid displays the location of the ship and other vessels nearby. The radar displays the location of nearby ships and selects a course for the ship, avoiding any type of collision. This navigation equipment on the bridge constantly monitors the environment of the ship and in this case automatically determines the number of targets; ships, boats, stationary or floating objects, etc., and plot their speed and course accordingly.

Speed & Distance Log Device: This bridge equipment on a ship is used to measure the speed and distance the ship has travelled from a given point.

Echo sounder: There are many modern shipborne navigation instruments on board and the echo sounder is one instrument that has been in use for nearly 100 years. It is used to measure the depth of water below the bottom of a ship using sound waves, which work on the principle of sound wave transmission, and a sound pulse that bounces off a reflective layer and returns as an echo to the source.

Information technology systems: This system focus on the use of data as information.

Operational technology systems: This system focus on the use of data to control or monitor physical processes.

1.6 Objectives

The objectives of this research are to:

- Understand the STIX methodologies inner workings
- Explore how the STIX standards are commonly implemented
- Understand the utilization of TAXII protocol
- Understand what data modelling tools are available and how they operate
- Explore the state of the current web application
- Determine what needs to be done to offer a product with increased scalability
- Comprehend what is a cyber incident and how do they work
- Be conscious of how to identify a potential cyberattack
- Understand and be acquainted with the types of cybersecurity incidents

2 Methodology

2.1 Research Questions

1. What are the most suitable web app development frameworks for the Stix Database Platform that are easy to build, maintain, secure, offer high performance and allow containerization?

2. What database technology or combination of technologies would be the fastest and most suitable for the STIX platform, executing each query within less than 10 milliseconds?

3. How can the STIX language be utilized to serve (maritime) cyber security needs?
   - What is STIX?
   - What are the (cyber security) needs of the stakeholders?
   - What proficiencies are needed to use the given tools in an alternative context?
   - What are the current developments in the (maritime) security sector?

4. What is a maritime cybersecurity incident, how to detect and identify potential threats, and what are different types of categorizations?
   - What is a maritime cybersecurity incident?
   - What is the purpose of maritime cyber incident?
   - What are the risks of maritime cybersecurity incident?
   - How susceptible is maritime industry to a cyberattack?
   - What is the biggest threat in maritime cybersecurity?
   - What are the characteristics of maritime cyber incident? What makes something a maritime cybersecurity incident?
   - What are the most common maritime cyberattack tactics?
   - How to identify a potential cybersecurity incident?
   - What is a cybersecurity incident categorization?
   - What is the purpose of cybersecurity incident classification?
   - How many incident categories are there?
   - Can a cyber incident belong to more than one category?
   - What is the categorization of the cyber incidents according to conventional taxonomy (NCSC: National Cyber Security Centre)?
   - What are ways to categorized cybersecurity incidents? Can they be categorized in one way?
   - Is there a type of prioritisation for categorizing the cyber incidents?

2.2 Research Design

The following research will use mixed methods to answer the research questions. The following table summarises the methods used to answer each question:

Research Question	Method Used to Answer Question
1. What are the most suitable web app development frameworks for the Stix Database Platform that are easy to build, maintain, secure, offer high performance and allow containerization?	Desk research
2. What database technology or combination of technologies would be the fastest and most suitable for the STIX platform, executing each query within less than 10 milliseconds?	Desk research
3. How can the STIX language be utilized to serve (maritime) cyber security needs?	Desk research
4. What is a maritime cybersecurity incident, how to detect and identify potential threats, and what are different types of its categorization?	Desk research

2.3 Instruments

During this research phase only qualitative data was collected, as all research teams gathered the needed data by means of desk research. However, it is possible that individual research questions may be expanded on if needed or requested by the clients, which may be done by means of comparison of different technological assets.

The method used to acquire data, as previously mentioned, will be qualitative desk research. There are extensive specifications documents pertaining to the STIX and TAXII methodology. Due to the previous iteration of the product having no accompanying documentation, the aforementioned specifications documentation and the OASIS Open GitHub will be the main source of information regarding STIX and its systems.

2.4 Sample

The research samples were selected based on their purpose and their convenience. Multiple articles, and GitHub repositories have been used regarding the research as well as tech sampling.

A substantial excel document displaying a list of cyber incidents that have occurred in the maritime industry is provided to the team for the purpose of scientific research. Multiple articles, videos, and documents also have been used regarding the research as well as sampling. Furthermore, there was a 2016 survey conducted by IHS Fairplay in association with BIMCO which the researchers will rely heavily on to investigate different types of threats that the shipping industry is more prone to.

2.5 Data Collection

The data was collected by the whole team of developers, named "Robert Rachita", "Atanas Hristov", "Victor Tromp", "Line Amini Kaveh", "Chris Sulistiyo" and "Stefan Untura"

The research group was split into 3 sub-groups, each covering a different aspect of the overarching project. As such, the tech requirements survey and network capabilities data was collected by Hristov and Rachita.

The data collected as part of the quantitative research, was gathered through desk research, specifically from the STIX/TAXII specifications documents and the OASIS Open GitHub, which also contains information pertaining to STIX and TAXII. The information will be gathered and processed by the individual research pairs.

The only qualitative data gathered, as part of this research is provided by either the client or as part of specific requests.

2.6 Data Analysis

The research data regarding STIX and TAXII, is qualitative and will be primarily acquired from one source, namely the OASIS Open GitHub and their specification documentation regarding these two systems. The validity of the information regarding STIX/TAXII is high, due to the information coming directly from the company GitHub. Qualitative information regarding current maritime developments and the proficiencies required will come from a mix of specification documents and alternative sources. The validity regarding maritime developments will still be high, as the data will come from various (maritime) organisations and institutions. The validity of information regarding the required proficiencies are high, due to the information being pulled from the TAXII specification document substantiated with additional information from an alternative source.

2.7 Limitations

2.7.1 Research question 3

Due to the nature of the research, there is no population nor sample. Furthermore, the specification documents are very in-depth, and this research is only conducted by one research pair, that needs to cover several different topics differing in complexity. Consequently, there are some time constraints as it impractical and improbable for one research pair extensively study both documents. Moreover, the information is coming from one source, namely OASIS Open, the company that developed the STIX methodology. So, there is a possibility that the research is inherently biased, as there is no alternative source for information regarding the methodology. However, this is unlikely to be an issue, as the use of the STIX methodology was predetermined by the client and the intention was to gauge how it should be applied and not if it was feasible or logical to do so.

Finally, the questions regarding the scalability of the product may be difficult to answer only using specification documents, but also due to the lack of documentation regarding the previous iteration of this product and therefore information pertaining to the scalability of the web application will be given in a general context and/or refer to other research documents.

2.7.2 Research question 4

Because of the nature of this research that relies heavily on an official study of the maritime cyber incident, the limitation of this research is the sample size, which was the surveys done by various official maritime cybersecurity official investigators. This was the last survey conveyed and may or may not has the complete incidents. However, various videos and articles online with regards to the topic dating back to the year 2020 and 2021 also have been assessed, and thus the incomplete/ missing data is then based on them.

3 Results

3.1 Research question #1

What are the most suitable web app development frameworks for the Stix Database Platform that are easy to build, maintain, secure, offer high performance and allow containerization?

While researching this topic, the main requirements formulated by the client have been always kept in mind: secure, fast, easily maintained, and scalable, easily accessible web application. The previous iteration of this product was built using PHP and one of its frameworks, namely Laravel. This has not been considered up to the requirements by the client, and support for one of the libraries used for data visualization was completely dropped, rendering the entire platform not usable. Or in short: the previously used technologies were outdated and unscaleable, which is the polar opposite of what our team is attempting to achieve with this project.

Paying attention to this, the group's attention has been diverted to the other alternatives: Node.js and ASP .NET (Core). As such, here the focus will be on each of these specifications, the differences, and benefits, with a conclusion and recommendation written in the following chapter

Node.js is a back-end JavaScript runtime environment that runs JavaScript code outside of a web browser. It is open-source and cross-platform. Node.js allows developers to utilize JavaScript to create command-line tools and server-side scripting, which involves running scripts on the server before sending the page to the user's browser. As a result, Node.js represents a "JavaScript everywhere" paradigm, unifying online application development together around a single programming language rather than separate languages for server-side and client-side scripts. The event-driven architecture of Node.js allows for asynchronous I/O. These design choices are intended to improve throughput and scalability in web applications with a large number of input/output activities, as well as in real-time Web applications.

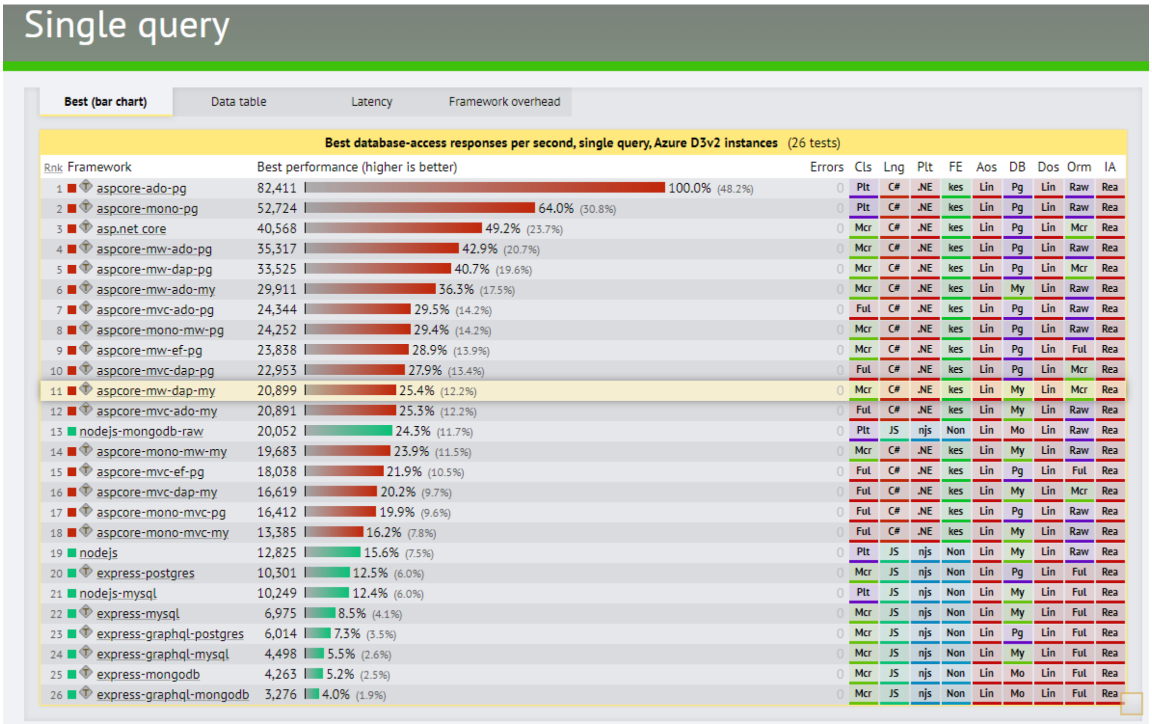
.NET (pronounced "dot net"; formerly known as.NET Core) is a managed computer software framework for Windows, Linux, and macOS that is free and open-source. It succeeds the.NET Framework as a cross-platform framework. Microsoft is primarily responsible for the project's development. The.NET Core framework is a C#-based framework that can only be used in.NET projects. The Common Dialect Runtime underpins NET Core, allowing developers to use languages such as C#, F#, C++, and Visual Basic. MVC (Model View Controller) and Web Shapes are the two main branches of ASP.NET. The most popular.NET libraries are Swagger and Unity. WebAPI, Serilog, IdentityServer04AspNetIdentity, and Nswag all use prewritten code to provide data structures. Different libraries for encryption, security, and database access are also included. To render graphics and communicate with databases, these class libraries are utilized.

Node.js runs on a single thread so it uses fewer resources than traditional process request methods. What happens inside node.js request handling is it takes a request and if it can be processed immediately, sends the response. If it's a time-consuming API call or long-running job pass it to the event loop and continue to the next incoming request. After the event loop has processed the request, the response is sent back using a call back function. While the code is executed in the main thread, it spawns on different threads to perform other tasks. This is a secret to efficient and lightweight solutions even if the app must be heavily loaded with data.

Asynchronous programming is also supported by.NET Core. Each request is handled by a different thread, and no I/O blocks the thread. As a result, ASP.Net Core is currently one of the fastest web frameworks among all web frameworks in all languages, thanks to asynchronous processing. When it comes to performance, there are two terms that stand out: *fast* and *load*. A server can be fast when processing a single request, but it can slow down when handling hundreds or thousands of requests simultaneously (under *load*).

.NET dedicates a single thread for each request from the thread pool. When it gets out of the threads, requests get queued up. But if the requests are non-blocking it won't be a problem again. What's more, .Net has a built in IIS server, which means that requests requesting the static page are not even going to get out of the kernel. On the other side, due to the asynchronous model, Node.js apps have high performance without being too heavy.

These differences can be easily seen from the test of processing single queries vs multiple queries. The first test involved fetching a single row from a single database table, then being serialized as a JSON response. In the following test, 20 queries per request were run.

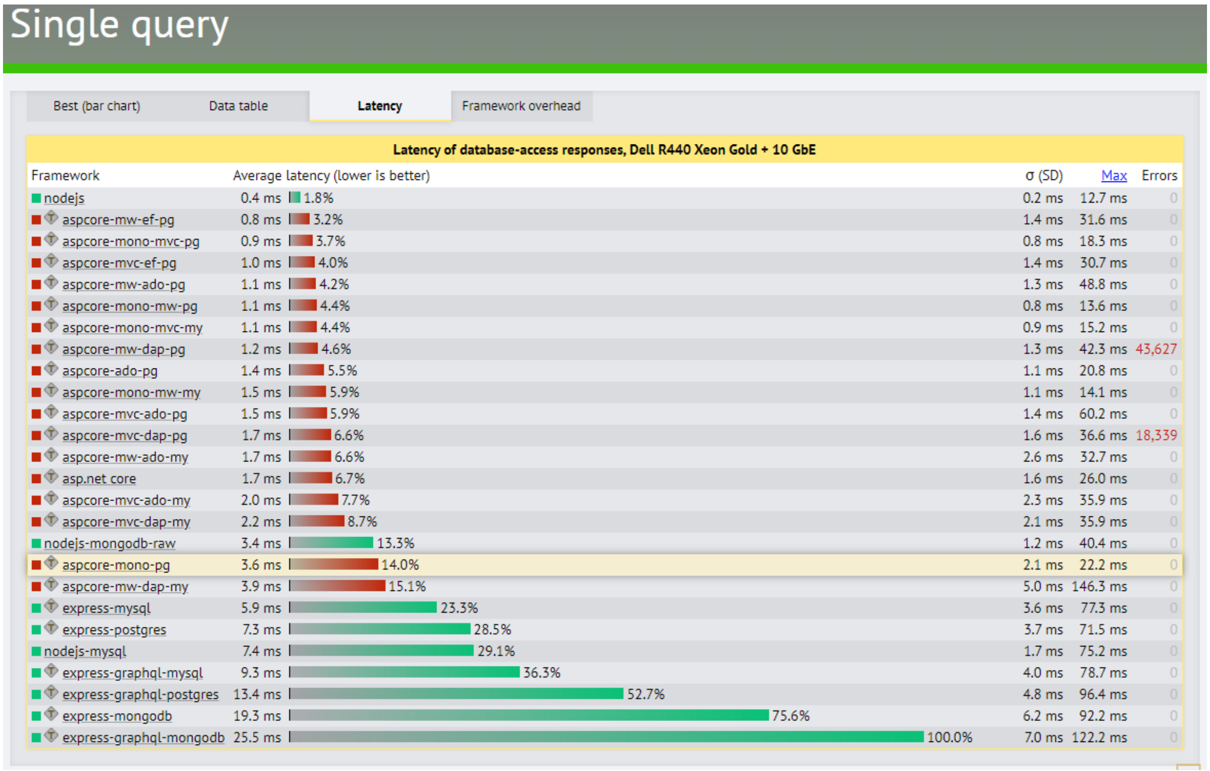


Single query

Best database-access responses per second, single query, Azure D3v2 instances (26 tests)

Rnk	Framework	Best performance (higher is better)	Errors	Cls	Lng	Plt	FE	Aos	DB	Dos	Orm	IA
1	🟥 aspcore-ado-pg	82,411 100.0% (48.2%)	0	Plt	C#	.NE	kes	Lin	Pg	Lin	Raw	Rea
2	🟥 aspcore-mono-pg	52,724 64.0% (30.8%)	0	Plt	C#	.NE	kes	Lin	Pg	Lin	Raw	Rea
3	🟥 asp.net core	40,568 49.2% (23.7%)	0	Mcr	C#	.NE	kes	Lin	Pg	Lin	Mcr	Rea
4	🟥 aspcore-mw-ado-pg	35,317 42.9% (20.7%)	0	Mcr	C#	.NE	kes	Lin	Pg	Lin	Raw	Rea
5	🟥 aspcore-mw-dap-pg	33,525 40.7% (19.6%)	0	Mcr	C#	.NE	kes	Lin	Pg	Lin	Mcr	Rea
6	🟥 aspcore-mw-ado-my	29,911 36.3% (17.5%)	0	Mcr	C#	.NE	kes	Lin	My	Lin	Raw	Rea
7	🟥 aspcore-mvc-ado-pg	24,344 29.5% (14.2%)	0	Ful	C#	.NE	kes	Lin	Pg	Lin	Raw	Rea
8	🟥 aspcore-mono-mw-pg	24,252 29.4% (14.2%)	0	Mcr	C#	.NE	kes	Lin	Pg	Lin	Raw	Rea
9	🟥 aspcore-mw-ef-pg	23,838 28.9% (13.9%)	0	Mcr	C#	.NE	kes	Lin	Pg	Lin	Ful	Rea
10	🟥 aspcore-mvc-dap-pg	22,953 27.9% (13.4%)	0	Ful	C#	.NE	kes	Lin	Pg	Lin	Mcr	Rea
11	🟥 aspcore-mw-dap-my	20,899 25.4% (12.2%)	0	Mcr	C#	.NE	kes	Lin	My	Lin	Mcr	Rea
12	🟥 aspcore-mvc-ado-my	20,891 25.3% (12.2%)	0	Ful	C#	.NE	kes	Lin	My	Lin	Raw	Rea
13	🟩 nodejs-mongodb-raw	20,052 24.3% (11.7%)	0	Plt	JS	njs	Non	Lin	Mo	Lin	Raw	Rea
14	🟥 aspcore-mono-mw-my	19,683 23.9% (11.5%)	0	Mcr	C#	.NE	kes	Lin	My	Lin	Raw	Rea
15	🟥 aspcore-mvc-ef-pg	18,038 21.9% (10.5%)	0	Ful	C#	.NE	kes	Lin	Pg	Lin	Ful	Rea
16	🟥 aspcore-mvc-dap-my	16,619 20.2% (9.7%)	0	Ful	C#	.NE	kes	Lin	My	Lin	Mcr	Rea
17	🟥 aspcore-mono-mvc-pg	16,412 19.9% (9.6%)	0	Ful	C#	.NE	kes	Lin	Pg	Lin	Raw	Rea
18	🟥 aspcore-mono-mvc-my	13,385 16.2% (7.8%)	0	Ful	C#	.NE	kes	Lin	My	Lin	Raw	Rea
19	🟩 nodejs	12,825 15.6% (7.5%)	0	Plt	JS	njs	Non	Lin	My	Lin	Raw	Rea
20	🟩 express-postgres	10,301 12.5% (6.0%)	0	Mcr	JS	njs	Non	Lin	Pg	Lin	Ful	Rea
21	🟩 nodejs-mysql	10,249 12.4% (6.0%)	0	Plt	JS	njs	Non	Lin	My	Lin	Ful	Rea
22	🟩 express-mysql	6,975 8.5% (4.1%)	0	Mcr	JS	njs	Non	Lin	My	Lin	Ful	Rea
23	🟩 express-graphql-postgres	6,014 7.3% (3.5%)	0	Mcr	JS	njs	Non	Lin	Pg	Lin	Ful	Rea
24	🟩 express-graphql-mysql	4,498 5.5% (2.6%)	0	Mcr	JS	njs	Non	Lin	My	Lin	Ful	Rea
25	🟩 express-mongodb	4,263 5.2% (2.5%)	0	Mcr	JS	njs	Non	Lin	Mo	Lin	Ful	Rea
26	🟩 express-graphql-mongodb	3,276 4.0% (1.9%)	0	Mcr	JS	njs	Non	Lin	Mo	Lin	Ful	Rea

Figure 1: Best database-access responses per second for a single query

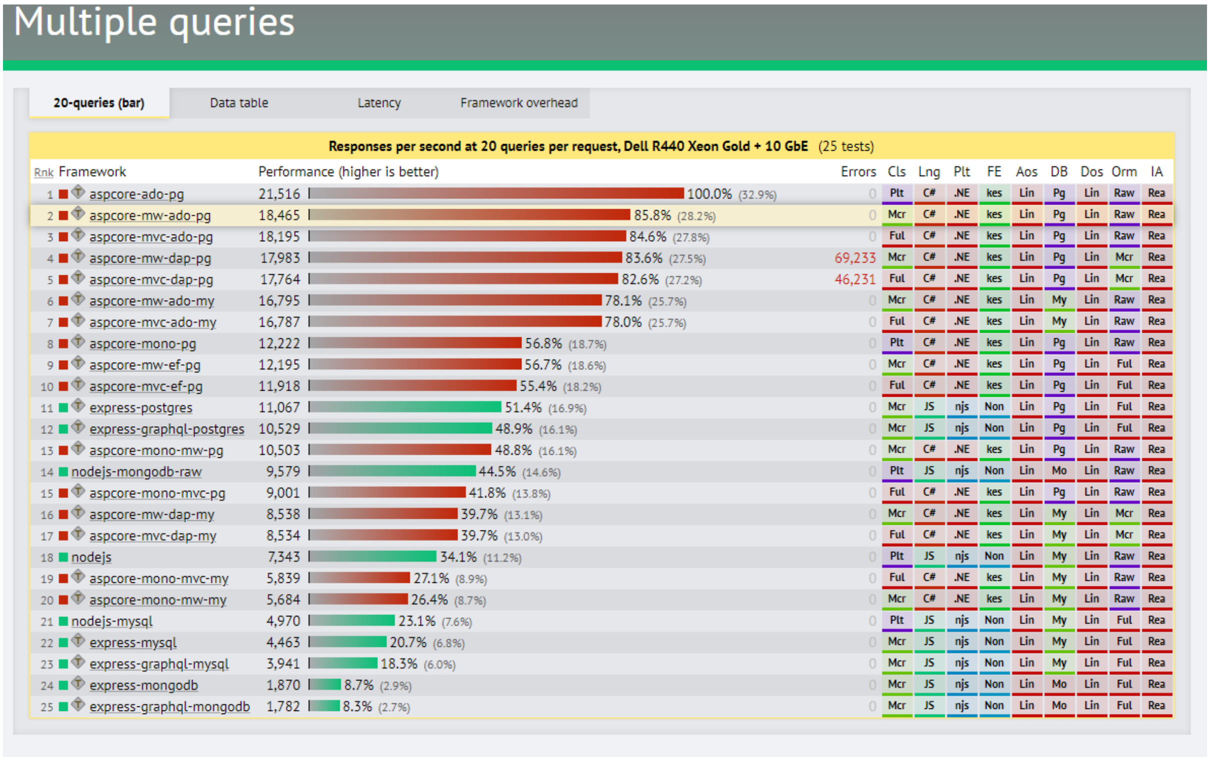


Single query

Latency of database-access responses, Dell R440 Xeon Gold + 10 GbE

Framework	Average latency (lower is better)		σ (SD)	Max	Errors
🟩 nodejs	0.4 ms	1.8%	0.2 ms	12.7 ms	0
🟥 aspcore-mw-ef-pg	0.8 ms	3.2%	1.4 ms	31.6 ms	0
🟥 aspcore-mono-mvc-pg	0.9 ms	3.7%	0.8 ms	18.3 ms	0
🟥 aspcore-mvc-ef-pg	1.0 ms	4.0%	1.4 ms	30.7 ms	0
🟥 aspcore-mw-ado-pg	1.1 ms	4.2%	1.3 ms	48.8 ms	0
🟥 aspcore-mono-mw-pg	1.1 ms	4.4%	0.8 ms	13.6 ms	0
🟥 aspcore-mono-mvc-my	1.1 ms	4.4%	0.9 ms	15.2 ms	0
🟥 aspcore-mw-dap-pg	1.2 ms	4.6%	1.3 ms	42.3 ms	43,627
🟥 aspcore-ado-pg	1.4 ms	5.5%	1.1 ms	20.8 ms	0
🟥 aspcore-mono-mw-my	1.5 ms	5.9%	1.1 ms	14.1 ms	0
🟥 aspcore-mvc-ado-pg	1.5 ms	5.9%	1.4 ms	60.2 ms	0
🟥 aspcore-mvc-dap-pg	1.7 ms	6.6%	1.6 ms	36.6 ms	18,339
🟥 aspcore-mw-ado-my	1.7 ms	6.6%	2.6 ms	32.7 ms	0
🟥 asp.net core	1.7 ms	6.7%	1.6 ms	26.0 ms	0
🟥 aspcore-mvc-ado-my	2.0 ms	7.7%	2.3 ms	35.9 ms	0
🟥 aspcore-mvc-dap-my	2.2 ms	8.7%	2.1 ms	35.9 ms	0
🟩 nodejs-mongodb-raw	3.4 ms	13.3%	1.2 ms	40.4 ms	0
🟥 aspcore-mono-pg	3.6 ms	14.0%	2.1 ms	22.2 ms	0
🟥 aspcore-mw-dap-my	3.9 ms	15.1%	5.0 ms	146.3 ms	0
🟩 express-mysql	5.9 ms	23.3%	3.6 ms	77.3 ms	0
🟩 express-postgres	7.3 ms	28.5%	3.7 ms	71.5 ms	0
🟩 nodejs-mysql	7.4 ms	29.1%	1.7 ms	75.2 ms	0
🟩 express-graphql-mysql	9.3 ms	36.3%	4.0 ms	78.7 ms	0
🟩 express-graphql-postgres	13.4 ms	52.7%	4.8 ms	96.4 ms	0
🟩 express-mongodb	19.3 ms	75.6%	6.2 ms	92.2 ms	0
🟩 express-graphql-mongodb	25.5 ms	100.0%	7.0 ms	122.2 ms	0

Figure 2: Latency of database-access responses

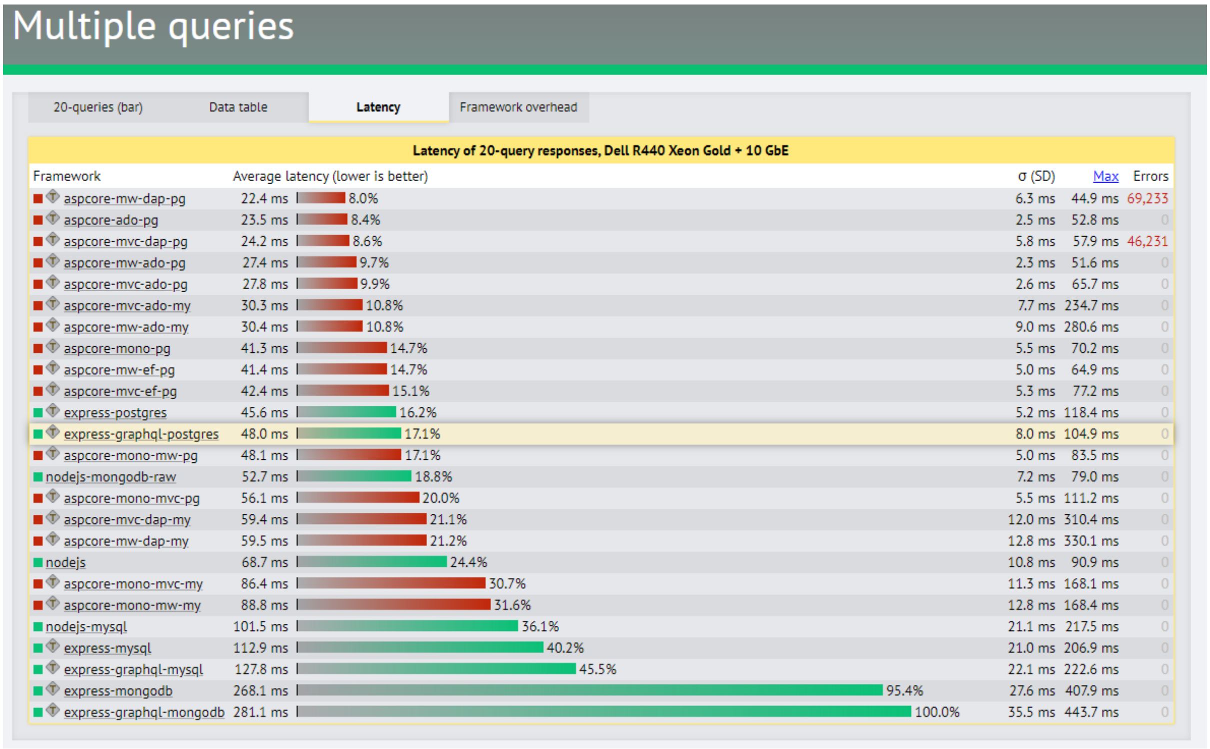


Multiple queries

Responses per second at 20 queries per request, Dell R440 Xeon Gold + 10 GbE (25 tests)

Rnk	Framework	Performance (higher is better)		Errors	Cls	Lng	Plt	FE	Aos	DB	Dos	Orm	IA
1	🟥 aspcore-ado-pg	21,516	100.0% (32.9%)	0	Plt	C#	.NE	kes	Lin	Pg	Lin	Raw	Rea
2	🟥 aspcore-mw-ado-pg	18,465	85.8% (28.2%)	0	Mcr	C#	.NE	kes	Lin	Pg	Lin	Raw	Rea
3	🟥 aspcore-mvc-ado-pg	18,195	84.6% (27.8%)	0	Ful	C#	.NE	kes	Lin	Pg	Lin	Raw	Rea
4	🟥 aspcore-mw-dap-pg	17,983	83.6% (27.5%)	69,233	Mcr	C#	.NE	kes	Lin	Pg	Lin	Mcr	Rea
5	🟥 aspcore-mvc-dap-pg	17,764	82.6% (27.2%)	46,231	Ful	C#	.NE	kes	Lin	Pg	Lin	Mcr	Rea
6	🟥 aspcore-mw-ado-my	16,795	78.1% (25.7%)	0	Mcr	C#	.NE	kes	Lin	My	Lin	Raw	Rea
7	🟥 aspcore-mvc-ado-my	16,787	78.0% (25.7%)	0	Ful	C#	.NE	kes	Lin	My	Lin	Raw	Rea
8	🟥 aspcore-mono-pg	12,222	56.8% (18.7%)	0	Plt	C#	.NE	kes	Lin	Pg	Lin	Raw	Rea
9	🟥 aspcore-mw-ef-pg	12,195	56.7% (18.6%)	0	Mcr	C#	.NE	kes	Lin	Pg	Lin	Ful	Rea
10	🟥 aspcore-mvc-ef-pg	11,918	55.4% (18.2%)	0	Ful	C#	.NE	kes	Lin	Pg	Lin	Ful	Rea
11	🟩 express-postgres	11,067	51.4% (16.9%)	0	Mcr	JS	njs	Non	Lin	Pg	Lin	Ful	Rea
12	🟩 express-graphql-postgres	10,529	48.9% (16.1%)	0	Mcr	JS	njs	Non	Lin	Pg	Lin	Ful	Rea
13	🟥 aspcore-mono-mw-pg	10,503	48.8% (16.1%)	0	Mcr	C#	.NE	kes	Lin	Pg	Lin	Raw	Rea
14	🟩 nodejs-mongodb-raw	9,579	44.5% (14.6%)	0	Plt	JS	njs	Non	Lin	Mo	Lin	Raw	Rea
15	🟥 aspcore-mono-mvc-pg	9,001	41.8% (13.8%)	0	Ful	C#	.NE	kes	Lin	Pg	Lin	Raw	Rea
16	🟥 aspcore-mw-dap-my	8,538	39.7% (13.1%)	0	Mcr	C#	.NE	kes	Lin	My	Lin	Mcr	Rea
17	🟥 aspcore-mvc-dap-my	8,534	39.7% (13.0%)	0	Ful	C#	.NE	kes	Lin	My	Lin	Mcr	Rea
18	🟩 nodejs	7,343	34.1% (11.2%)	0	Plt	JS	njs	Non	Lin	My	Lin	Raw	Rea
19	🟥 aspcore-mono-mvc-my	5,839	27.1% (8.9%)	0	Ful	C#	.NE	kes	Lin	My	Lin	Raw	Rea
20	🟥 aspcore-mono-mw-my	5,684	26.4% (8.7%)	0	Mcr	C#	.NE	kes	Lin	My	Lin	Raw	Rea
21	🟩 nodejs-mysql	4,970	23.1% (7.6%)	0	Plt	JS	njs	Non	Lin	My	Lin	Ful	Rea
22	🟩 express-mysql	4,463	20.7% (6.8%)	0	Mcr	JS	njs	Non	Lin	My	Lin	Ful	Rea
23	🟩 express-graphql-mysql	3,941	18.3% (6.0%)	0	Mcr	JS	njs	Non	Lin	My	Lin	Ful	Rea
24	🟩 express-mongodb	1,870	8.7% (2.9%)	0	Mcr	JS	njs	Non	Lin	Mo	Lin	Ful	Rea
25	🟩 express-graphql-mongodb	1,782	8.3% (2.7%)	0	Mcr	JS	njs	Non	Lin	Mo	Lin	Ful	Rea

Figure 3: Reponses per second per 20 queries per request



Multiple queries

Latency of 20-query responses, Dell R440 Xeon Gold + 10 GbE

Framework	Average latency (lower is better)		σ (SD)	Max	Errors
🟥 aspcore-mw-dap-pg	22.4 ms	8.0%	6.3 ms	44.9 ms	69,233
🟥 aspcore-ado-pg	23.5 ms	8.4%	2.5 ms	52.8 ms	0
🟥 aspcore-mvc-dap-pg	24.2 ms	8.6%	5.8 ms	57.9 ms	46,231
🟥 aspcore-mw-ado-pg	27.4 ms	9.7%	2.3 ms	51.6 ms	0
🟥 aspcore-mvc-ado-pg	27.8 ms	9.9%	2.6 ms	65.7 ms	0
🟥 aspcore-mvc-ado-my	30.3 ms	10.8%	7.7 ms	234.7 ms	0
🟥 aspcore-mw-ado-my	30.4 ms	10.8%	9.0 ms	280.6 ms	0
🟥 aspcore-mono-pg	41.3 ms	14.7%	5.5 ms	70.2 ms	0
🟥 aspcore-mw-ef-pg	41.4 ms	14.7%	5.0 ms	64.9 ms	0
🟥 aspcore-mvc-ef-pg	42.4 ms	15.1%	5.3 ms	77.2 ms	0
🟩 express-postgres	45.6 ms	16.2%	5.2 ms	118.4 ms	0
🟩 express-graphql-postgres	48.0 ms	17.1%	8.0 ms	104.9 ms	0
🟥 aspcore-mono-mw-pg	48.1 ms	17.1%	5.0 ms	83.5 ms	0
🟩 nodejs-mongodb-raw	52.7 ms	18.8%	7.2 ms	79.0 ms	0
🟥 aspcore-mono-mvc-pg	56.1 ms	20.0%	5.5 ms	111.2 ms	0
🟥 aspcore-mvc-dap-my	59.4 ms	21.1%	12.0 ms	310.4 ms	0
🟥 aspcore-mw-dap-my	59.5 ms	21.2%	12.8 ms	330.1 ms	0
🟩 nodejs	68.7 ms	24.4%	10.8 ms	90.9 ms	0
🟥 aspcore-mono-mvc-my	86.4 ms	30.7%	11.3 ms	168.1 ms	0
🟥 aspcore-mono-mw-my	88.8 ms	31.6%	12.8 ms	168.4 ms	0
🟩 nodejs-mysql	101.5 ms	36.1%	21.1 ms	217.5 ms	0
🟩 express-mysql	112.9 ms	40.2%	21.0 ms	206.9 ms	0
🟩 express-graphql-mysql	127.8 ms	45.5%	22.1 ms	222.6 ms	0
🟩 express-mongodb	268.1 ms	95.4%	27.6 ms	407.9 ms	0
🟩 express-graphql-mongodb	281.1 ms	100.0%	35.5 ms	443.7 ms	0

Figure 4: Latency of 20-query responses

While Node.JS showed lower latency in the single query latency test, it was in milliseconds which is negligible. In all other tests, both for database access and the latency, ASP .NET was significantly faster, sometimes up to 10 times.

For platform support, while.NET was once solely accessible for Windows, both frameworks are now widely utilized by developers on Windows, Linux, and macOS. Both development environments can claim of active and substantial community, which ultimately means it won't be burdensome to find a solution to the problem. However, they have different integrations and places where these communities can be found. .NET has more community support on Stack Overflow whereas Node.js is supported more via GitHub. The best example is the Stack Overflow question and answer website that has around 4 million registered users.

Node.js is perfectly designed for distributed systems. Micro services-based software enables components to scale autonomously which prevents an app from falling apart due to its weight. Because a single instance utilizes just one thread, you'll need to use the Cluster module or external process control to load all of the server's cores. A load balancer like AWS ELB/NGINX is required to scale to several servers. There are additional technologies, such as Istio, that allow you to manage services and traffic. As more and more enterprises prefer to develop the software on top of micro services, Node.js has gained tremendous popularity among businesses like eBay, Netflix, Twitter, Uber, and others.

.NET Core is also a very scalable platform. It fits into micro services architecture equally well. Since the platform allows you to develop multi-threaded apps, loading a single server is an easier than with Node.js. Microsoft has published the Azure Service Fabric SDK for activities such as installation, scaling, and service resilience. This SDK may be used on your own servers, Azure, or AWS.

In the security and stability aspect, ASP.NET Core/.NET Core is definitely a winner. The security and reliability the platform provides make it a great option to create robust software with C# language. Node.js is more reliable for complex enterprise software developed with Typescript than on its own.

3.1.1 Use cases for both technologies

Node.js is better at handling multiple tasks simultaneously due to its single-threaded event loop. Node.js was designed as an alternative to Apache HTTP Server. Node.js is based on JavaScript and developers can use a single language to write both front end and backend. Developers can use pre-written code from the node package manager(npm) and use the modules for quick and efficient programming.

Node.js can handle a high amount of traffic with stable requirements without needing to increase the server capacity. Node.js tends to handle all its requests in an asynchronous manner with its high speed V8 engine and single-threaded asynchronous model.

.NET core is based on object-oriented programming language C#. It also can be used to build projects with F# language. The .NET core ecosystem provides prebuilt class libraries including both simple and complex data structures. Net core also provides specific libraries for database access, security, encryption, and database access. These libraries can be used in graphic rendering projects database interactions and XML manipulations. .NET core unlocked the possibility of cross-platform application development as well. Net corecan be used within a vast variety of development requirements such as Desktop, Web, Cloud, Mobile, Gaming, IoT, and AI.

Due to its modular and scalable nature, .NET can be used to build micro services. so .NET can be used to build highly scalable systems with micro services to give an efficient user experience.

Node.JS	ASP .NET (Core)
• Can use a single language JavaScript to perform both front end backend development • Node is cross-platform and asynchronous by origin. • Node.js interprets the JavaScript code via Google's V8 JavaScript engine. It compiles JavaScript code into the machine code(directly) • Support of NPM package manager to use external modules • Code modularity causes less dependency inside the project and exported modules can be used in multiple projects without any side effects on existing code.	• Possibility of separation of concerns • Reduced coding time with inbuilt libraries • High Performance • Security • Supportive coding environment and tooling with Visual Studio • Cross-Platform support • Development and deployment across multiple platforms Cloud, IoT, Desktop, etc. • Maintaining a large .Net Core application is much more easier comparing to Node.js.

Table 1: Node.JS vs ASP .NET (Core)

3.2 Research question #2

What database technology or combination of technologies would be the fastest and most suitable for the STIX platform, executing each query within less than 10 milliseconds?

During our study, we came to an agreement with the customer that the technology (or combination of technologies) used to develop and manage the database must meet certain criteria: the database must be scalable, quick, and secure. This prompted a thorough examination of several database technologies.

To begin, we focused on the ones that were most often utilized in the past and why. The intention was that this technique would provide an overview of the technologies in use, after which all of them would be extensively investigated and the best choice selected. Due to apparent considerations, only a few technologies will be described.

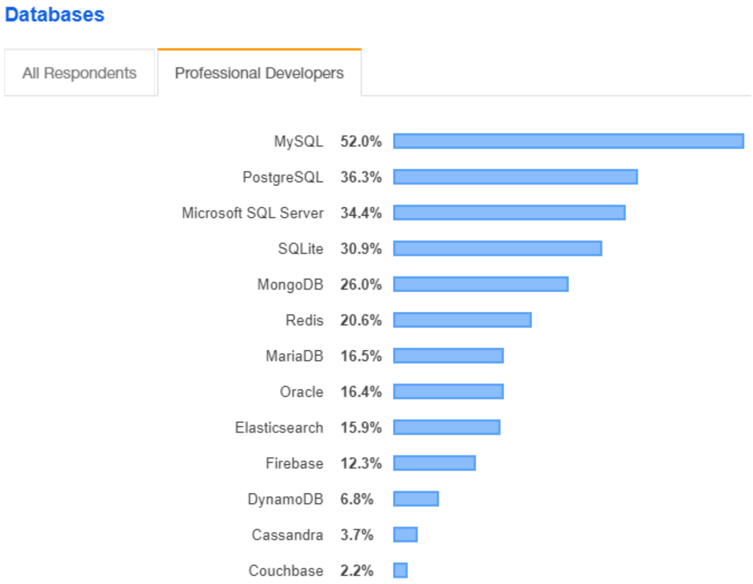


Figure 5: Database system usage amongst Professional Developers

As you can see SQL is still going strong, with more than 52% of professional developers using it in their daily workflow.

Furthermore, you should investigate whether one of those platforms offers considerable benefits over SQL in your process. Although SQL is a tried-and-true technology, it has its limitations when it comes to large-scale enterprises. Even among the most advanced software organizations, MySQL remains popular despite the proliferation of various competing database systems designed for current uses. Netflix, Twitter, Slack, and other well-known organizations are just a handful of the millions of businesses who use MySQL in some form.

So, what are the prominent features of MySQL?

- **Client & Server Communication.** MySQL makes it simple to communicate back and forth between the client and the server. This makes scaling MySQL applications across multiple servers, networks, and cloud platforms a breeze.
- **Flexibility.** MySQL excels at allowing developers to manage data in a variety of ways. This is mostly accomplished through its SELECT syntax, which offers immense capacity for organizing even the most massive data sets.

- **Performance.** Although "MySQL and Performance" did not always go hand in hand, the system's most recent versions have shown to be incredibly performance friendly.
- **Support.** You get access to a significantly wider pool of support requirements since MySQL has been around longer than most other database systems. This includes the material itself, as well as Q&A websites and, of course, online lessons.

All of these factors combine to make MySQL an excellent choice for any project. However, there are some disadvantages as well. One of them is that MySQL uses tables, rows, and columns to store data. To put it another way, as your database increases, so does the need for resources to access, process, and use it.

MariaDB has a fascinating backstory behind its creation. MariaDB is a derivative of MySQL that is backed by some of the original MySQL developers. Because of worries about Oracle (business) purchasing MySQL, MariaDB was created in 2009. Since then, the project has evolved and grown to the point that, as indicated in the graph above, it is now utilized by around 17% of professional developers.

MongoDB is frequently chosen by developers who are already aware that they will be dealing with large volumes of mixed data since it offers the most flexibility. That is, for the purpose of handling the data. Data and its types can be readily preserved, either for immediate or later usage, without being scheme-bound.

- **Scalability.** Do you want to move your data between servers? MongoDB promotes this behaviour since it adds to the overall speed of the database. Furthermore, all data is recorded in the JSON format.
- **Mobile Development.** MERN (MongoDB, Express, React, and Node) is a new type of tech stack designed for the building of high-end mobile apps. MongoDB is the obvious choice for this task, given that mobile applications generate massive amounts of data at a rapid rate.
- **Dynamic Structure.** Problems might occur without warning. In the case of MongoDB, you gain a lot of freedom when it comes to dynamically moving or transferring your database without having to deal with any data structure difficulties.

There is a reason MongoDB is used by more than 26% of professional developers on a regular basis. It is a robust, document-based database system designed for current software and applications.

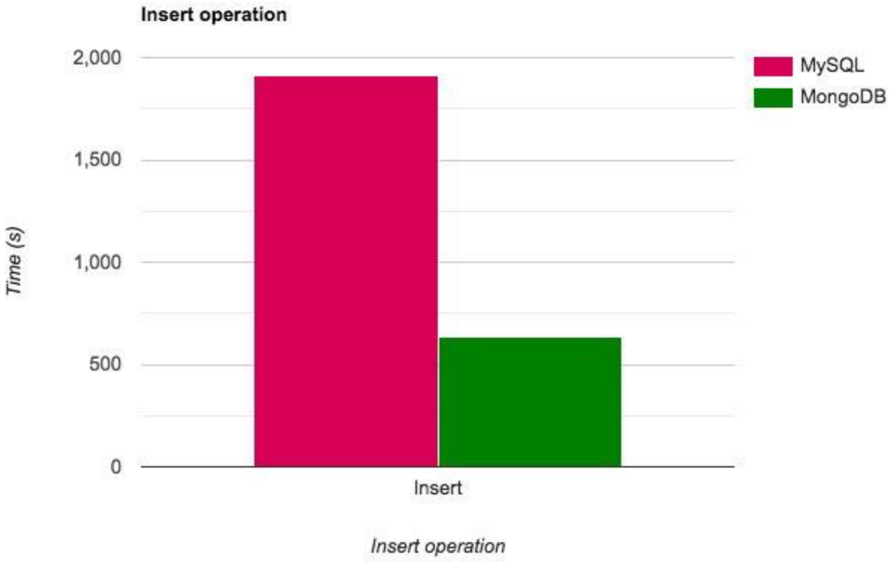


Figure 6: MySQL vs. MongoDB insertion response time

3.3 Research Question #3

How can the STIX language be utilized to serve (maritime) cyber security needs?

The section “How can the STIX language be utilized to serve (maritime) cyber security needs?” is comprised of multiple sub-questions that ultimately build up to answer this question within the conclusion chapter.

3.3.1 What is STIX?

This part of the research is to answer what exactly STIX is and what it encompasses. Note that the information here is gathered directly from the STIX specification document as there are no alternative sources for STIX outside their own GitHub. Furthermore, all features and processes described in the following paragraphs refer to STIX version 2.1 unless stated otherwise.

Structured Threat Information eXpression is a language and serialization format used to exchange cyber threat intelligence (CTI). STIX enables companies and organizations to share CTI amongst each other in a standardized and machine-readable manner, specifically UTF-8 encoded JSON. This enables security communities to gain a better understanding of the kind of cyber-attacks they are most likely to face, as well as to anticipate and respond to these attacks in a more effective, efficient, and holistic manner. STIX is designed to improve many different capabilities, such as collaborative threat analysis, automated threat exchange, automated detection and response, and more.

3.3.1.1 Overview

STIX is a structed framework that defines a classification of CTI’s. This is broken down into STIX Core Objects and STIX Meta Objects.

STIX Core objects can be further broken down into STIX Domain Objects, STIX Cyber-observable Objects, STIX Relationship Objects.

As for STIX Meta Objects, these are broken down into Extension Definition Objects, Language Content Objects and Marking Definition Objects.

Finally, there is the STIX Bundle Object, that provides a wrapper mechanism for grouping STIX content together. STIX is a connected graph of nodes and edges. STIX Domain Objects and STIX Cyber-observable Objects define the graph nodes and STIX relationships define the edges.

The table underneath provides an overview of the various Objects mentioned above:

STIX Objects						STIX Bundle Object
STIX Core Objects			STIX Meta Objects (SMO)			
STIX Domain Objects (SDO)	STIX Cyber-observable Objects (SCO)	STIX Relationship Objects (SRO)	Extension Definition Objects	Language Content Objects	Marking Definition Objects	

Table 1: STIX Objects (OASIS Open, 2021)

3.3.1.2 STIX Objects

This section will briefly dive into the various STIX objects.

3.3.1.3 STIX Domain Objects

STIX defines a set of STIX Domain Objects and each of the following objects corresponds to a concept commonly used in CTI: Attack Pattern, Campaign, Course of Action, Grouping, Identity, Indicator, Infrastructure, Intrusion Set, Location, Malware, Malware Analysis, Note, Observed Data, Opinion, Report, Threat Actor, Tool, and Vulnerability. Individuals can generate and share broad and comprehensive cyber threat intelligence using SDOs, STIX Cyber-observable Objects (SCOs), and STIX Relationship Objects (SROs) as building blocks.

STIX offers in-depth descriptions of these objects. please refer to Appendix 7.1 for more information regarding descriptions, properties, and examples for each object.

3.3.1.4 STIX Cyber-observable Objects

STIX defines a set of SCOs for characterizing host-based and network-based information. SCOs are used by various SDOs for providing context. STIX SCOs document the facts concerning what happened on a network or host but not the who, when, and why. By associating SCOs with SDOs, it becomes possible to communicate a higher-level understanding of the threat landscape, as well as provide insight into the who and the why specific intelligence may be significant to organizations.

For in-depth definitions regarding SCOs, please refer to Appendix 7.2.

3.3.1.5 STIX Relationships

A relationship is a link that specifies the way in which two SDOs, SCOs, or an SDO and a SCO are associated. Relationships can be represented by an external SRO or, in some cases, by specific attributes that store an identifying reference for an embedded relationship (for example, the created_by_ref property).

One of two SROs, the generic SRO, is utilized for most STIX partnerships. This generic SRO has a property named relationship_type that describes the relationship in greater detail. This specification establishes a vocabulary for the relationship_type attribute between SDOs of different sorts. The Indicator SDO, for example, establishes a link between itself and Malware using a relationship_type of *indicates* to specify how the indicator can be used to identify the presence of the relevant Malware. STIX permits user-specified terms to be used as the relationship type in addition to the terms defined in the standard.

The Sighting SRO is currently the only other SRO (apart from a generic Relationship). The Sighting object is used to record instances in which an entity has "seen" an SDO, such as when an indicator has been sighted. Because it has additional attributes such as count that are only applicable to Sighting connections, Sighting is a different SRO. If new relationships are discovered that require extra features not provided on the general Relationship object, other SROs may be developed in future versions of STIX.

STIX uses ID references to express embedded relationships in addition to relationships built using the SROs (Relationship and Sighting). Embedded relationships are just STIX Objects' ID reference attributes that hold the ID of another STIX Object. When the property is an inherent component of the item and not something that a third party might contribute or that requires the addition of a confidence score, embedded relationships are employed. An SRO is not required to represent them because they represent an intrinsic relationship and have no additional properties. Only the creator of the object in which it is contained ("object creator") can assert an embedded connection. Because the entity that created a STIX Object is an intrinsic, factual element of that object, rather than using an SRO, that information is stored in an embedded relationship contained in the created by ref attribute.

For in-depth information regarding STIX relationships, please refer to Appendix 7.3.

3.3.1.6 STIX Patterning

The STIX Patterning language allows activity on networks and endpoints to be detected. This language enables comparisons with time-stamped cyber observable data acquired by a threat intelligence platform or other comparable system. The STIX Indicator object is the sole object that uses STIX Patterning at the moment, but it can be utilized in other situations.

STIX Patterning was created with STIX Indicators in mind. As such, it's a way of communicating where malicious code and/or threat actors are present on a network or endpoint.

The language release focuses on supporting a common set of use cases, allowing the expression of a first set of patterns that STIX producers and consumers can employ. The STIX patterning language will be extended in future editions when more complicated patterns are considered essential to improve its effectiveness as an automated detection/remediation approach.

For more in-depth information, please refer to Appendix 7.4.

3.3.1.7 Transporting STIX

The structures and serializations in STIX 2.1 are transport-agnostic, which means they don't rely on any single transport mechanism. TAXII a companion CTI specification, is specifically developed to transfer STIX Objects. And will be further discussed in the following TAXII section.

3.3.1.8 TAXII

Like STIX, TAXII (Trusted Automated Exchange of Intelligence Information) is an OASIS standard and thus developed, managed, and updated by the CTI Technical Committee (TC), with all relevant, publicly available documentation accessible on the official STIX website and the CTI TC repository. All features and processes described in the following paragraphs refer to TAXII version 2.1 unless stated otherwise.

TAXII is a protocol that defines how cyber threat intelligence can be shared among organizations via HTTPS, in a manner that promotes scalability and simple efficiency. Now, this is handled via API's that can be set up by independent organizations using the TAXII requirements to provide information in line with commonly used CTI sharing models. This allows for easy integration with sharing models that are already in place. While TAXII is first and foremost meant to support the exchange of STIX 2 intelligence, the use for external, unrelated content is neither prohibited nor hindered, as it is payload-agnostic.

There are several different ways TAXII can be used for CTI exchanges, with publishers pushing data via the TAXII server to a bunch of different subscribers or clients requesting CTI from the server utilizing a set of metadata filters embedded in the request. Other than that, there is the possibility for the client to request an overview of the available CTI content and relevant information on the collections structure.

These actions are made possible through the two primary services defined by TAXII, that work well with most known CTI sharing models. The first of which are collections, that act as an interface to an assortment of CTI objects located on a TAXII server. A producer can host data on said server that can be accessed via specific requests made by the consumer as explained in the paragraph above. The illustration below depicts a successful interaction, wherein a single client sends a request which is fulfilled by the server with the information available.

Channels on the other hand operate on the source-subscriber model, wherein one or multiple producers are able to push data to subscribers (clients) on specific channels that are hosted on TAXII servers. Below, there is only one producer depicted, however there is the possibility to host multiple channels per API Root, which contains links to available sources in the API. This goes for both models. It should be noted, that the TAXII 2.1 standard reserves the channel keywords but does not define Channel services as of now, as these and other related services are to be established in subsequent versions of the application layer protocol.

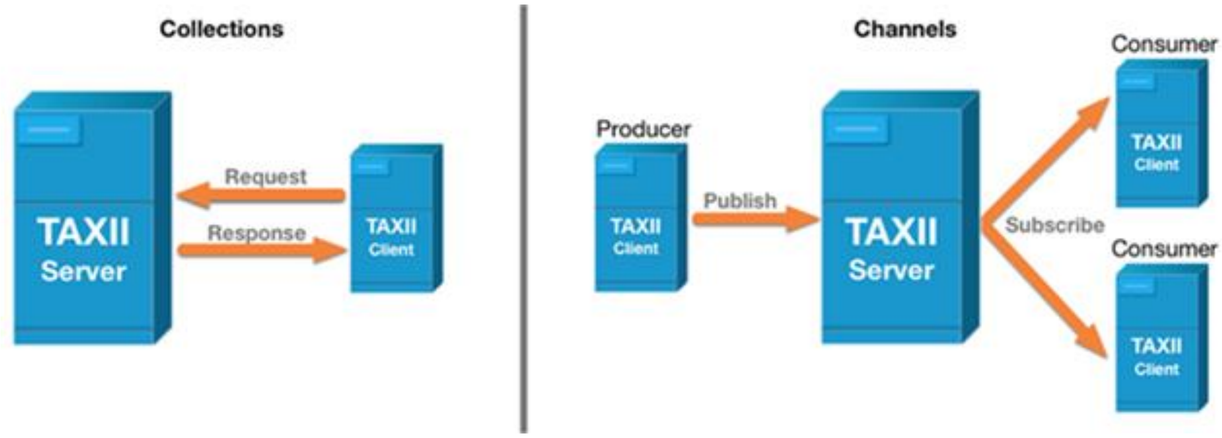


Figure 7: TAXII Collections and Channels (OASIS Open, 2021)

Lastly, there is the possibility for peer-to-peer data transmission, which is not depicted above and not further defined in the official TAXII standards documentation.

As stated above, one server can host multiple channels and collections using API roots, however, it is also possible for one TAXII server to support several API Roots. These logical groupings of channels and collections can be seen as separate instances of the TAXII API.

3.3.1.9 Discovering and Accessing TAXII servers

When given the opportunity TAXII prefers to rely on protocols that are already in place, which is why TAXII servers can be found in networks via DNS Service records and by a Discovery Endpoint. With the update to TAXII 2.1, the old DNS SRV record (“taxii”) and discovery URL (“/taxii/”) have become outdated, and no longer refer to servers of the latest TAXII iteration.

One way of discovering TAXII servers is via a DNS SRV record, which occurs on network level and provides a list of IP-based services offered under that domain with their individual metadata. The current DNS SRV record is “taxii2” and can be used to locate TAXII 2.1 servers within a closed organizations network or the general internet.

The other discovery method is a discovery endpoint that allows authorized clients to request a list of server API roots and general information. This enables them to make specific URL based requests to server-based API root endpoints that contain logically grouped and formatted CTI objects. Said logical groups are vital for server maintenance as they simplify access control, by allowing a single server to host API roots for multiple groups of clients with different access permissions. The endpoints contained in these individual API roots are ultimately what the client interacts with to access server content.

An example for an API Root URL is “https://example.org/taxii2/api1/” (OASIS Open, 2021). This URL consists of four main components, with the last one specifying the API Root endpoint on a TAXII 2.1 server.

3.3.1.10 Authentication and Authorization

The abovementioned access control for API endpoints within the network or organization is not defined within official TAXII documentation. However, in regard to authentication, it is recommended to use the HTTP Basic authentication for TAXII servers. Thus, clients and servers should be set up to support the authentication scheme, however this can be overridden if alternative schemes are implemented. Ultimately, there is no requirement for this particular mode of authentication and other resources can be used as needed.

Regardless of HTTP method, clients will be rejected from accessing the server should the request be sent sans an acceptable authorization header and the access is in no other way authorized.

3.3.4 What are the (security) needs of the stakeholder?

The main stakeholder is Stephen McCombie, PhD in Computer Science and has over 25 years of international experience in the field of IT security. His career developed from his initiating role within the New South Wales Police Force (Australia) as a specialist in investigating complex computer crime through the aviation and banking sectors, among others. In the area of maritime IT security, McCombie's work has included collaborating on a maritime cyber scenario simulation of a maritime conflict in the South China Sea. Most recently, he served as a Senior Lecturer in Cyber Security at Macquarie University, Sydney, Australia. Stephen holds a PhD on the impact of Eastern European cybercrime groups.

Moreover, he is a Professor of Maritime IT Security (MITS) at NHL Stenden University of Applied Sciences, a state-funded professional university in the north of the Netherlands. The MITS research group, that has an emphasis on digital safety and ship IT security, is aligned with the Dutch Maritime Strategy and Seaports, in which ship security and port security are fundamental pillars. More than 90 percent of international trade is carried out by sea transport. Shipping transports a vast majority of all fuels, raw materials, and goods that are used to power the global economy. Port systems and ships are becoming more integrated as a result of digital infrastructure and the internet. This expands the possibilities for autonomous (unmanned) shipping while simultaneously increasing the risk of ransomware and other cyber threats.

Naval digital security is a critical area of research in support of free trade and open ocean shipping. The recent cyber-attack on the Colonial Pipeline in the U.S. shows the significant risks and potential impacts to critical infrastructure. From Texas to New York, the Colonial Pipeline transports gasoline, diesel, and jet fuel. The pipeline system transports about 45 percent of all fuel consumed on the East Coast. On May 7, 2021, the Colonial Pipeline was hit by a ransomware cyber-attack that affected the pipeline's digital equipment, causing all pipeline operations to be halted to contain the attack. As a result, the group that initiated the attack demanded a 75-bitcoin ransom (~$4.4 million at the time) to provide a tool that would restore the system. After several high-profile attacks, there has been growing concern about the vulnerability of (critical) infrastructure to cyberattacks. The ransomware attack on Colonial Pipeline illustrates the need for separate, offline backup systems and cyber incident response plans.

There is a severe lack of (public) tools to help prevent / counter cyber-threats. Stephen McCombie would like to rectify this by setting out the task to develop a web application where various organizations can share cyber threat information amongst themselves to help anticipate and respond to possible attacks in a more effective and efficient manner. There was already a proof-of-concept web application built, but it had a multitude of issues that boiled down to the concept not being up to par and the systems in place were outdated and unscalable.

The end goal of this web application is to be secure, fast, easily maintained, scalable, intuitive web application that can be utilized globally by organizations. In the hope that the more informed these organizations are, the more difficult it becomes to launch cyber-attacks in the Future. This web application will have multiple iterations, each iteration building upon / improving the previous iteration, until the web application reached its final state. The web application will be an online catalogue, where organizations can upload cyber threat information to a database, where this information will be publicly available for other organizations to inform themselves. This web application will be developed utilizing the STIX / TAXII systems.

For more information regarding STIX and TAXII, refer to research question #x and its appendices. For more information regarding which technology & infrastructure will be utilized for the web application, refer to research question #1.

3.3.5 What proficiencies are needed to use the given tools in an alternative context?

In this section, the proficiencies needed for both the development team as well as the end users are specified. The former will address the process of setting up an API, storing data appropriately and visualizing data at a rudimentary level. The information regarding API and database setup, is primarily if not solely gathered from official STIX documentation, whereas the data modelling needs are gauged based on different sources, such as independent available data visualisation tools. Proficiencies pertaining to the end users and their usage of the finished product, are detailed under the assumption that the target group has a basic understanding of cyber security and the contents to be shared on said web application.

As of now, there is no definite answer to this question, as many aspects such as the database structure are not yet decided on and the answers seen in the following paragraphs consist of technologies and concepts that are frequently mentioned in either official STIX documentation or related publications regarding the implementation and alteration process.

3.3.5.1 Required proficiencies for development

Since the STIX methodology, and thus the TAXII protocol, are going to be utilized, all parties involved in the development process need to be familiar with their contents on a basic level. To fully grasp the inner workings of the TAXII protocol, a basic understanding of networking and network structures, to be able to set up a suitable API.

As discussed previously in question 3.1 ("What is STIX?"), TAXII servers can be set up to host collections and channels using API Roots for accessibility. These concepts are detailed in Appendix 7.5, although a brief explanation is provided in the exploration of the aforementioned question.

For the development of the website frontend, the team needs to be proficient in either JavaScript or typescript, depending on the final decision, to retrieve the data to be visualized. As scalability is of the essence, one needs to keep in mind which technologies are most suitable for supporting large amounts of data. The creator of a publicly available drag-and-drop modeler for STIX 2.1 recommends, utilizing forced directed graph algorithms for the visualization of greater quantities of nodes. Furthermore, he directly states that React, MobX (state management) and Webpack are used as the three primary technologies used and that "Some level of competence with these techs will be needed to make code changes", (Minnick, 2020). This, however, is not necessarily what is going to be used within this project.

Now as mentioned, forced directed graphs are beneficial to use for data heavy visualisations, as they are frequently used for network visualization, due to their typically symmetrical and clustered structure.

Regarding the database structure, in question 3.2, the benefits and detriments of different database technologies have been explored, and while technical documentation for said aspect, is not yet finalised, the mentioned technologies are likely to be part of the project realisation. The same goes for the web app development frameworks that are detailed in section 3.1.

Furthermore, to ensure security, fitting methods of authentication and authorization need to be decided on and implemented. It is recommended to make use of Basic HTTP authentication for TAXII servers, however other methods may be implemented as needed. The HTTP basic authentication framework is used to determine the validity of client requests made to a server and allows the client to authenticate data. For a client to successfully authenticate themselves, they need to include an authorization request header with credentials, as unauthorised attempts to access the server are responded to with a 401-response status.

3.3.5.2 Required proficiencies for end users

For users to understand the information provided through the API, they need to be aware of the STIX formatting norms, that are meant to be applied. Although, the information will likely be displayed on the web application, thus the user will not need to know what the information looks like formatted in JSON files, for example. However, the categorization of threats will retain STIX structure, though expanded.

Finally, the UI design will follow conventional design patterns to ensure accessibility, meaning there is no need for particular knowledge beyond basic tech literacy. The user experience is going to be predominantly focused on the products output and information presented in an accessible manner, with no regard for the inner workings such as server connections and data encoding.

3.3.6 What are the current (technological) developments in the (maritime) security sector?

Trade and the sea have long been intrinsically tied, and the world's reliance on healthy, productive oceans is profound. Global governments are acknowledging the maritime sector's importance and future. In today's marine industry, achieving security compliance is an arduous task because there is so much to consider and so many possibilities for a security breach to occur.

Security personnel must be cautious not only to avoid attempts to undermine the nation's laws and security, but they must also be vigilant against local and internal threats. Small-scale attacks can nonetheless be devastating to a marine enterprise, resulting in the loss of lives, severe environmental damage, or property damage. There are a multitude of issues plaguing the current maritime sector, such as thievery, terrorist threats, piracy, etc. However, the marine business is clearly changing. And, as more processes are automated and technology improves, security will eventually shift with it.

The push toward fully automated cargo vessels is one such example of technical advancement in the maritime sector. If developed, a vessel without crew members would be much more cost-effective for shipping corporations, as well as potentially more resistant to security threats – if a ship did not require crew members to operate it, limiting access to the vessel may drastically minimize the risk of piracy. To operate at peak efficiency, these ships can analyse real-time data based on information from other vessels, ports, weather conditions, and other factors. However, these ships are still in the testing phase, and firms will most likely be slow to embrace completely automated ships due to liability, cybersecurity, and safety concerns.

The International Maritime Organization (IMO) recognizes that a ship's onboard information technology and operational technology systems can be hacked just as easily as systems ashore, and that such security breaches have the potential to cause significant harm to the safety and security of ships, ports, marine facilities, and other elements of the maritime transportation system, the IMO has taken the initiative to raise awareness across the industry on how to deal with risks by promoting guidelines on maritime cyber risk management. The Guidelines provide high-level recommendations and best-practices on managing marine cyber risk to protect ships from current and emerging cyberthreats and vulnerabilities. But these are only guidelines, and guidelines alone cannot hope to protect the entire maritime sector. Therefore, technologies are being developed to tackle this issue and to bolster the maritime security infrastructure. Two of such technologies in the spotlight are Automatic Identification Systems (AIS) spoofing and Global Navigation Satellite System (GNSS) manipulation.

AIS Spoofing involves the creation of a non-existent vessel or the concealment of a vessel's actual identity, leading in the concealment or transmission of false positional data. AIS were initially developed as a safety precaution to avoid collisions. The usage of AIS has evolved throughout time, as has the reliance of vessels on it. AIS is being used by governments and security organizations to detect and deter illegal activity at sea.

Signal manipulation was originally thought to be a specialized and improbable concern for commercial users of GNSS. Transmitting bogus GNSS signals to induce a receiver to report an incorrect time and position required radio frequency (RF) experts and pricey equipment. As a result, practically all spoofing occurrences were undertaken for academic research or nation-state-level electronic warfare. However, the emergence of low-cost, software-defined RF signal generators, along with a growing awareness of spoofing as a strong disruption tool, has resulted in a rise in the number of instances. Simultaneously, business organizations are increasing their reliance on GNSS-based positioning, navigation, and timing (PNT) systems.

Both technologies have the potential to conceal many aspects of a maritime vessel. For example, the cargo, its current location, destination, and route. Both of these technologies can and have been used maliciously. However, these are two sides of the same coin, as maritime vessels can use these technologies to protect themselves from potential threats. But hiding from threats can only go so far. The industry requires a more proactive shift in terms of maritime security. STIX / TAXII are aimed to help towards this endeavour.

3.4 Research Question #4

What is a maritime cybersecurity incident and how do they operate?

The maritime industry – including container ships, bulk tankers, drill ships, cruise ships, mobile offshore drilling units and the ports and terminals that support them – is critical to the global economy as well as national and international security. The global marine vessels market is projected to reach $220 billion USD by the end of 2026.1.
Today, the maritime industry is highly vulnerable to cybersecurity threats due to the integration of previously standalone OT systems, which physically control multiple systems onboard the ship, with IT systems that are deployed onboard and on shore. As the maritime industry continues to adopt cloud computing, IoT and autonomous technologies, interconnectivity between OT and IT will rapidly increase, leading to ever-higher cybersecurity risks. In fact, cyberattacks on the maritime industry's OT systems have already increased by 900 percent over the last three years.

When the experts were prompted to share their concerns about digital tools substituting part of the instrument learning process, they voiced their frustration with not only a lack of collaborative tools but also detriments of the features that are available.

Maritime cybersecurity incident is an event or occurrence on maritime interconnected IT and OT systems could have massive consequences, both regionally and globally.

3.4.1 What is the purpose of maritime cybersecurity incidents?

The maritime business, which includes ports and ships, is an important part of the worldwide supply chain for food, medicine, consumer goods, fuel, and a variety of other things. The majority of globally traded items are transported by water. As a result, maritime security is critical for supply chain safety. Maritime infrastructure tends to be outdated and complicated, thus further hampering the cybersecurity aspect.
Often, ships rely on various digital tools and systems in order to function, many of which are automated:

- Gyro compasses radars
- Magnetic compasses
- Autopilot mode
- ARPA
- Speed and distance log device
- Echo sounder
- Bridge systems
- Cargo handling and management systems
- Propulsion and machinery management and power control systems
- Access control systems
- Passenger servicing and management systems
- Passenger facing public networks
- Administrative and crew welfare systems

- Communication systems

A cybersecurity incident could control or shut down a ship or drive it off-course causing a crash or life-threatening situations as some ships transport dangerous and fragile cargo, such as: explosive fuel and toxic chemicals.

Despite, ships being the main and most common targets of the cyberattacks, ports are also heavily reliant on complex digital network logistic management systems, thus attackers are able to delay, erase the knowledge of, redirect or steal actual cargo by infiltrating the given systems. Oftentimes, digital attacks delay shipping and cause millions or billions of dollars' worth of damage to shipping companies, ports and shipping customers.

Nowadays ports have seen an increasing trend in automation and digitalization. This era of technological progression is commonly referred to as the 'Smart Port' generation. Port authorities are responsible for coordinating the implementation of new digital technology solutions to deliver connectivity, visibility and control, improving service across supply chains, including port community systems (PCS) to manage digital trade.

PCS entities are recognized as trusted "Orchestrators" or commonly referred to as third parties providing key 'one-to-many' business relationships and services that are at a high risk of being targeted with the intention of disruption in local, regional, national and even global supply chains.

Recently, the European Cyber Security agency (ENISA) has introduced four common cyberattacks scenarios at the port community level:

- **Scenario A:** Acquiring critical data to steal high value cargo or allow illegal trafficking through a targeted attack.
- **Scenario B:** Propagation of ransomware leading to a total shutdown of port operations.
- **Scenario C:** Compromise of port community systems for manipulation or theft or data.
- **Scenario D:** Compromise of operational technology systems creating a major accident in port areas.

3.4.2 What are the characteristics of maritime cyber incident? What makes something a maritime cybersecurity incident?

A measure of the extent to which a technology asset could be threatened by a prospective circumstance or event that could result in shipping-related operational, safety, or security problems as a result of information or systems being corrupted, lost, or compromised is known as maritime cyber risk.

Although, the cyberattack tactics targeting the maritime field are composed of common cyberattacks such as phishing, ransomware and malware attacks, the goals and targets of this attacks are what separate them and categorize them as maritime cyber incidents.

Generally, there are two types of cyberattacks that can affect a marine company or a ship: targeted and untargeted attacks. Untargeted attacks look for potential vulnerabilities in multiple companies and ships, whereas the targeted attacks are directed toward a specific company or ship and can prove exponentially more difficult to deter.

As mentioned in research question 3 in A recent example of a targeted maritime cyberattack is the ransomware attack on the Colonial Pipeline that took place on May 7, 2021. The Colonial Pipeline controls nearly half of the gasoline, jet fuel and diesel flowing along the East Coast of the United States. The pipeline was taken offline because there was no separation between data management and the pipeline's actual operational technology. As reported the attacker did not aim to take hold of the pipeline, but held data for ransom. The attacker, DarkSide, has caused $4.4 million dollars' worth of damage to the pipeline and proved the importance of enhancing the cybersecurity of the maritime field.

3.4.3 What are the most common maritime cyberattack tactics?

Nowadays, the most common maritime cyberattack tactics take the form of: (For more information see chapter 3.4.7)

- Malware
- Social engineering
- Phishing
- Water holing
- Port scanning
- Build-in software weaknesses
- Third party contribution
- Brute force
- DDoS
- Spear-phishing
- Subverting the supply chain

Although no business can predict which path an attacker would take across its network, hackers often use a series of steps to enter a network and steal data. Each step represents the attacker's progress toward a certain objective. The most common series of steps used by attackers nowadays are as follows:

Reconnaissance (identify the targets). The attacker assesses the targets from outside the organization to identify the targets that will enable him to meet his objectives. The aim of the threat actor is to identify and exploit vulnerabilities within the information systems and gain access into the target system.

Weaponization (prepare the operation). The attacker design and develop malware designed specifically for the vulnerabilities discovered during the previous phase.

Delivery (launch the operation). The attacker sends the malware to the target by any intrusion method, such as a phishing email, a man-in-the-middle attack or a watering hole attack.

Exploitation (gain access to victim). The threat actor exploits a vulnerability to gain access to the target's network.

Installation (establish beachhead at the victim). Once the hacker has infiltrated the network, he installs a persistent backdoor or implant to maintain access for an extended period of time.

Command and control (remotely control the implants). The malware opens a command channel, enabling the attacker to remotely manipulate the target's systems and devices through the network, thus enabling the attacker to take control of the entirety of the information system.

Actions on objectives (achieve the mission's goals). Now that the attacker has the command and control of the target's system, is entirely up to the attacker, who may corrupt or steal data, destroy systems or demand ransom, among other things.

Due to the changes in operations and procedures during the travel restrictions and operational hurdles encountered during the pandemic, the maritime industry has seen a rise in cybersecurity incidents and vulnerabilities. The most common cyberattacks have surfaced in the form of phishing attempts, malware and ransomware attacks.

3.4.4 How to identify a potential maritime cybersecurity incident?

Due to the increase in cybercrime, having an effective cyber security plan is becoming increasingly important to counter the ever-evolving security risks they face. It is essential that the plan implements strong information technology security tools, a strategy for emerging threats and education programs for staff since most attacks are caused by human error.
According to CISCO, the most common cyberattack tactics nowadays are as follows:

Mysterious emails

Email phishing refers to a method used by malicious actors to access sensitive information by pretending to be a trusted organisation or website.

Unusual password activity

If an employee is locked out of their system and/or receives an email stating that a password has been changed, it is a potential sign that the password is compromised if they did not initiate any of this action.

Identify suspicious pop-ups

Employees should avoid clicking on unknown pop-ups as this pop-ups can be infected with malware or spyware that can compromise the network.

Slower than normal network activity

A hacking attempt or malware outbreak often times results in spikes in network traffic that can reduce the internet speed.

3.4.5 What is a maritime cybersecurity incident categorization?

Confusion about threats and response options originates in part from impression in how the government comprehends and measures the range of disruptive cyber events. By failing to recognize the distinctions between specific forms of attack, the effects they produce on the targeted networks, and financial strain they place on the targeted organizations, or their broader effects on society, this confusion leads to the misallocation of resources. Therefore, profound findings are needed for deepen the analysis by sector to assess the risk for specific organizations and critical infrastructure. Thus, cyber incident categorization is an attempt to systematically arrange cyber incidents in groups or categories to established criteria specifically in order to get a better understanding of the event.

Maritime cyber incident categorization is the type of incident categorization that specifically concentrates on the maritime sector of industry.

3.4.6 What is the purpose of maritime cybersecurity incident classification?

As described above, the cyber incident classification is used to analyse over thousands of publicized cyber events by industrial sector. After arranging the incidents into classes or categories, it will enable the analyst to run a search for knowledge in the form of incidents, problems, or known errors. When an incident can be categorized, the search against previous knowledge is more effective. This will provide the analyst with the ability to track similar incidents related to the current one.

3.4.7 What are the different types of maritime cybersecurity incident? How many incident categories are there?

There are 4 types of maritime cyberattacks according to its general intended victims and areas:

1. Untargeted attacks (shotgun approach) – they look for potential cyber weak spot in multiple companies or ships
2. Targeted attacks – are directed toward a specific company or ship and can be harder to deter
3. Intentional attacks – where the cyber breach comes from intentional malicious actions
4. Unintentional (accidental) attacks – where the breach is an effect of negligence or ignorance

	Intentional	*Unintentional*
Targeted	Brute force Denial of service Spear-phishing Subverting the supply chain Port scanning	Falling victim to social engineering Escaped proof-of concept, runaway pentest
Untargeted	Malware Phishing Water holing Scanning	User error

Table 3.1: Categorizing the attacks

The cyber incidents targeting the maritime sector reflect the diverse motivations of the attackers:

- Weakening the economy of a country
- Protesting the political decision
- Sabotaging opposition
- Spying out information
- Stealing money or cargo
- Smuggling

3.4.7.1 Cyber Threats Classification According to ENISA

A cyber or cybersecurity threat is a malicious act that seeks to damage, steal data, or disrupt digital life in general. Any cyber threats that can cause a disturbance in the IT system can be called a cyber incident.

Below is the threat taxonomy introduced by ENISA [3] to bring the attacks analysed in order. They derived sub-categories of threats out of port hazards. Even though attack goal, source of infection and other possible consequences may be linked back to different threats, these are the main threats to group the attack. The main threat categories are included as below:

- Abuse and theft of data
- Data manipulation
- Denial of service
- Geo-localisation signal spoofing and jamming
- Interception of emission
- Different malware categories
- Man-in-the-middle
- Social engineering, phishing and identity theft
- Targeted attacks
- Unknown attack vectors
- Vulnerabilities exploitation of systems or devices

3.4.7.2 Abuse and Theft of Data

For various reasons data from maritime participants can be valuable for attackers. When hackers steal sensitive data or abuse certificates used in port operations like customs, the availability and integrity of these operations can be violated. In most cases, it was not clear how the attackers got the data, whether by malware, credential theft or by other attack vectors.

3.4.7.3 Data Manipulation

Shipping companies rely on their data sources about vessels and customer information. Ships need reliable nautical data to navigate. Manipulating data unnoticed or even visibly is attractive for attackers to achieve goals including finding valuable goods, sabotage companies, or deface companies.

3.4.7.4 Geo-location Signals Spoofing and Jamming

Geo-localisation signals (GPS) and navigation systems can be spoofed or jammed in order to hinder navigation and change the trajectory of a vessel [3].

3.4.7.5 Interception of Emission

For many reasons, hackers would like to intercept the communication between stakeholders like port and vessels. Often such interceptions are used to trick victims into sending money to the wrong target.

3.4.7.6 Malware

Since electronic components with common operating systems are part of ships and harbour control infrastructure, they can be infected with malware that also infects company systems in other sectors. The abilities and damage skills of malware vary and are therefore divided into different sub-categories. Ransomware will have its own part as its relevance has increased significantly in recent years. Also, a lot of malware infections are part of the attack vectors from targeted attacks in section 4.9 therefore we do not mention them again here. Most of the malware incidents we found were caused by ransomware. In some other cases, too little information is known to be able to say whether ransomware or other malware was the cause.

3.4.7.7 MitM (Man-in-the-Middle)

In a MitM attack, an attacker inserts himself into a communication and pretends to one or both communication partners that they are talking directly to the other. Thereby the attacker can normally read and modify the communication.

3.4.7.8Targeted Attacks

Targeted attacks use infection strategies and attack vectors tailored to one specific target and are usually scheduled over a longer period. Often a combination of social engineering, phishing and public knowledge about the target is used for infection with a broad number of tools ranging from data theft tools and remote-access Trojans to destructive malware.

3.4.7.9 Unknown Threats

In some cases, only the damage of the attack or the fact that an attack took place are known. This can be triggered by the fact that the attackers removed all evidence, the attack was not an attack but an incident, the system monitoring is too incomplete to reconstruct an attack, or the company hides the investigation results for some reasons. In most cases, the minimal damage is a temporary shutdown of servers and other IT infrastructure.

3.4.7.10 Technical Classification of Cyber Incidents

This classification is based on the types of cyber-attacks that has been occurred. This relies heavily on what technologies they used, how do they work, and the characteristics of their features

- **Malware:** is designed to gain access or cause damage to a computer, server or network without the knowledge of the victim. The term is used to determine many threats in the Internet landscape and contains unique traits and characteristics. Hackers, people who use their knowledge in coding to bypass security measures and harm a computer, device or network, mostly create Malware. The purpose of Malware is to steal resources from a computer and to exploit known deficiencies and problems of the network (for example, an outdated or unpatched business software)
    - Virus - Very similar to viral illnesses from which it takes its name virus is a program that can create a copy of itself and spread to other connected computers. Viruses often spread to other computers by attaching themselves to legitimate programs or documents and executing code when a user launches one of those programs.
    - Trojan horse – these bits of malware hide in what looks like a harmless software
    - Ransomware - with Ransomware threat actors can lock up a computer holding it “hostage”, encrypt files and force users to pay ransom in order to get their files back
    - Spyware - this type of Malware is designed to hide on a computer and monitor everything the user does. It can track web activity, access E-mail and even steal usernames and passwords
    - Worms - main goal of a computer worm is to make as many copies as possible of itself in any way possible from computer to computer. A worm can replicate itself without any human interaction and does not need to attach itself to a program in order to cause damage. Worms can modify and delete files and even inject additional Malware into the computer

- **Social engineering**: A non-technical practice used by potential cyber attackers to manipulate insider individuals into breaking security procedures, normally, but not exclusively, through interaction via social media.
  This type of attack can either be fake internal notices from the company scare tactics that force the user to respond immediately due to an emergency. Clicking on that link can trigger a malicious software download.
- **Phishing**: sending E-mails to many potential targets asking for pieces of sensitive information, phishing, is designed to fool the user and claim confidential data. Such an E-mail may also ask for that person's username, password, PIN number or make the victim visit a fake website by using the provided hyperlink.
- **Water holing**: this type of attack is created not to get to the user but for the user to fall victim to the attack. Water holing attacks are the development of fake websites based on a user's or personnel's interests or a compromise of a genuine website to exploit visitors and gain access to their data. They are untargeted, intentional attacks.
- **Port Scanning**: ports are a software construct that an application will use to communicate through the host operating system and out over the Internet. A port scanning attack occurs when an attacker sends packets to a computer, varying the destination port. The main goal of that attack is to check which ports the user has opened for incoming connections.
- **Built-in software weaknesses**: vulnerabilities that relate to insufficient control of users that have access to the systems or wrong sanitizing of users input before passing them in the database.
- **Third party contribution**: having access to the company's systems, vendors or service technicians can leave vulnerable backdoors that the company is unaware of.
- **Brute force**: An attack trying many passwords with the hope of eventually guessing correctly. The attacker systematically checks all possible passwords until the correct one is found. This practice is the reason almost all websites require passwords with puzzled combinations of letters and numbers and evaluate their security level.
- **DDoS**: is designed to prevent legitimate and authorized users from accessing information, usually by flooding the target with a constant flood of traffic from different sources. It aims to disrupt normal operation on a specific server or network
- **Spear-phishing**: like phishing, but individuals are targeted with personal E-mails, containing malicious software or links.
- **Subverting the supply chain**: attacking a company or ship by compromising equipment, software or supporting services by delivered to the company or ship

3.4.7.11 Designation of Malwares That Has Occurred in the Maritime Industry

- NotPetya - ransomware, happened in Ukraine 2017
- SamSam - ransomware, is a custom infection used in targeted attacks, often deployed using a wide range of exploits or brute-force tactics)
- Emotet
- Ragnar Locker
- Medusa

3.4.7.12 Classification Based on Threat Actors

There are different threat actors or threat actors' groups with a variety of motivations for performing malicious acts against a company. The most noteworthy is that some groups are motivated by monetization of cybercrime, and some are motivated by political, ideological or religious reasons. The accomplishment of the attack is depending on the purpose and interest of those groups [2].

Activist Group

Often referred to as hacktivist, these groups comprise ideologically motivated individuals that may form dynamic groups or sub-groups. Their actions are effectively online protests, which may have the aim of disrupting systems or acquiring confidential or sensitive information for publication or dissemination so as to embarrass their targets.

The impact of small activist groups can be significantly magnified when, as some groups have demonstrated, they recruit or persuade naïve third parties to join in by allowing the installation of malicious software on the recruits' computers, creating backdoors to the systems. Specifically, in the Maritime domain, hacktivists are seeking publicity or creating pressure on behalf of a specific objective or cause, for example, to prevent handling of specific cargoes or to disrupt the operation of a ship or a third party such as the supplier or recipient of the cargo.

Competitors

This group is typically large corporations seeking to create competitive advantage. They may act directly or through third parties, with the aim of harming a rival by collecting business intelligence, stealing intellectual property, gathering competitive intelligence on bids or disrupting operations to cause financial or reputational loss. Depending on size, sector, geographic location and the sophistication of a large corporation's cyber capabilities they may be able to perform sophisticated malicious activities to target and infiltrate their competitors.

Cyber criminals

The cybercriminals utilize advanced methods, tools and software to profit from their illegal activities. These are highly skilled criminal groups trying to perpetrate a wide range of resources while using sophisticated techniques. The motivation lies to profit from illegal activities as well as "show-of-skill", and their focus has mainly been on fraud and theft. However, cyber-criminal activities also include blackmail and extortion through use of Ransomware to encrypt data or threats of denial-of-service attacks on corporate websites. The methods and techniques evolve with the advancement in technology and business, such as e-finance, e-commerce, and e-payment. The anonymization, encryption and virtual currencies, such as Bitcoin, makes the cybercriminals difficult to identify and their attacks hardly detectable. In respect of ports, cybercriminals may seek to intercept or access information related to cargo shipments or to security arrangements as a precursor to criminal activities or a physical attack on these premises [9]. Cybercriminals have created a business around cybercrime as-a-service and can potentially be involved in espionage-as-a-service [8]. The sophistication of the Malware.

Terrorists

Cyber Terrorism poses a significant risk because of the increased dependency upon information technology by organizations. Terrorists are becoming increasingly IT aware and seek to distribute propaganda.

The objective behind such an attack is to take control of major IT infrastructure, spread a Malware, encrypt, or steal confidential data, commit fraud, and/or carry out any act with the aim of damaging IT assets. There are cases where well-funded groups could take advantage of the service offered by cybercriminals, seek support from a nation-state or encourage internal members to adopt these methods of attack. With the widespread use of electronic and computer-based technologies in the Maritime environment, terrorist groups could rely on the various toolkits available for download to disrupt or damage ships by attacking ship and/or connected shore-based systems. Terrorists may also exploit poorly secured ship data to enable remote hostile reconnaissance of targets, thus reducing the time they need to spend in or near their target.

A vessel full of gasoline could be a highly attractive target for cyber terrorists who want to spread fear and cause physical and economic disruption to a port or to another vessel.

Nation states and state-sponsored threat actors

It is acknowledged that some nation states are actively involved in cyber-attacks on a wide range of organizations in order to acquire state secrets or sensitive commercial information. They may also threaten the availability of critical infrastructure in other nation states. During periods of heightening international tension and conflict, these activities may include more widespread attacks as evidenced by Malware such as Stuxnet and WannaCry. The state-sponsored threat actors effectively have the capacity and sophisticated technical support available to a nation-state made available by the sponsoring nation. This group could include cyber fighters, i.e. groups of nationally motivated individuals who threaten or attack other groups, businesses and the infrastructure of other nation states. The cyber fighters may be seen as a type of hacktivist, but their interest is the support of a nation-state and as such, they may enjoy significant sophisticated technical support from that nation-state.

Group	Motivation
Accidental Actors	• No malicious motive but still end up causing unintended harm through bad luck, lack of knowledge or lack of care, e.g. by inserting infected USB in onboard IT or OT systems
Activists (including disgruntled employees)	• Revenge • Disruption of operations • Media attention • Reputational damage
Criminals	• Financial gain • Commercial espionage • Industrial espionage

Opportunist	- The challenge - Reputational gain - Financial gain
Terrorists State-sponsored organisations States	- Political/ ideological gain e.g. (un)controlled disruption to economies and critical national infrastructure - Espionage - Financial gain - Commercial espionage - Industrial espionage - Commercial gain

Table 3.2: Actos and their motivation

3.4.7.13 Human factor: the weakest link

Looking at the list of maritime cyber incidents that has occurred over the years, one cannot simply ignore the factor of human error

Even in the most secure system, there is a vulnerability which cannot be patched, corrected, or rewritten. While the people keep interacting with systems onshore and on board, the human element will still play a significant role in most cyber security incidents. During the Maritime Transport Summit at DTU, the Head of Maritime Technology Regulation at BIMCO mentioned *"80% of the cybersecurity incidents could have been prevented if single users were able to recognize the threats. It is vitally important to educate the crew on board to raise awareness about the vulnerabilities arising from human error".*

People that can cause cyber-breach are usually the ones that interact with the IT or OT such as:

- Shareholder/ owners
- Management
- Employees
- Business Partner
- Service Providers
- Contractors
- Customers/ Clients

The severity and sophistication of the threat is determined by the individual's capabilities, for example:

- A negligent and careless employee or partner
- Non-malicious individual
- A disaffected employee or contractor with limited IT skills
- Script kiddies – individual hackers with limited knowledge who use techniques and tools devised and developed by other people
- Lone wolf – individuals outside of the organisation possessing advanced technical knowledge

3.4.8 Can a maritime cyber incident belong to more than one category?

Speaking from theory, then yes. One incident can belong to different taxonomy of categories. It can have a specific threat actor, with different intended target attack using a special malware and causes a certain level of disruption.

3.4.9 What is the categorization of cyber incidents according to conventional taxonomy (NCSC: National Cyber Security Centre)?

Below is the categorization of cyber incidents by NCSCN and law enforcement to implement new cyber incident prioritisation framework [4]. Keep in mind that these categories are not regarding maritime cyber incident specifically but more towards the general approach. NCSC is a part of GCHQ (Government Communication Headquarters) which is an intelligence and security organisation to the government of United Kingdom. And thus, this information provides an information for cyber security at a national level in the UK's technical authority on cyber. Primarily, the NCSC defines cyber security incident as:

- A breach of a system's security policy to affect its integrity or availability
- The unauthorised access or attempted access to a system

And activities or common traits associated and commonly recognised as cyber incidents are:

- Attempts to gain unauthorised access to a system and/or to data
- The unauthorised use of systems and/or data
- Modification of a system's firmware, software, or hardware without the system-owner's consent
- Malicious disruption and/or denial of service

They categorize the cyber incidents by the type of priorities according to how severe they are and to whom they are being responded to.

	Category Definition	**Who responds?**	**What do they do?**
Category 1 National cyber emergency	A cyber-attack which causes sustained disruption of UK essential services or affects UK national security, leading to sever economic or social consequences or to loss of life	Immediate, rapid and coordinated cross-government response. Strategic leadership from Ministers/ Cabinet Office (COBR), tactical cross-government coordination by NCS, working closely with Law Enforcement	Coordinated on-site presence for evidence gathering forensic acquisition and support. Collocation of NCSC, Law Enforcement, Lead Government Departments, and others where possible for enhance response
Category 2 Highly significant incident	A cyber-attack which has a serious impact on central government, UK essential services, a considerable proportion of the UK	NCSC (escalated to COBR if necessary), working closely with Law Enforcement (NCA) as required. Cross-	NCSC will often provide on-site response, investigation, and analysis, aligned with Law Enforcement

	population, or the UK economy	government response coordinated by NCSC	criminal investigation activities
Category 3 Significant incident	A cyber-attack which has a serious impact on a large organisation or on wider/ local government, or which poses a considerable risk to central government or UK essential services	NCSC, working with Law Enforcement (NCA) as required	NCSC will provide remote support and analysis, standard guidance; on-site NCSC or NCA support may be provided
Category 4 Substantial incident	A cyber-attack which has a serious impact on a medium-sized organisation, or which poses a considerable risk to a large organisation or wider/ local government	Either NCSC or Law Enforcement (NCA or ROCU), dependent on the incident	NCSC or Law Enforcement will provide remote support and standard guidance, or on-site support by exception
Category 5 Moderate incident	A cyber-attack on a small organisation, or which poses a considerable risk to a medium-sized organisation or preliminary indications of cyber activity against a large organisation or the government	Law Enforcement (ROCU or local Police Force), with NCA input as required	Law Enforcement will provide remote support and standard guidance, with on-site response by exception
Category 6 Localised incident	A cyber-attack on individual, or preliminary indications of cyber activity against a small or medium-sized organisation	Automated Protect advice or local response by Law Enforcement (local Police Force)	Remote support and provision of standard advice. On-site response by exception

3.4.10 What are ways to categorize maritime cybersecurity incidents? Can they be categorized in one way?

It is highly possible for a new category or a completely new classification of cyber incident to appear soon as there is no end to the cyber-attacks and hackers are keep attempting to find a breach or implementing a new attack. Kjaerland [5], classifies that there are 4 main effects from various kinds of cyber incidents that will remain the same: Disrupt, Distort, Destruct, and Disclosure.

A good taxonomy must have successfully met the basic standards of well-defined taxonomy [7], as including:

Exclusiveness – No two categories should overlap or should have the same scope and boundaries.

Ascertain-ability – Each category should be definitively and immediately understandable from its name.

Consistency – The rules for making the selection should be consistently adhered to.

Affinity and Context – As the writer move from the top of the hierarchical classification to the bottom, the specification of the classification should increase.

Currency – Names of the categories in the classification should reflect the language in the domain for which it is created.

Differentiation - When differentiating a category, it should give rise to at least two subcategories.

Exhaustiveness – The classification should provide comprehensive coverage to the domain of interest.

3.4.11 Is there a type of prioritisation for categorizing maritime cyber incidents?

A confusing array of cyber threat classification systems have been proposed over the past two decades. Some are based on distinct phases of the hacking process, while others focus on specific targets. While classifying the incidents, few points should be taken into consideration such as: specific techniques that is used (like DDoS attacks), specific targets, and particular IT capabilities.

Other ways to classify the attacks that have been going on are the impact on the target (the key question for risk assessment), effect of the victim, the attack vector, vulnerabilities, and exploits, while also incorporating on the effect categories as part of a broader classification system.

4 Conclusion

The conclusion will consist out of three separate conclusions for each distinct topic.

4.1 Research question 1

Throughout research, exploration and testing shows that ASP .NET performs faster and more stable in most of the tested cases, not only in the ones required by the STIX platform. Moreover, .NET has better integration with multiple types of databases as well as better security practices. Given its agnostic nature, .Net makes it easier to build a self-contained application, that can be containerised and developed/maintained independently, along with its multithread approach which results in better performance when running/compiling. Adding to this MS Office built in tools and JSON serialising, both requirements of the clients. Many companies that have developed large scale apps for years prefer .NET, such as Dell, Intel, Cisco or Siemens.

While Node has a clear advantage when it comes to single language integration, where one can use JavaScript for both back end and front end, the performance, database integration, security and scalability, building a web application for the STIX DATABASE using .NET is the more logical choice.

4.2 Research question 2

Based on the given data, you can see that different use cases require different technologies. For our specific requirements we can already exclude MariaDb, since it is used with PHP, a technology which we are avoiding in the first place (the old platform was built in php). Then we are left with the two options MqSql and MongoDb. MySql is slower than MongoDb, but it is better for storing large files, therefore the best solution the team could produce is a combination of both, accessing both the speed, scalability, reliability, efficiency, and storage of both databases.

The ideal solution is a main MongoDb database, so the queries could be executed faster, and then the images, and larger files being stored on MySql(since it is more suitable for larger files), while connecting MongoDb and Mysql, with the id of each image/file being stored in Mongo. That way we will remain with the fast connection, and the only thing that could be slowed down is the loading of the images/files, which is not of upmost importance. Using 2 databases also ensures that if one of the databases has an exploit and gets hacked, the other one will remain inaccessible by the hacker, thus providing an extra security layer.

4.3 Research question 3

Ultimately, the research question, “How can the STIX methodology and its systems be utilized to serve (maritime) cyber security needs?”, was meant to evaluate how the OASIS Cyber Threat Intelligence Technical Committee’s language and serialization format is a benefit to the (maritime) cyber security sector.

Threat information can be transported across IT security and intelligence solutions via STIX and TAXII. The development of STIX/TAXII is an open, community-driven project that provides free specifications to aid in the automated expression of cyber threat information. Ultimately, STIX and TAXII aim to meet users where they are at, thus allowing for easy incorporation with existing networks, and adaption to fit specific needs in the categorization of incidents. This in turn enables for CTI sharing, storing and tracking for niche/specific fields that may be underrepresented in commercial channels of relevant exchange, while maintaining a standardized intelligence sharing format that is universally known and accessible. The reactive nature of one organization is the proactive nature of the other.

It is now easier to contribute to and consume cyber threat information. STIX/TAXII attempts to increase security measures in several ways:

- By increasing the capabilities of present cyber threat intelligence exchange.
- By balancing response with proactive detection.
- By encouraging a multifaceted approach to cyber threat intelligence.

All components of suspicion, compromise, and attribution may be properly represented with objects and descriptive relationships in STIX. STIX data can be visually rendered for analysts or kept as JSON for easy machine readability for automation. An application tailored to this system will ensure that future users only require basic tech literacy to fully benefit from this system.

4.4 Research question 4

The shipping industry is moving into the world of digitalization. Big Data Analytics, systems automation and IoT are some of the technological advances that the Maritime Industry is already using to acquire better control of their assets anytime and everywhere in the world. Along with this evolution of onboard systems, autonomous and unmanned ships an increasing risk in the maritime cyber security field has been observed.

The aim of the research was to conduct a holistic investigation on what the cyber maritime incident is that arose along with the adoption of those technological developments, how to identify them, and what are its categorization to familiarise with it. In the future, the cyber threats will become more prevalent, rendering cybersecurity a compelling requirement and competitive advantage among the industry.

Appendix 7.6 shows that increased information security incidents are reported publicly, which also correspondents to a general trend in any other industry. But in absolute terms, considering the size of the maritime sector and estimates of unreported incidents, still very few incidents are reported.
Most of incidents that are found belong to one of two kinds of attacks: First, there are attacks that target companies and vulnerable systems in general, and not specifically the maritime sector, for example to make money via ransomware. Second, there are targeted attacks in which specific maritime companies, such as shipyards, are targeted for espionage purposes or other political reasons. In most of these cases it is suspected that state sponsored actors may be involved, but in most cases, there are no concrete publicly available evidence.

Overall, the threats for participants of the maritime sector are rising and therefore the individual parties should cooperate to hold out against these threats. The attackers learned already how to coordinate, and the attack vectors are growing constantly.

5 Discussion

As the research is solely conducted during the documentation phase of the project, there were significant time-constraints that had a notable impact on the student's ability of sifting through and processing the large quantities of data provided within the respective STIX and TAXII specifications documents. Furthermore, the workings of STIX and TAXII was only one sub question out of total of four, thus the research pair could not focus all their attention on the specification documents. As a result, the information available about STIX/TAXII in this research document is condensed down to what the pair believed to be essential.

Notably, there is a definite lack of data comparison within this report, e.g. weighing the detriments and advantages of the STIX methodology against others, as the usage of STIX and TAXII was already decided on by the client(s), so for the purposes of this research, it was irrelevant to try to compare STIX/TAXII other methodologies or systems. Consequently, this research did not have a typical population & sample format. Additionally, due to a lacking understanding of the entirety of the technical scope of the intended project outcome, there is a strong potential for miscalculations when it comes to the prediction of needed proficiencies. The technologies and methodologies listed are suitable options for the development process, however there is no guarantee they are to be used in the realization phase.

Remarkably, there was a previous iteration of the web-application as mentioned in this document. But when inquired if there were any previous documentation, there were none. For the sake of future research and the project, this iteration will be documented in an effort to “keep the ball rolling”, as this will not be the final version of this product. This aspect also weighted a lot in choosing the technologies on what the platform will be built on, firstly by choosing a different architecture and secondly by thoroughly researching and testing the alternatives and setting on what suits the requirements the most

Cyber security is a well-known and documented part of the information technology field, notably the cybersecurity incidents on the maritime industry are still seldomly analysed and documented. Due to the pandemic’s travel constrictions and general rules the maritime industry has seen a remarkable rise in cybersecurity incidents.

The lack of reports and documentation about the detection of vulnerabilities and identification of maritime cybersecurity incidents, the research team had to rely on both general cybersecurity information and maritime reports, thus the risk of general cybersecurity information not applying to the maritime field rises.

Moreover, the research team could have interviewed experts in the maritime cybersecurity field for additional expertise and knowledge on the topic, especially as the client of the project is a PhD in the maritime cyber security industry.

6 References

This chapter details all the sources used as the research was conducted.

Generally, the references are listed in the APA format, however the STIX and TAXII specification documents that were used offered their own citation format that they requested to be used. These sources are marked with a "*" before the author and are listed twice, once in their preferred format and then again using APA.

6.1 References Research Question 1

GeeksforGeeks. (2020, March 27). *Difference Between Node.js and Asp.net*. GeeksforGeeks. Retrieved on March 16th from https://www.geeksforgeeks.org/difference-between-node-js-and-asp-net/

Hanselman, C. (n.d.). *WebMatrix 2 - Front End Web Developers take note (ASP.NET, PHP, node.js and more)*. Hanselman Retrieved March 25, 2022, from https://www.hanselman.com/blog/webmatrix-2-front-end-web-developers-take-note-aspnet-php-nodejs-and-more

Microsoft (n.d*.). Microsoft What is ASP.NET? | .NET.* Microsoft. Retrieved on March 12th from https://dotnet.microsoft.com/en-us/learn/aspnet/what-is-aspnet

Node.js. (2019). *Docs | Node.js. Node.js Foundation.* Node.js Retrieved on March 16th from https://nodejs.org/en/docs/

Rathnayaka, L. (2021, November 8). *Node.js vs .NET Core the winner?* Medium. https://levelup.gitconnected.com/node-js-vs-net-core-the-winner-5ba06efb4c35

ReadWrite. (2013, November 7). *What You Need To Know About Node.js.* ReadWrite. Retrieved on March 25th from https://readwrite.com/what-you-need-to-know-about-nodejs/

TechMagic (2021, April 20). *Node.js vs. .NET: What to Choose in 2022*. - TechMagic. Blog . Retrieved on March 15th from https://www.techmagic.co/blog/node-js-vs-net-what-to-choose/

TechMagic. (2020, July 23). *Node.js vs .NET: What to Choose in 2020?* TechMagic. Retrievede https://medium.com/techmagic/node-js-vs-net-what-to-choose-in-2020-3b53ddfebd28

6.2 References Research Question 2

Db-Engines.com. (n.d.). *MariaDB vs. MongoDB vs. MySQL Comparison.* Retrieved on: March 12, 2022, from: https://db-engines.com/en/system/MariaDB%3BMongoDB%3BMySQL

Geekflare (2019, December 11*).. MongoDB vs MariaDB vs MySQL.* Retrieved on: March 12, 2022, from: https://geekflare.com/mongodb-vs-mariadb-vs-mysql/

GeeksforGeeks. (2020, June 5). *Difference between MongoDB and MariaDB.* Retrieved on: March 12,2022, from:https://www.geeksforgeeks.org/difference-between-mongodb-and-mariadb/

Plesk (2021, June 2*).. MariaDB vs MongoDB: Which is Right for You?* Retrieved on March 12,2022, from:.. https://www.plesk.com/blog/various/mariadb-vs-mongodb/

6.3 References Research Question 3

Dinsmore, N., Salzman,D. (2021, October 24) *GNSS manipulation 2.0: Practice makes perfect.* Windward. Retrieved on March 10, 2022 from https://windward.ai/blog/gns-manipulation-2-0-practice-makes-perfect/

Disney, A. (2021, February 24) *Force-directed graph layouts explained. Cambridge Intelligence,* Retrieved on March 12th. 2022 from https://cambridge-intelligence.com/keylines-faq-force-directed-layouts/

IBM (2022, March 7) *HTTP Basic Authentication*. Retrieved on March 12, 2022 from https://www.ibm.com/docs/en/cics-ts/5.4?topic=concepts-http-basic-authentication

IMO (2019) *Maritime Security*. Retrieved on March 10th, 2022 from https://www.imo.org/en/OurWork/Security

Minnick, J. (n.d.) *STIX 2.1 Drag and Drop Modeler*. GitHub. Retrieved on March 8th, 2022 from https://github.com/STIX-Modeler/UI

MITAGS (2021, June 25) *Guide to Maritime Security*. Retrieved on March 9, 2022 from https://www.mitags.org/security-guide/

OASIS Open. (2021, October 4). *Introduction to TAXII.* Retrieved on March 4, 2022 from https://oasis-open.github.io/cti-documentation/taxii/intro

OASIS Open. (n.d.). *Introduction to trusted automated eXchange of intelligence information (TAXII).* Retrieved on March 4, 2022 from https://oasis-open.github.io/cti-documentation/docs/Introduction_to_Trusted_Automated_eXchange_of_Intelligence_Information.pdf

Oasis Cyber Threat Intelligence, (CTI) TC. (2021, June 10). *TAXII version 2.1.* Retrieved on March 4, 2022 from https://docs.oasis-open.org/cti/taxii/v2.1/os/taxii-v2.1-os.pdf

*STIX Version 2.1. Edited by Bret Jordan, Rich Piazza, and Trey Darley. 10 June 2021. OASIS Standard. https://docs.oasis-open.org/cti/stix/v2.1/os/stix-v2.1-os.html. Latest stage: https://docs.oasisopen.org/cti/stix/v2.1/stix-v2.1.html.

*TAXII™ Version 2.1. Edited by Bret Jordan and Drew Varner. 10 June 2021. OASIS Standard. https://docs.oasis-open.org/cti/taxii/v2.1/os/taxii-v2.1-os.html. Latest stage: https://docs.oasisopen.org/cti/taxii/v2.1/taxii-v2.1.html

Windward. (2020, December 30*). AIS spoofing: new technologies for new threats*. Retrieved on March 18th 2022 from https://windward.ai/blog/ais-spoofing-new-technologies-for-new-threats/#:~:text=AIS%20spoofing%20is%20the%20deliberate

6.4 References Research Question 4

[1] 2016 cyber security survey in association with BIMCO. (September):1017206, 2016.

[2] Passeri, P. , (2018). *2017 cyber-attacks statistics.* Retrieved on March 15th 2022 from https://www.hackmageddon.com/2018/01/17/2017-cyber-attacks-statistics/

[3] ENISA (European Union Agency for Cybersecurity). (2019, November 26). *Port Cybersecurity. Good practices for cybersecurity in the maritime sector*. Retrieved on March 10th from https://www.enisa.europa.eu/publications/port-cybersecurity-good-practices-for-cybersecurity-in-the-maritime-sector\

[4] NCSC (National Cyber Security Centre). (2018). *New Cyber Attack Categorisation System to Improve UK Response to Incidents.* Retrieved on March 12th 2022 from https://www.ncsc.gov.uk

[5] CSO Alliance Maritime. Maritime cybercrime reporting portal, (2017). Retrieved on March 19th 2022 from https://www.csoalliance.com/page/maritime-cyber-crime-reporting-portal.

[6] Ranganathan, S. R. (1957). Prolegomena to library classification. London: The Library Association

[7] Kjaerland, M., (2005) *"A taxonomy and comparison of computer security incidents from the commercial and government sectors"*. Computers and Security, Vol 25 pgs 522–538.

[8] International Association of Ports and Harbors (IAPH), ICHCA International, TT Club established expertise, & WPSP. (2020, June). *IAPH Port Community Cyber Security Report.* World Ports Sustainability Program. Retrieved on March 15th 2022 from https://sustainableworldports.org/wp-content/uploads/IAPH-Port-Community-Cyber-Security-Report-Q2-2020.pdf

[9] Rosencrance, L. (2019, June 28). *10 types of security incidents and how to handle them*. Retrieved on March 11th, 2022 from SearchSecurity. https://www.techtarget.com/searchsecurity/feature/10-types-of-security-incidents-and-how-to-handle-them

[10] *Huffpost (2017, February 22) 5 Ways To Detect A Cyber Attack*. Huffpost.Com. Retrieved on March 19th 2022 from https://www.cisco.com/c/dam/m/en_ca/business-transformation/pdf/5-ways-to-detect-a-cyber-attack.pdf

7 Appendices

7.1 Appendix 1 – STIX Domain Objects

7.2 Appendix 2 – STIX Cyber Observable Object

7.3 Appendix 3 – STIX Relationships

7.4 Appendix 4 – STIX Patterning

7.5 Appendix 5 - TAXII

7.6 Appendix 6 – Number of publicly reported incidents per year

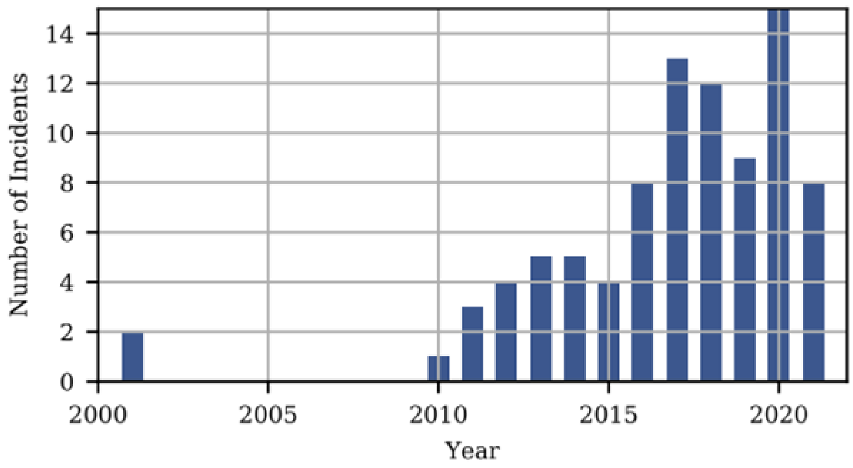


Figure 8: Number of publicly reported incidents per year

7.7 Appendix 7 - List of Publicly Reported Incidents

Threat Category	Year		Target	Threat Actor	Source
Abuse and Theft of Data	2012		Australian Customs and Border protection	Crime Syndicate	
	2013		General Dynamics		
	2014		Unknown		
	2016		US Navy Subcontractor		
	2016		DCNS		
	2016		Shipping company		
	2016		Ports	BRpsd	
	2016		DSME	North Korea	
	2017		Australian Defence Subcontractor		

	2017		Clarksons		
	2017		Svitzer		
	2018		US Navy Subcontractor	China	
	2018		Austal		
	2019		Holland America Line		
	2019		London Offshore Consultants	Maze Group	
	2020		DSME		
	2021		Carnival		
Data Manipulation	2014		Massachusetts Maritime Academy	Moroccan Electronic Islamic Union	
DDoS	2001		Port of Houston	UK Teenager	
	2001		Port of Houston	US Citizen	
	2013		Port of Long Beach		
	2017		Port of Vancouver		
Geo-localisation Spoofing	2001		Unspecified	PoC	
	2009		Unspecified	PoC	
	2010		South Korea	North Korea	
	2012		Unknown	North Korea	
	2013		A single yacht	PoC	
	2013		Ship owners, Ship operators	PoC	
	2013		Generally Ships	PoC	
	2016		Generally Ships		
	2017		Generally Ships		
	2018		Commercial Vessels		
	2018		Generally Ships	PoC	
	2018		Generally Ships	PoC	
	2019		Generally Ships		
Interception of emissions	2014		World Fuel Services		
Malware	2013		Mobile Offshore Drilling Unit		
	2017		JNPT	Sandworm	
	2017		Maersk Line, APM Terminals, Damco	Sandworm	
	2017		Port pf Rotterdam	Sandworm	
	2018		Cosco	Sandworm	
	2018		Port of San Diego	Iran	
	2018		Naval Industry		
	2019		Pitney Bowes		
	2019		A single vessel		
	2019		Ports	PoC	
	2019		MTSA regulated facility		
	2020		Toll Group		
	2020		CGA CGM		
	2020		MSC Cargo		
	2020		Blue Water Shipping		
	2020		Carnival		

	2020		Anglo-Eastern		
	2020		Hurtigruten		
	2020		Toll Group		
	2021		Rubin Design Bureau	Chinese APT	
Man in the Middle	2013		Container carriers	PoC	
	2015		Container carriers	PoC	
Social Engineering, Phishing and Identity Theft	2013		Major Shipping & Trading Inc		
	2014		Container carriers	PoC	
	2015		Shipping company		
	2015		Nautilus Minerals, Marine Assets Corporation		
	2016		Seafarer		
	2016		Shipping company		
	2017		Shipping company		
	2018		Bukner Fuel Company		
	2019		Commercial Vessels		
	2020		Tug Operating Organization		
	2020		Maritime Sector		
	2021		Ship broker; Bunker Supplier		
	2021		MSC Cargo; Shipping company		
	2021		HARMONIA Schiffscholding AG		
	2021		Shipping company		
Targeted Attack	2011		Port of Antwerp	Crime Syndicate	
	2011		Shipbuilding, Maritime Operations	Icefog	
	2011		Islamic Republic of Iran Shipping Lines		
	2012		Danish Maritime Authorities	China	
	2012		Transportation companies		
	2013		Hyundai Merchant Marine	Kimsuky	
	2014		Logistic companies		
	2014		Maritime Sector	Leviathan	
	2015		Maritime Sector	Leviathan	
	2017		A single vessel		
	2018		Shipbuilding		
	2018		Government institutions	Carbanak Group	
	2018		Shipbuilder	Leviathan	
	2018		Government institutions	Gamaredon Group	
	2019		Hamburg shipping company		
	2020		Ports		
	2020		Hamburger logistic company		
Unknown Threat	2015		TKMS, DCNS, Mitsubishi Kawasaki HI		

	2017		BW Group		
	2018		Port of Barcelona		
	2019		James Fisher and Sons		
	2020		Shahid Rajaei Port		
	2020		Bourbon		
	2021		Hyundai Merchant Marine		
Vulnerabilities	2013		Lockheed Martin	PoC	
	2014		US Navy	PoC	
	2015		Maritime Sector	PoC	
	2017		A single vessel	PoC	
	2017		VSAT systems	Researcher	
	2020		A single vessel	Researcher	

7.7 Appendix 7 - Leeuwarden Group - 150 Maritime Incidents Reports