NOVA SCHOOL OF SCIENCE & TECHNOLOGY

DEPARTMENT OF COMPUTER SCIENCE

FRANCESCO LUPO ELISA

Master in Computer Science

A SITUATIONAL AWARENESS DASHBOARD FOR A SECURITY OPERATIONS CENTER

COMPUTER SCIENCE

NOVA University Lisbon
September, 2021

NOVA SCHOOL OF SCIENCE & TECHNOLOGY

DEPARTMENT OF COMPUTER SCIENCE

A SITUATIONAL AWARENESS DASHBOARD FOR A SECURITY OPERATIONS CENTER

FRANCESCO LUPO ELISA

Master in Computer Science

Adviser: Paulo Rosado
Cybersecurity Consulting Senior Manager, EY Portugal

Co-adviser: Pedro Abílio Duarte de Medeiros
Associate Professor, NOVA University Lisbon

COMPUTER SCIENCE

NOVA University Lisbon
September, 2021

A Situational Awareness Dashboard for a Security Operations Center

Para a minha mãe Ana e o meu irmão Stefano, que foram o meu rochedo e aturaram os momentos mais difíceis desta jornada. Para a Maria, a Filipa, o João, as Catarinas e a Marta que me deram os melhores conselhos e foram a melhor companhia virtual neste ano muito diferente e difícil.

Acknowledgements

I would like to thank the following people, without whom I would not have been able to complete this research.

To both my adviser Paulo Rosado and co-adviser Pedro Medeiros, whose insight and knowledge into the subject matter steered me through this research.

I would also like to thank every member of EY's Cybersecurity Staff who helped me during the course of this dissertation both academically and professionally. Their valuable insight into the matter was key towards successfully completing this dissertation.

Abstract

As a result of this dissertation, a solution was developed which would provide visibility into an institution's security posture and its exposure to risk. Achieving this required the development of a Situational Awareness Dashboard in a cybersecurity context. This Dashboard provides a unified point of view where workers ranging from analysts to members of the executive board can consult and interact with a visual interface that aggregates a set of strategically picked metrics. These metrics provide insight regarding two main topics, the performance and risk of the organization's Security Operations Center (SOC).

The development of the dashboard was performed while working with the multinational enterprise entitled EY. During this time frame, two dashboards were developed one for each of two of EY's clients inserted in the financial sector. Even though the first solution did not enter production, hence not leaving testing, the dashboard that was developed for the second client successfully was delivered fulfilling the set of objectives that were proposed initially.

One of those objectives was enabling the solution to be as autonomous and self-sustained as possible, through its system architecture. Despite having different architectural components, both solutions were based on the same three-layered model. Whereas the first component runs all data ingestion, parsing and transformation operations, the second is in charge of the storage of said information into a database. Finally, the last component, possibly the most important one, is the visualization software tasked with displaying the previous information into actionable intelligence through the power of data visualization.

All in all, the key points listed above converged into the development of a Situational Awareness Dashboard which ultimately allows organizations to have visibility into the SOC's activities, as well as a perception of the performance and associated risks it faces.

Keywords: Cybersecurity, Situational Awareness, Dashboard, Security Operations Center, Risk and Data Visualization

Resumo

Como resultado desta dissertação, foi desenvolvida uma solução que proporcionaria visibilidade sobre a postura de segurança de uma instituição e sua exposição ao risco. Para tal foi necessário o desenvolvimento de um *Situational Awareness Dashboard* num contexto de cibersegurança. Este *Dashboard* pretende fornecer um ponto de vista unificado onde os trabalhadores, desde analistas a membros do conselho executivo, podem consultar e interagir com uma interface visual que agrega um conjunto de métricas escolhidas estrategicamente. Essas métricas fornecem informações sobre dois tópicos principais, o desempenho e o risco do *Security Operations Center (SOC)* da organização.

O desenvolvimento do *Dashboard* foi realizado em parceria com a empresa multinacional EY. Nesse período, foram desenvolvidos dois dashboards, um para cada um dos dois clientes da EY inseridos no setor financeiro. Apesar de a primeira solução não ter entrado em produção, não saindo de teste, o painel que foi desenvolvido para o segundo cliente foi entregue com sucesso cumprindo o conjunto de objetivos inicialmente proposto.

Um desses objetivos era permitir que a solução fosse o mais autónoma e auto-sustentável possível, através da sua arquitetura de sistema. Apesar de terem diferentes componentes arquiteturais, ambas as soluções foram baseadas no mesmo modelo de três camadas. Enquanto a primeiro componente executa todas as operações de ingestão, análise e transformação de dados, a segundo é responsável pelo armazenamento dessas informações numa base de dados. Finalmente, o último componente, possivelmente o mais importante, é o software de visualização encarregue em exibir as informações anteriores em inteligência acionável através do poder da visualização de dados.

Em suma, os pontos-chave listados acima convergiram no desenvolvimento de um *Situational Awareness Dashboard* que, em última análise, permite que as organizações tenham visibilidade das atividades do SOC, bem como uma percepção do desempenho e dos riscos que esta enfrenta.

Palavras-chave: Cibersegurança, Situational Awareness, Dashboard, Security Operations Center, Risco e Visualização de Dados

Contents

List of Figures

List of Tables

Acronyms

APT Advanced Persistent Threat i

CIF Collective Intelligence Framework i
CRITs Collaborative Reasearch Into Threats i
CSA Cyber Situational Awareness i
CSIRT Computer Security Incident Response Team i
CTI Cyber Threat Intelligence i
CVE Common Vulnerabilities and Exposures i

DoS Denial of Service i

EDR Endpoint Detection and Response i

FAIR Factor Analysis Information Risk i

IDS Intrusion Detection System i
IoC Indicator of Compromise i
IoT Internet of Things i
IPS Intrusion Prevention System i
IT Information Technology i

KPI Key Performance Indicator i
KRI Key Risk Indicator i

MISP Malware Information Sharing Platform i
MSSP Managed Security Service Provider i

NIST National Institute of Standards and Technology i

OSINT Open Source Intelligence i

SA Situational Awareness i
SEM Security Event Management i
SIEM Security Information Event Management i
SIM Security Information Management i
SME Small to Midsize Enterprise i
SOAR Security Orchestration, Automation, and Response i
SOC Security Operations Center i
SOCaaS Soc-as-a-Service i
SP Special Publication i
STIX Structured Threat Information Expression i

TAXII Trusted Automated Exchange of Intelligence Information i
TI Threat Intelligence i
TIP Threat Intelligence Platform i
TTD Time to Detect i
TTPs Tactics, Techniques and Procedures i
TTR Time to Recovery i

UEBA User and Entity Behavioural Analytics i

1

Introduction

The following chapter's structure is divided into seven different sections. The first section(1.1) describes the context behind the development of this thesis, followed by the second section(1.2) showcasing the main motivations behind the dissertation. The third section(1.3) covers the problem statement and the fourth section(1.4) describes this thesis' main objectives. In the fifth section(1.5), there is a brief overview of the technologies and methodology employed. Finally, the sixth section(1.6) covers a list of expected contributions culminating in the final section(1.7) that showcases the overall structure of the document.

1.1 Context

This document showcases a dissertation which was conducted in an academic and entrepreneurial context in collaboration with the multinational enterprise EY. The labor conducted during the development phase of this thesis took place in two different phases, each working with a different client inserted in the financial sector. In the same way, the original purpose of the integration of the student in both projects was also different.

In the first client, a Portuguese financial institution, the original goal of developing the solution was to assist the Security Operations Center (SOC) staff in their daily incident response activities by filling a missing Threat Intelligence component. In turn, the work conducted in the second client, an international financial organization, aimed to develop a solution which would provide the organization with a centralized view of the SOC's security functions by letting the viewers monitor a set of strategically picked performance and risk metrics.

During the development of the first system, the student joined the SOC's Computer Security Incident Response Team (CSIRT) with the objective of gaining insight into how a typical SOC operates. Ideally, this would evolve into the development of a visual interface tasked with aggregating security intelligence from various sources, in order to provide a holistic view of the cybersecurity risk landscape. Unfortunately, due to constraints with the client, the development of the solution was halted during the finalization of the

dashboard mockups. Nonetheless, EY proposed a similar solution to another client in their portfolio, which aligned with the profile of the previous one. This proposal ended up being accepted and the project began its development. Contrary to the first system, the second system would be composed by three dashboards, each aimed towards a different audience: the SOC Staff, the Chief of Information Security Officer (CISO) and the Top Management. Furthermore, the focus of the metrics presented in the second system would be more directed towards an internal overview of the SOC's security functions, leaving behind the main Threat Intelligence component that was, previously, heavily focused on.

Ultimately, the development of the solution aims to offer a Situational Awareness archetype to the client through the means of a Dashboard. This feature not only provides a continuous risk assessment analysis, to better measure the level of risk the organization is exposed to at a daily basis, but also allows access to the variables behind the calculation of this metric.

To sum up, two different systems ended up being developed (referred in the rest of the document as Systems A and B). On the one hand, System A is associated with the first implementation which never left its testing phase, whereas System B refers to a full fledged architectural model which completed its testing phase and was finalized in its production stage. The main system this thesis is going to introduce and describe is System B, due to the fact it was implemented in a production environment contrary to System A. Hence, the remaining sections of this chapter will refer to "the solution"as System B.

1.2 Motivation

One of the main reasons behind SOC inefficiency rests upon the following principle: a lack of visibility between the SOC analysts and the data gathered by security tools. Analysts examine hundreds of security alerts every day in order to, quickly and swiftly, respond to incoming threats. Only decisions that are backed by data can be trusted to bring the desired results, but in order to gain valuable insights from their data, enterprises must first understand it, and that's very difficult if one is not able to visualize the data in a way that makes it easy to understand. Therefore, effective data visualization techniques are key to continuously understand the endless flow of intelligence gathered, thus minimizing any possible noise that ends up hindering an analyst's job. Additionally, by leveraging Threat Intelligence, analysts can strengthen their security posture by taking preventive action against incoming security threats.

The quality and quantity of cyber attacks has been increasing over the years as threat actors, not only increase in number, but also employ increasingly more sophisticated Tactics, Techniques and Procedures (TTP). Attackers are motivated by financial gain, gaining confidential data or to disrupt business operations, which makes any enterprise susceptible to this kind of threat. Hence, on top of a visual representation, implementing a proper cyber risk management strategy helps to better categorize and identify the threats

to an organization, in order to help in the prioritization of security threats and mitigation of possible entry points that threat actors might leverage.

Finally, as it was previously stated the aim of this thesis aligns with the development of a potential solution to tackle the key points listed above. In fact, through the development of a Situational Awareness Dashboard displaying several relevant metrics as well as a dynamically computed risk assessment index, it is possible to provide a perception of how exposed and susceptible an organization is to risk.

1.3 Problem Statement

Taking into account both the context and the motivation behind this dissertation, the problems that are addressed by the solution can be summarised below:

- How can we provide a level of visibility into a SOC's collective intelligence (information generated by security tools) so as to not overwhelm analysts with data?
- How can we contribute to help analysts pinpoint security risks?
- How can we provide a perception of the exposure and associated risk of a designated institution to its executive branch, which is not familiarized with security terminologies?
- How can we accurately select the most appropriate metrics to track in a cybersecurity context?
- How can we select the most appropriate visualization models to illustrate each metric in an intuitive fashion?

1.4 Objectives

In order to provide a solution that can address the problems discussed above, a list of key objectives was defined. Therefore, this dissertation aims to:

- Implement a Situational Awareness Dashboard that provides a higher degree of visibility into data to help analysts grasp the real-time cybersecurity picture.
- Produce a Risk Assessment model that measures risk in a daily basis.
- Offer a drill-down perspective of the variables that are responsible for the computation of the Risk Score.
- Give access to the historic evolution of the values affecting the Risk Score, in order to gradually evaluate the performance of the SOC over time.

- Aid analysts in the process of prioritization, through the association of an intuitive risk categorization mechanism and a quantifiable risk value to every metric that measures the SOC's performance.

- Help institutions understand in what ways can their security posture be lacking, which in turn may lead to a finer allocation of resources and budget investment.

- Level out the communication with a client, the Chief Information Security Officer (CISO) or even the Executive Branch as the dashboards can be integrated in monthly progression reports.

1.5 Technologies and Methodology

During the investigation phase of this thesis, several technologies and methodologies were taken into consideration in order to select the most adequate options when moving forward into the development phase. This section aims to address this subject by briefly introducing how the proposed solution interacts with the different components that constitute a typical SOC architecture.

A typical SOC architecture is composed by 3 core components, those being its: People, Technologies and Processes. Further information regarding each component can be consulted in the following chapter.

On the topic of the first component, the proposed solution can be visualized and analyzed by the SOC staff, as well as other important members of the organization that want to get a grasp of the current cyber situational picture, such as the Chief of Information Security Operations (CISO) and the Top Management Board. In fact, System B took this one step ahead and was ended up building a different dashboard for each different user.

The list of bullet points below highlights how each individual can benefit from the implementation of a Situational Awareness Dashboard in a cybersecurity context:

- **SOC Staff** - A typical SOC's Staff is composed by managers, security analysts, and engineers that cooperate in order to quickly address all kinds of cybersecurity issues. Therefore, through the support of a visual representation of data with carefully selected metrics that quantify risk, a Cybersecurity Situational Awareness Dashboard can help analysts prioritize the main security threats and incidents to respond to.

- **CISO** - The CISO is the individual who is responsible for managing information security risk, as well as being the spokesperson between the SOC Team and the Top Management Board. As such, through the power of data visualization, the communication between both entities can be leveled out. Coincidentally, this means that management obtains an overview of the cybersecurity state of the organization, which leads to better decision making and a finer allocation of resources and budget investment.

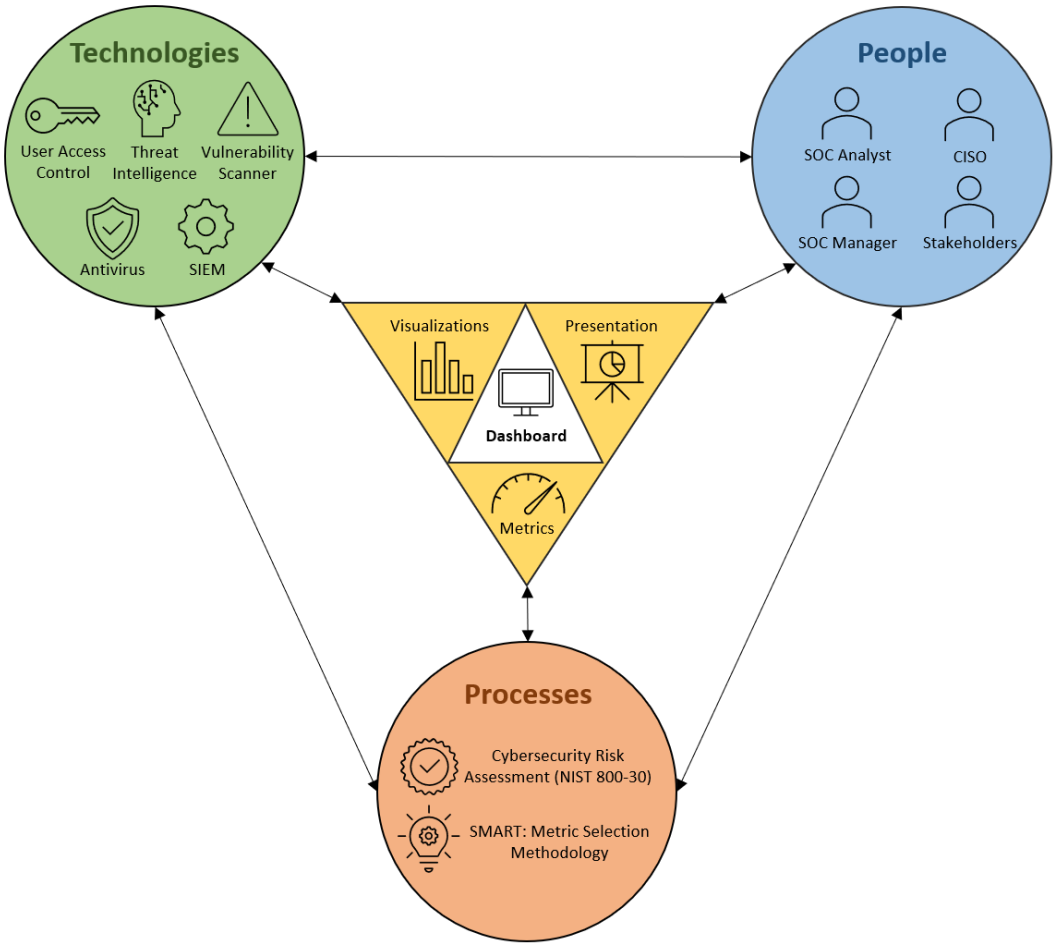


Figure 1.1: Interaction between a Situational Awareness Dashboard and a Security Operations Center.

- **Top Management and Stakeholders** - Stakeholders, who can take the form of investors, are parties with an interest towards a designated organization. As such, clear communication is paramount towards building a trustworthy relationship with such individuals. This goal can be reached through the integration of a Situational Awareness Dashboard, which can be used to smooth out the communication process.

Regarding the second topic, technologies are the core components that compromise the architecture of the solution. The decision making process that took place prior to the development phase of the solution, where several architectural components were analyzed, is entailed in Chapter 3. Nonetheless, they are introduced below and can be divided into three layers: Visualization, Storage and Data Ingestion.

- **Visualization Layer** - Two main data visualization tools were taken into consideration and experimented upon through the development phase: Kibana and Power BI. Ultimately, Power BI was the tool that was chosen to produce the final dashboard.

- **Storage Layer** - Two NoSQL databases were taken into consideration: MongoDB and Elasticsearch. Ultimately, MongoDB was the database that was chosen to be incorporated in the architecture of the solution.

- **Data Ingestion Layer** - Two data ingestion and parsing methods were considered: Logstash and a set of scripts written in Python. Ultimately, the Python Scripts option was the one that was implemented and incorporated in the final solution, due to its intrinsic versatility.

Finally, regarding the methodologies that were taken into consideration, this topic can be divided into: Risk Assessment and Effective Metric Selection.

- **Risk Assessment** - Two different approaches for assessing risk were investigated: the FAIR Model and a methodology proposed by the National Institute of Standards and Technology (NIST) on a guide for conducting Risk Assessments. Ultimately, the latter was the preferred option, as the Factor Analysis Information Risk (FAIR) Model is a methodology which is strongly reliant on a quantitative approach to measuring risk. Quantitative approaches are better suited for enterprises with a higher level of maturity since they have well defined risk assessment frameworks. Both organizations did not achieve this level of maturity.

- **Metrics** - The SMART methodology helps to define appropriate Key Performance Indicators (KPI) and Key Risk Indicators (KRI). According to this practice, appropriate metrics have to be Simple, Measurable, Actionable, Relevant and Time Based. Therefore, this was the standard that was followed for the verification of both system's metrics.

1.6 Contributions

Previously, it has been showcased a list of the key problems being tackled and a set of objectives that the proposed solution aims to fulfill. Lastly, this section summarizes the core contributions that would optimally be delivered by the developed work.

- A set of dashboards targeting different audiences (e.g. Top Management, CISO, Operational) which provide a visual interface aggregating a set of intuitive visualizations with relevant information that point the viewer's attention to the main security risks present in the organization.

- A risk assessment model that is based on the calculation of risk scores through the usage of a set of performance and risk metrics that the viewer (e.g SOC analysts) can consult to understand how the SOC is evolving over time, thus aiming to promote a cyber situational awareness archetype.

- A fully automated and self-sustained system architecture for the proposed solution, tasked with injecting information from different data sources. This information is later parsed, transformed and stored in the database. Finally, the database is connected to a visualization software that renders the dashboard solution with all its visualization components.
- A user manual which explains, in detail, the dashboard's capabilities, how to navigate it and how to perform modifications to the risk formulas. Hence, this manual could help new users understand and learn how to manipulate the information being displayed, allowing for the possibility of the solution's expansion.
- An alert mechanism that triggers notifications/emails that notify, for instance, the SOC's staff when designated metrics reach a certain threshold.

1.7 Document Structure

This document is divided into seven different chapters:

- **Introduction** - The first chapter presents an overview of the contents of the dissertation through its context, motivation, problem statement, objectives, technologies, methodology and contributions.
- **State of the Art** - The second chapter encompasses the State of the Art associated with the investigation that drove this dissertation to fruition. Therefore, it covers essential topics such as common Cybersecurity Threats, Threat Intelligence, Situational Awareness, Risk Assessment, the historical evolution of the SOC, defines metrics and, finally, provides some examples of Situational Awareness Dashboards used in different contexts.
- **System Architecture** - The third chapter showcases and describes the architecture that was used to develop both systems A and B. It explores the different technologies that were researched, justifying the reason why each architectural model came to be as it is.
- **System A/B: Implementation** - The fourth and fifth chapters provide a focused view of the components that are part of each system architecture. In fact, each chapter explains every step of the implementation from the gathering, parsing and mapping of data to the dashboard construction, detailing every metric and every visualization.
- **Product Evaluation** - After describing both implementations, the sixth chapter aims to evaluate System B, through a series of test cases and scenarios. These scenarios verify if the developed solution achieves the set of contributions it was tasked to fulfill.

- **Conclusion and Future Work** - The seventh and final chapter summarizes the most important conclusions taken from this thesis and tackles some suggestions that could be incorporated in a future implementation, in order to refine the product that was developed.

2

State of the Art

The following chapter is divided into 3 main sections. It was decided that all the core concepts should be defined first so that afterwards we could tackle the environment where the thesis was developed in and finally, what was developed and with what purpose. Hence, the following chapter's main sections are: Current Cybersecurity Threat Landscape, Security Operations Center and Situational Awareness Dashboard.

The first section (2.1) starts by explaining to the reader the state of the cybersecurity threat landscape, as well as what kind of threats exist and who takes advantage of them to conduct attacks on organizations across the world (2.1.1). This topic is followed by 3 core concepts that exist to help organizations on their fight against cybercrime, those being Threat Intelligence (2.1.2), how it can be leveraged to achieve Situational Awareness (2.1.3) and contribute to cyber risk management (2.1.4).

The second section (2.2) describes the environment where the development of the solution was achieved. It consists on an overview of the history of the SOC and how it evolved across different generations (2.2.1). Afterwards, the 3 main SOC components are addressed (Staff: 2.2.2, Technologies: 2.2.3, Processes: 2.2.4), as well as some challenges and limitations of a typical SOC architecture (2.2.5). Finally, the section is closed with some alternatives to an in-house SOC (2.2.6).

The third section (2.3) gathers the principles detailed in the previous sections into the concept of a Situational Awareness Dashboard inside the SOC of a financial institution, detailing ts importance, benefits and architectural necessities.

Finally, the fourth section (2.4) is used to sum up this chapter with some key conclusions.

2.1 Current Cybersecurity Threat Landscape

Cybercrime is criminal activity committed by individuals or organizations through computers, computer networks or network devices. It is an attractive option for criminals due to its low chance of getting caught [71]. Throughout recent years cybercrime has seen an unprecedented increase in numbers as cybercriminals continue to disrupt businesses and

take advantage of the inherent financial profit earned through this practice. In fact, this constant surge in cybercrime has been especially predominant since March 2020 which marked the beginning of a certain worldwide event.

Nowadays we live in a time period subjugated by the COVID-19 pandemic and as such, this worldwide phenomenon has served as a catalyst for cybercrime to grow at an exponential rate. In fact, according to a published article by "IMC Grupo"[12], since the pandemic began, the FBI reported a 300 percentage increase in reported cybercrimes. Accordingly, COVID-19 gave threat actors new opportunities to strike due to the ramifications caused by this pandemic to businesses. This phenomenon caused several changes to not only people's social but also to their business life. Corporations adapted to this new reality by shifting their business operations from their offices to a decentralized, work-from-home model. Nonetheless, this solution came with its shortcomings as adversaries know that employees are working remotely. This new way of working introduces cybersecurity risks since employees sometimes rely on their home network and personal devices to complete tasks. Additionally, offices provides firewalls among other technologies which protect employees while they are working. Coupling the fact that users lack the degree of cybersecurity controls at home that they have at corporate offices with the new array of endpoint devices used, emerges a whole new realm of vulnerabilities and attack vectors for threat actors to leverage. Thus, ever since the COVID-19 pandemic started, the numbers of scams and malware attacks have significantly risen, with phishing being reported to have increased by 600 percent in March 2020 [54].

This new wave of cyber attacks put all organizations on high alert, mainly the financial and healthcare sectors, as 27 percent of COVID-19 cyberattacks target banks or healthcare organizations [40]. Finally, COVID-19 is credited for a 238 percent rise in cyber attacks on banks [40] and therefore, more than ever have organizations across the globe felt this unprecedented need to protect their clients, their employees and their assets.

In order to help the reader get a better understanding of the aim of this thesis some key concepts need to be addressed. The list of topics below will serve as an overview of some core concepts that will be tackled in the following sections, aiming to answer some questions about:

- Who and what should organizations be concerned about in their missions towards fighting cybercrime?

- What is Situational Awareness and why should organizations care about aiming to achieve a Situational Aware archetype?

- What is Threat Intelligence and how can enterprises leverage this concept in their favor?

2.1.1 Cybersecurity Threats

The topic of cyber attacks is a multi-layered one as it involves understanding different factors, such as, who conducts these attacks and what motivation drives each different type of threat actor, but also what kind of threats exist and whats steps do threats actors take into exploiting and conducting a cyber attack.

First and foremost, threat actors or cyber threat actors are "states, groups, or individuals who, with malicious intent, aim to take advantage of vulnerabilities to gain unauthorized access to information systems in order to access or otherwise affect victims' data, devices, systems, and networks"[14]. Acquiring a perception into threat actors, their motivations, tactics, techniques and procedures is a key step in the cybersecurity process, in order to help organizations develop a more focused cybersecurity plan. Additionally, by placing oneself in the position of the attacker, organizations can better outmaneuver attackers successfully. According to an article published by the Rand Corporation[1], threat actors can be segregated into five major different types each with their own set of motivations and techniques:

- **Cyberterrorists** - Threat actors that conduct acts of terrorism by performing attacks via the use of technology in cyberspace. Aiming to influence an audience through fear and violence they try to force political changes, motivated by their usually extreme ideologies.

- **Hacktivists** - Another example of a type of actor which is motivated by their ideologies, they aim to bring awareness into a cause by targeting and exposing corporations, agencies or any entity deemed as 'evil' by the hacktivist group. For instance, leaking sensitive information about the target or disclosing existing vulnerabilities to the public are examples of attacks performed by this type of actor.

- **State-Sponsored actors** - This type of actors are contracted, assisted and funded by nation-states in order to further their political, commercial, or military agendas. As a manner of fact, by being backed up by the nation's government they are granted more resources than the average actor posing as a dangerous threat for their targets. Usually, victims of this threat actor can be nations or entities within the technological and financial sectors.

- **Cybercriminals** - Cybercriminals are all about monetary gain, monetizing any type of stolen data through underground black markets. Phishing, spear phishing and leveraging known vulnerabilities are just some of the techniques employed by this actor in order to extract any type of confidential information they can later sell. Thus, entities that hold a significant of personal data about their customers are prime targets for this type of criminal activity, such as banks, retail companies and healthcare institutions.

- **Insider Threats** - Threat actors that conduct attacks from within the targeted organization. Potential insider threats are any individuals with knowledge of the organization's confidential data, IT, or network resources. They pose as extremely dangerous threats as organizations can't rely on traditional security measures since these actors are already inside the organization.

The categorization depicted above shows that threat actors differ in terms of resources, motivations and skills, characteristics which help to predict the identity of their future victims, how they will carry out their attack and to pinpoint the type of asset or data they are after. Thus, entities can take a proactive posture and re-enforce their defenses, limiting an actor's options to attack.

On the topic of attacks, there are a multitude of different cyber threat categories that enterprises should acknowledge and take measures against. In fact, the European Union Agency of Cybersecurity (ENISA) identified the top 15 cyber threats of 2020 (Figure 2.1). These constitute some examples of the most predominant attacks targeting enterprises all over the world.

Figure 2.1: The Top 15 Cyber Threats of 2020 (ENISA) [20]

For years, enterprises have been successful at keeping some of these threats at arm's length. Despite this, there will never be sure fire way of bulletproofing an organization's defenses. Therefore, understanding the attacker's behaviours becomes essential. In fact, the Pyramid of Pain (Figure 2.2) highlights this topic perfectly.

The idea behind the pyramid is that the higher we get, the harder it is for the attacker to overcome or replace the method the victim has gotten rid of. Whereas simply blocking file hashes, IP addresses or domains is easy to outmaneuver, taking away an attacker's Tool or TTP seriously hinders the way they conduct their attacks, since the threat event is being treated from its roots. Coincidentally, the process of identifying and inhibiting a TTP is challenging and time consuming and as times moves this is only going to get harder, as attacks are getting even more sophisticated due to threat actors always being one step ahead of their victims.

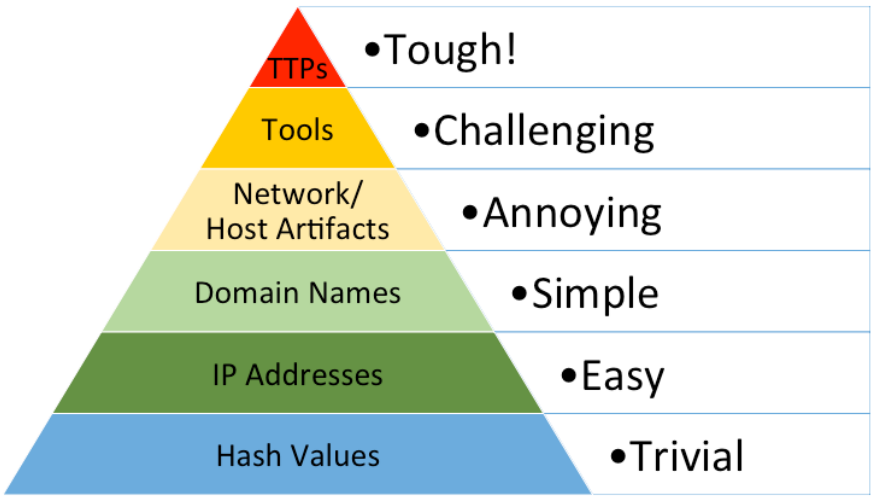


Figure 2.2: The Pyramid of Pain [64]

As the level of sophistication of attacks increases over time threat actors began creating more advanced ways to breach the commonly implemented signature based security measures. Hence, recently a new class of threats emerged, considered one of the most dangerous ones targeting organizations: the Advanced Persistent Threat (APT). APTs are unique in the sense that they are categorized as cyber threats with a particular goal and targeting a specific business or political entity[50]. These targeted multi stage attacks employ sophisticated techniques, leverage unknown security vulnerabilities and are conducted in a long term fashion until the intruder's goal is reached. Hence, due to its high sophistication, the potential financial repercussion of these threats can be huge, posing as an extremely dangerous threat.

In light of the ever evolving threat landscape and the emergence of ATPs, several approaches were developed in order to track and analyze the various characteristics of cyber intrusions, such as the Cyber Kill Chain Model and the MITRE ATT&CK® framework.

The kill chain model was originally employed in a military context and the concept was related to the structure of an attack. The idea was to preemptively stop an attack by breaking one of the "chains"of the kill chain, as this would halt the attack. Hence, this model was adapted to a cybersecurity context and extended into the Cyber Kill Chain Model. According to the model proposed by Lockheed Martin, the process of conducting

a cyber attack can be divided into 7 different phases visible in Figure 2.3: Reconnaissance, Weaponization, Delivery, Exploitation, Installation, Command and Control and Actions on Objectives [58].

Figure 2.3: The Lockheed Martin Cyber Kill Chain Model [58]

Ever since the this model started being implemented, methods for conducting cyber attacks have evolved outstandingly, leaving the cyber kill chain model outdated over time. Likewise, there exist other disadvantages to the kill chain, the main one being that it does not meet specifications for all types of attack vectors. For instance, insider attacks, which represent malicious threats that come from inside the organization, are not taken into consideration by this model [55]. Despite this historical deprecation, the kill chain it still widely used, however another framework has been gaining traction and is favored by researchers as the eventual replacement for the kill chain: the MITRE ATT&CK [15].

The MITRE ATT&CK framework is a knowledge base of adversary TTPs which takes into account real-world observations of millions of attacks gathered from publicly available threat intelligence and incident reports [37]. It expands the Cyber Kill Chain model into 14 categories of Tactics, which represent the steps an attacker will usually go through when orchestrating an attack. Different routes can be taken to achieve the same Tactic,

as such, a set of Techniques are documented for each Tactic. Finally, each Technique or attack, has a Procedure for being executed which is also documented by the framework. The ATT&CK framework is built up of several matrices, one for Enterprises, Mobile and Industrial Control Systems.

Despite being two frameworks that try to frame and document the behaviour behind attacks conducted by threat actors, according to MITRE, ATT&CK and the Cyber Kill Chain are complementary models. Whereas, ATT&CK Tactics are unordered, define adversary behaviour at a low level and do not trace a linear path for attacks to follow, the Cyber Kill Chain uses linear, ordered phases to describe high-level adversary objectives [38].

To sum up, overtime there has been a surge in the quantity and level of sophistication of cyber threats orchestrated by threat actors. This led organizations to shift their reactive posture to a more preventive one, starting to document their adversary's motivations, behaviour and techniques. Through models like The Pyramid of Pain, cybersecurity specialists understood the need to catalog and inhibit the tools and TTPs that cyber criminals used to conduct their attacks. Hence, several models were devised to document the cyber intrusion process, such as the Cyber Kill Chain and the MITRE ATT&CK matrix. In fact, these standards add value to cyber threat intelligence through the contextualization of security incidents and attacks of their stage in the Cyber Kill Chain or the TTPs employed. The next section expands on this topic by defining the concept Threat Intelligence, its life cycle and purpose.

2.1.2 Threat Intelligence

Threat Intelligence(TI) is "the contextualised output of a strategically-driven process of collection and analysis of information pertaining to the identities, goals, motivations, tools and tactics of malicious entities intending to harm or undermine a targeted organisation's operations"[9]. In other words, TI is evidence-based knowledge about existing or imminent threats that can help organizations identify, evaluate and respond to cyber threats. Given the fact that different threat actors might employ the same of tools and TTPs, sharing TI is becoming increasingly important for organizations who want to improve their security posture. In fact, according to NIST Special Publication 800-150 on a Guide to Cyber Threat Information Sharing [42], there are several benefits to sharing TI among trusted partners, such as, shared Situational Awareness and an improved Security Posture. Whereas the former helps to enhance defensive capabilities by trading valuable adversary intelligence among a trusted group therefore helping entities map the current cyber situational picture and contributing to the overall pool of intelligence regarding cyber threat actors and their TTPs, the latter is a consequence of better understanding the threat environment and the behaviour, motivations and TTPs of cybercriminals.

In order to better understand the concept of TI it is important to grasp its life cycle. The Cyber Threat Intelligence (CTI) cycle is an iterative five-step process that illustrates

the transition of raw data into fully fledged intelligence. As illustrated in Figure 2.4, the cycle can be divided into 5 different and fundamental stages: Planning, Collection, Processing, Analysis and Dissemination [27].

- **Planning** - This phase defines the TI from the data collection phase to the delivery of intelligence. Additionally, an intelligence team is formed and their roles and responsibilities are assigned.
- **Collection** - This phases has a focus on data collection. Data can be collected in different ways: Human Intelligence (HUMINT), Covert Human Intelligence Sources (CHIS), Open Source Intelligence (OSINT), Signals Intelligence (SIGINT) and Technical Intelligence (TECHINT). Finally, once the collection phase is finalized data is sent for processing.
- **Processing** - The data that was collected in the previous phase is raw data in different formats which makes it difficult to aggregate and analyze it in the next phase. Therefore, data is converted into a usable format that can be directly used in the data analysis phase.
- **Analysis** - This phase involves extracting the knowledge of the normalized data attained from the previous phase. Data analysis techniques such as machine-based techniques and statistical methods are used to transform the raw data into refined readable intel.
- **Dissemination** - Finally, in the final phase the resulting data from the previous phase is captured in order to reach the audience through, for example, machine-readable data feeds or Application Programming Interfaces (APIs).

Now that it has been established how raw data is turned into practical and actionable threat intelligence, let us overview some major types of threat information, those being Indicators of Compromise (IoCs) and TTPs[42].

IOCs are "artifacts observed on a network or in an operating system that can be utilized to indicate a computer intrusion and detect cyber-attacks in an early stage"[32]. Hence, enterprises can leverage this intelligence in order to prevent incoming threats or even detect threats that are actively present inside of the organization's network. For instance, common examples of indicators are IP addresses, URLs, domains, email addresses and file hashes.

As we have seen in the previous section, a TTP stands for Tactics, Techniques and Procedures applied by threat actors. In the context of TI and how TTPs can be leveraged in an organization's favor, this intelligence lets entities understand how adversaries orchestrate and execute attacks. As a manner of fact, through TTPs cybersecurity personnel can understand a threat actor's tendency to deliver their attacks and exploit their victims. This lets analysts prepare for upcoming attacks by, for instance, patching existing vulnerabilities a designated adversary might tend to leverage, thus closing a possible attack

Figure 2.4: The Cyber Threat Intelligence Cycle [27]

vector. In fact, when properly employed, TI has the potential to offer protection against APTs, which constitute extremely dangerous threats for any kind of organization.

Among the different ways that data can be collected, (highlighted in the Collection phase of the CTI cycle) the most relevant source in the context of this thesis and of the environment it will be applied on is OSINT. OSINT is an intelligence model which aims to find, select and retrieve data from publicly available sources. To clarify, examples of OSINT sources include, but are not limited to: no-cost public threat data feeds and commercial providers of fee-based TI services that aggregate and enhance existing public threat data feeds or provide TI based on their own OSINT collection [9]. "Feed de Segurança Informática"is an example of a no cost public TI feed which compiles phishing and malware campaigns targeting Portuguese citizens. On the other hand, IBM X-Force Exchange and OTX Alienvault are examples of the latter, organizations that offer services that let interested parties retrieve their published TI.

The main challenge faced by TI is the fact that no isolated entity has access to a quantity and quality of information that introduces an accurate situational awareness picture. Moving forward, it is paramount for institutions to achieve optimal situational awareness by sharing, gathering and analyzing TI among trusted partners and communities. Thus, institutions need TIPs that can combine internal and external threat data from numerous sources for correlation. To illustrate, a few examples of several TIPs are the Malware Information Sharing Platform (MISP), the Collective Intelligence Framework (CIF) and Collaborative Research Into Threats (CRITs). Taking MISP as an example, MISP is an Open Source TIP meant for "sharing, storing and correlating Indicators of Compromise of targeted attacks, threat intelligence, financial fraud information, vulnerability information or even counter-terrorism information"[35]. Additionally, MISP is highly praised

amongst the community because of its features. First and foremost, in regards to its ability to import and export data in different formats, MISP can handle numerous formats such as but not limited to JSON, txt, PDF and STIX. Secondly, it contains a flexible API which enables the easy integration of MISP in other systems. Finally, it contains data exchange mechanisms as it supports popular standards such as STIX and TAXII.

Expanding on the final step of the CTI Cycle, over the years a few conventions were established in order to smooth out the process of sharing and communicating threat information. Standards like CVE, STIX and TAXII emerged to satisfy this gap in standardization.

Firstly, Common Vulnerabilities and Exposures (CVE) is a dictionary, developed by MITRE in 1999, created to identify, define, and catalog publicly disclosed cybersecurity vulnerabilities. It is a free to use standard where collaborators of the CVE program can append security vulnerabilities as they are detected. Hence, cybersecurity professionals and specialists around the world can communicate in a universal fashion helping each other to close existing attack vectors that threat actors might be trying to exploit to achieve their criminal agendas [39].

Lasty, Structured Threat Information Expression (STIX) and Trusted Automated Exchange of Indicator Information (TAXII) were standards also developed by MITRE each with their own objective and contributions to the dissemination of Threat Intelligence. As a manner of fact, through STIX and TAXII sharing TI became an accessible possibility for every institution.

Structured Threat Information Expression (STIX) is a structured language and format used to exchange CTI. Through its machine readability it enables organizations to exchange CTI in an automated fashion, promoting faster responses to threats. This standard represents data as objects, existing 18 different categories, such as, indicators, malware, vulnerabilities, threat actors, tools and campaigns. Figure 2.5 represents a STIX object in JSON describing a "Campaign", in other words, a set of malicious activities or attacks.

```
{
    "type": "campaign",
    "id": "campaign--8e2e2d2b-17d4-4cbf-938f-98ee46b3cd3f",
    "spec_version": "2.1",
    "created": "2016-04-06T20:03:00.000Z",
    "modified": "2016-04-06T20:03:23.000Z",
    "name": "Green Group Attacks Against Finance",
    "description": "Campaign by Green Group against targets in the financial services sector."
}
```

Figure 2.5: JSON-based example of a STIX 2.1 Campaign object [45]

Additionally, these objects can have relationships between them making it possible to build graphs around the data. This way organizations can better understand their adversary's TTPs. For instance, Figure 2.6 showcases four STIX objects: a campaign, a threat actor, an indicator and a vulnerability. Despite being interesting intelligence, this

information is not actionable if we analyze it separately. However, once the objects start forming relationships amongst each other, some key conclusions might be extrapolated. Thus, after forming the links between objects we know that a designated indicator is being used by a threat actor to perform a campaign that targets a specific vulnerability. This way organizations can take preemptive action and close the vulnerability before they are targeted by this campaign. This is one of many examples that showcase the power of TI and standards like STIX.

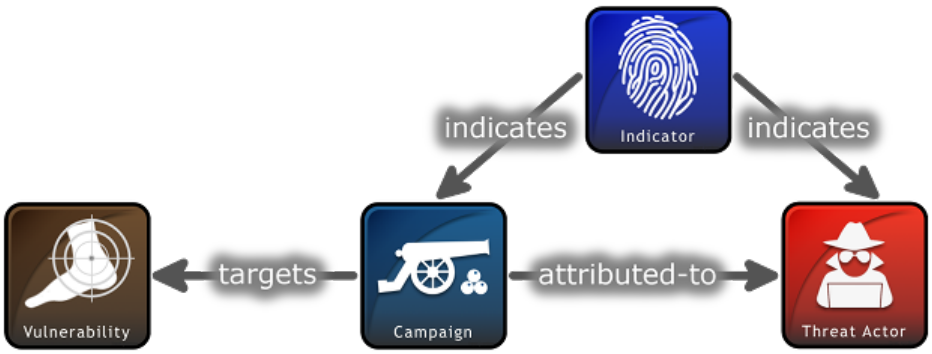


Figure 2.6: STIX 2 Relationship Example [45]

Finally, Trusted Automated Exchange of Intelligence Information (TAXII) is a transport layer protocol for sharing CTI over HTTPS specially designed for the STIX format. Communication is performed with the support of a RESTful API defined by TAXII with two types of exchange services: Collections and Channels. Whereas the Collection represents a connection to the TAXII Server with the data for the TAXII Client to make requests, the Channel is maintained by a TAXII Server and represents a channel wehere consumers (TAXII Clients) subscribe to the producer (TAXII Server) which pushes new data directly to the consumers following the publish-subscriber model [46].

To sum up, successfully incorporating Threat Intelligence in an organization's operations is becoming increasingly important in order to stay up to date to the latest cybersecurity attack trends. Monitoring adversary behaviour and TTPs through sharing CTI amongst trusted partners goes a long way towards strengthening an entity's security posture. Additionally, it can be inferred that the best CTI policies incorporate some sort of automation in their data collection, processing and sharing through the use of TIPs with STIX and TAXII compatibility. This in turn improves the current cyber situational awareness landscape as organizations are more aware of their cybersecurity posture. The following section will further explain Situational Awareness, as well as its importance in the context of cybersecurity.

2.1.3 Situational Awareness

Situational Awareness (SA) is defined as "the perception of the elements in the environment within a amount of time and space, the comprehension of their meaning and the projection of their status in the future"[19]. SA depicts the three fundamental steps towards achieving good decision making, those being the **perception** of critical factors in an environment, the **comprehension** of those same factors and what do they translate into and lastly, the **projection** of the state of the system in the future.

Accordingly, the Committee on National Security Systems extended the SA definition to describe Cyber Situational Awareness (CSA) as "within a volume of time and space, the perception of an enterprise's security posture and its threat environment; the comprehension/meaning of both taken together (risk); and the projection of their status into the near future"[11]. Therefore, in a SOC context achieving Cyber Situational Awareness (CSA) is divided into 3 main components: information **gathering**, **processing** this information and **projecting** this intelligence in a visual representation [78]. On this note, both terms SA and CSA will be used interchangeably for the rest of the document. All in all, CSA is a defense strategy that through the use of early warning data, like Threat Intelligence, factors to proactively assess and mitigate cybersecurity threats.

Taking into account the previous mentioned 3 main components, it's clear that achieving CSA requires an ongoing investment in data collection, management and analysis of the organization's computer systems, networks and users. Hence, risk inducing situations can be recognized and possibly predicted before they occur as entities have fundamental awareness of what's occurring across any affected domain.

Additionally, the CSA model can be compared to the OODA loop in the context of decision making. The OODA loop, which stands for "Observe, Orient, Decide, Act"is a four step process that serves as a methodology for achieving Situational Awareness. In fact, successfully implementing the OODA loop achieves the 3 main components of SA depicted above since the 4 steps of the loop align with those of CSA. The first phase, the observation one, is used to identify any threats and understand the internal and external environments through gathering data partaking to the organization, its competitors and the market. Next, the orientation phase is used to reflect on the discoveries found in the previous phase in order to know what should be done next. Afterwards, the decision phase is used to discuss response plans where all the possible outcomes are compared. Finally, on the last phase the response plan is acted upon and the cycle repeats itself, going back to the observation phase to improve our model and find new threats in the environment [28].

To sum up, organizations aiming to achieve a cyber situational awareness archtype benefit from:

- **Improved Agility and Security Posture** - Cybersecurity professionals who directly implement and apply situational awareness methodologies are better able to assess current vulnerabilities and act in the presence of security liabilities. This agility

enables companies to take preemptive action through the mitigation of potential attacks and elimination of vulnerabilities before threat actors have a chance to strike, negatively affecting the affected entity.

- **Smoother Communication** - All levels of the organizational hierarchy need to actively understand their cyber environment. Therefore, clear communication between the chain of command is paramount. As the level of awareness rises, management benefits from easier decision making and budget allocation, as they can now better understand the impact of a situation on the organization's ability to execute its operations.

Ultimately, CSA offers a holistic view of the situational environment surrounding an organization, as well as the ability to comprehend its threat environment in real time. Additionally, CSA and methodologies such as the OODA loop not only improve organization's agility in regards to their threat response mechanisms, but also serves to keep organizations safe and capable of improving their decision making across the board. On a final note, establishing a SA archtype involves comprehending the relationship between an enterprise's security posture and its threat environment taken together, which translates to the concept of risk. Therefore, the next section will further explain this concept as well as how to deal with risk in a corporate environment.

2.1.4 Risk Assessment

All types of organizations, from small enterprises to the most widely known corporate behemoths face risks. According to [43], risk is "a measure of the extent to which an entity is threatened by a potential circumstance or event, and typically a function of: (i) the adverse impacts that would arise if the circumstance or event occurs; and (ii) the likelihood of occurrence". On a macro scope, Enterprise Risk is defined as "the effect of uncertainty on objectives"[44] and as such, it should be managed through Enterprise Risk Management (ERM) policies. ERM is considered "an effective agency-wide approach to addressing the full spectrum of the organization's significant risks"[44]. Coincidentally, one of the main topics this thesis aims to tackle is Cybersecurity Risk, which is one significant portion of the spectrum of an enterprise's core risks.

As previous evidence has shown, enterprises all over the world have experienced a surge in the quality and frequency of attacks. In order for organizations to meet the requirements for addressing Cybersecurity Risk, they must ensure the confidentiality, integrity, and availability of information or information systems. Failing to fulfill this premise places an entity at risk, resulting in potential adverse impacts to its organizational operations. Ultimately, the concept of cybersecurity risk can be defined as "probability of exposure or loss resulting from a cyber attack or data breach on your organization"[67]. As such, implementing a proper Cybersecurity Risk Management (CSRM)

methodology as part of their ERM, should be one of the cornerstones for any successful business which aims to[62]:

- **Mitigate cyber risks and prevent attacks** - CSRM allows for an easier identification and mitigation of threats through the establishment of proper risk treatment plans.
- **Reduce costs and protect revenue** - Attackers are mainly motivated by the financial gains associated with cyber attacks. Therefore, by implementing a CSRM strategy organizations can mitigate the loss in revenue related to fines associated to the non-compliance of certain regulations.
- **Increase business reputation** - Implementing CSRM strategies provides a competitive edge, since entities show their customers that they prioritise their data leading to a more trustworthy relationship.

Despite the establishment of different risk management methodologies (a subject the will be explored in the next chapter), there are some core components that are ubiquitous to every risk management process.

Risk registers and Risk Assessment Reports (RAR) are complementary and critical documents that serve to document the Risk Management Process. Whereas the RAR "contains the results of performing a risk assessment or the formal output from the process of assessing risk"[43], the risk register helps to convey and coordinate cybersecurity risk activities by documenting the complete risk management life cycle. According to the NISTIR 8286 in "Integrating Cybersecurity and Enterprise Risk Management (ERM)"[44], a typical risk management life cycle, as can be viewed in Figure 2.7, is composed by 6 different steps: **identifying the context**, **identifying the risks**, **analyzing the risks**, **prioritizing risks**, **planning and executing risk response strategies** and **monitoring, evaluating and adjusting**. In fact, this final step is particularly important as it connects to the first step, in order to continuously re-evaluate and improve the Risk Management Plan.

- **Identify the Context** - The first step in the risk management cycle is understanding the context, in other words describing the environment in which risk-based decisions can be made. Organizational context can be segregated into two factors, the external context compromised by the stakeholders objectives and expectations about how risk is managed and the internal context which aligns with factors that influence the organization's CSRM.
- **Identify the Risks** - After identifying the context, the next step should focus on identifying a set of risks, in other words, identifying a list of factors that may jeopardize the confidentiality, integrity, and availability of information systems. Identifying an organization's assets, its potential vulnerabilities and associated consequences when compromised, constitute some of the tasks conducted during this stage.

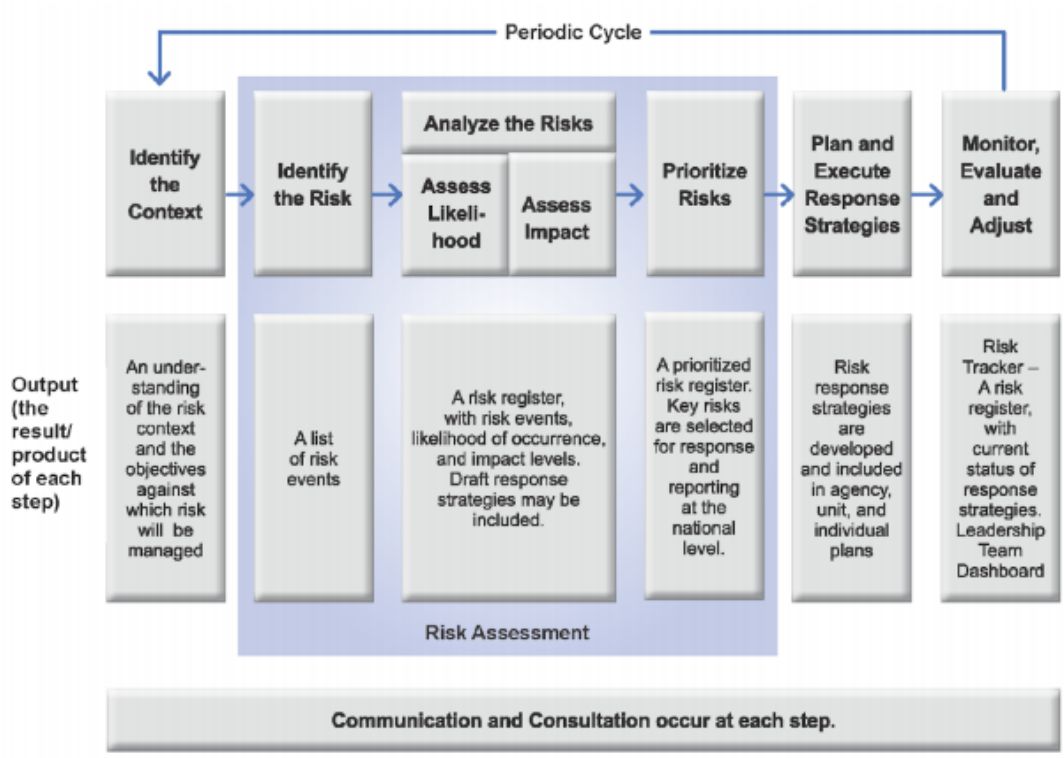


Figure 2.7: Risk Management Life Cycle [44]

- **Analyze the Risks** - After identifying the risks faced by the organization, the next step should focus on estimating a likelihood and potential impact for each risk event. Several approaches can be taken into consideration such as, a qualitative, quantitative or semi-quantitative analysis, each with its own set of advantages and disadvantages.

- **Prioritize Risks** - After estimating the likelihood and adverse impact of each threat event registered on the risk register, this information should be translated into a degree of exposure for each risk. Thus, through building a risk matrix and defining accurate thresholds, enterprises can categorize risk, facilitating the process of prioritization and mitigation of threats.

- **Plan and Execute Response Strategies** - After tracking down and extensively diagnosing all the potential threat events and risk scenarios the next step focuses on determining suitable response plans for each risk. Risk response plans should be introduced carefully as some response might introduce new risks into the environment. Additionally, since responding to every potential threat and risk is inconceivable, risk response plans should be performed in a cost-effective fashion. Thus, four different self-explained actions are available when treating risk: accept, transfer, mitigate or avoid.

- **Monitor, Evaluate and Adjust** - Establishing a proper risk management policy requires a continual monitorization of risk parameters, evaluation of their importance and adjustment of the appropriate risk treatments. Thus, this phase represents a key step that takes part during the risk management life cycle as it contributes to its evolution through ongoing dialogue between staff members and all relevant stakeholders.

Despite following a common layout, different risk management methodologies approach the 6 steps depicted above in different ways. Let us take the NIST Special Publication 800-30 as an example. According to a comparative evaluation made across different methodologies in [21], this framework received modest scores in regards to the context identification and monitorization phases (Steps 1 and 6) of the risk management life cycle. Additionally, it is classified as one of the best methodologies regarding the procedures that one must take into consideration during the Risk Assessment phase (Steps 2, 3 and 4). On the other hand, this same framework scores poorly in regards to its Risk Treatment component as it hardly mentions this topic. Coincidentally, since the solution proposed by this thesis mainly tackles the subjects of risk regarding context, communicating and monitorization, aiming to achieve optimal SA, much of the developed work was based on the NIST SP 800-30 which perfectly aligns with these requirements.

To sum up the contents of the previous sections, in order to reach a Situational Awareness archtype, it is necessary to both actively keep up with the latest cybersecurity attacks, exploits and adversary TTPs through the usage of Threat Intelligence (external component), as well as to monitor the organization's assets, users and network in search for potential security breaches (internal component). By combining both these factors it is possible to monitor the level of cybersecurity risk an entity is exposed to, which is part of the organization's ERM plan. With CSRM being a core component to the organization's ERM plan, the cybersecurity staff that is in charge with the security component is tightly connected with cyber risk monitorization. Hence, in order to successfully operate a Security Operations Center (SOC), which is the unit in charge with all cybersecurity matters, the staff should have a constant grasp of the cybersecurity environment through a visual representation of some intuitive and informative metrics. The sections below aim to introduce and further explore these topics, starting with the core unit that houses and centralizes all cybersecurity matters in an organization: the SOC.

2.2 Security Operations Center

The third Industrial Revolution, also known as the Digital Revolution, set the stage for the Information Age which marked its beginning on the latter half of the 20th century. Throughout this period, as technological innovations like computers, digital computation and the Internet emerged, an economical shift took place towards Information Technology

(IT), which is defined by "the use of computers to store, retrieve, transmit, and manipulate data"[16]. Throughout history, human society has striven to protect its physical possessions from those who mean them harm. Therefore, with the growing digitization of data and assets proportioned by IT, an arising need to protect them from threats in the cyberspace emerged. Consequently, one of several solutions devised to tackle this imminent threat was the development of the SOC, an infrastructure composed by a team of cybersecurity engineers, whose expertise coupled with threat intelligence gathered from multiple security tools, are tasked with identifying, analyzing and reacting to threats with the ultimate goal of protecting the resources of an institution [6, 8].

2.2.1 Historical Context

As with many other technological innovations, the first primitive instance of a SOC was deployed by military and government entities [6]. Similarly, corporations quickly followed their footsteps after realizing the potential of IT, quickly becoming dependent on it. For this reason, there was an unprecedented need to counteract malicious activity which sprouted the development of security mechanisms and tools that contributed for its evolution. According to [6], the evolution of the SOC can be partitioned across 5 different generations.

The first generation took place during the birth of the Internet. In fact, as enterprises had no defense mechanism, threat actors working individually at the time, took advantage of this through their creative thinking and social engineering skills. Therefore, solutions such as firewalls and antivirus constituted the first of many security tools to be created.

The second generation went from 1996 up till 2001 and was marked by an era of malware, worms and viruses who wreaked havoc among corporate and government institutions. To illustrate, the "Happy99"worm targeted Outlook Express and was considered "the first virus to spread rapidly by email"[1]. Consequently, as a means to counteract the rapidly evolving landscape of threat actors, a threat detection stance was taken with the creation of the first Intrusion Detection Systems (IDS). Furthermore, enterprises started offering security monitoring and management services, such as IBM and AT&T, also known as Managed Security Service Providers (MSSP). Equally important, was the development of the first Security Information Event Management (SIEM) system, tasked with achieving real-time analysis of security alerts, which became the core technology used in future SOC iterations.

By the time of the third generation, taking place in the mid-2000s, government and MSSP entities had developed fully-fledged SOCs as the rest of the industry also started adhering to this initiative. Despite the development of security standards, threat actors became more organized, using bots to steal identity and financial records and to perform Denial of Service (DoS) attacks, meaning cybercrime was beginning to take a more financially driven route. Meanwhile, malware attacks were still a reality with the SQL Slammer, which propagated through a buffer overflow vulnerability in Microsoft SQL

server 2000, being an example of a worm with devastating repercussions throughout the Internet [76].

The fourth generation was marked by cyber wars driven by political agendas and Hacktivist organizations performing Advanced Persistent Threat (APT) attacks on major enterprises. This attack employs sophisticated hacking techniques in order to gain access to a system, lurking for long periods of time and stealing confidential data recurrently. As a result, organizations started collaborating between each other through the use of Threat Intelligence (TI), with other Open Source Threat Intelligence Platforms (TIP) such as MISP, starting development in 2012 [36]. Nevertheless, TI is not a silver bullet because if an APT leverages undiscovered attack types, TI cannot provide APT-related intelligence for that specific threat.

Finally, during the current and fifth SOC generation, experts came to the realization that with cyber attacks growing exponentially, a reactive defensive posture was insufficient, as it was widely ineffective against APTs, and measures had to be taken to shift its stance to a more proactive and preventive one. In order to maintain the cybersecurity situational picture, minimize risk and find previously unknown attack vectors and Indicators of Compromise (IoC), organizations are beginning to combine SIEMs with big data analysis and employ constant and automated intelligence gathering and sharing with other organizations. To sum up, these are just a few of many changes that constitute the current and future generation of SOCs which are summarized in Figure 2.8.

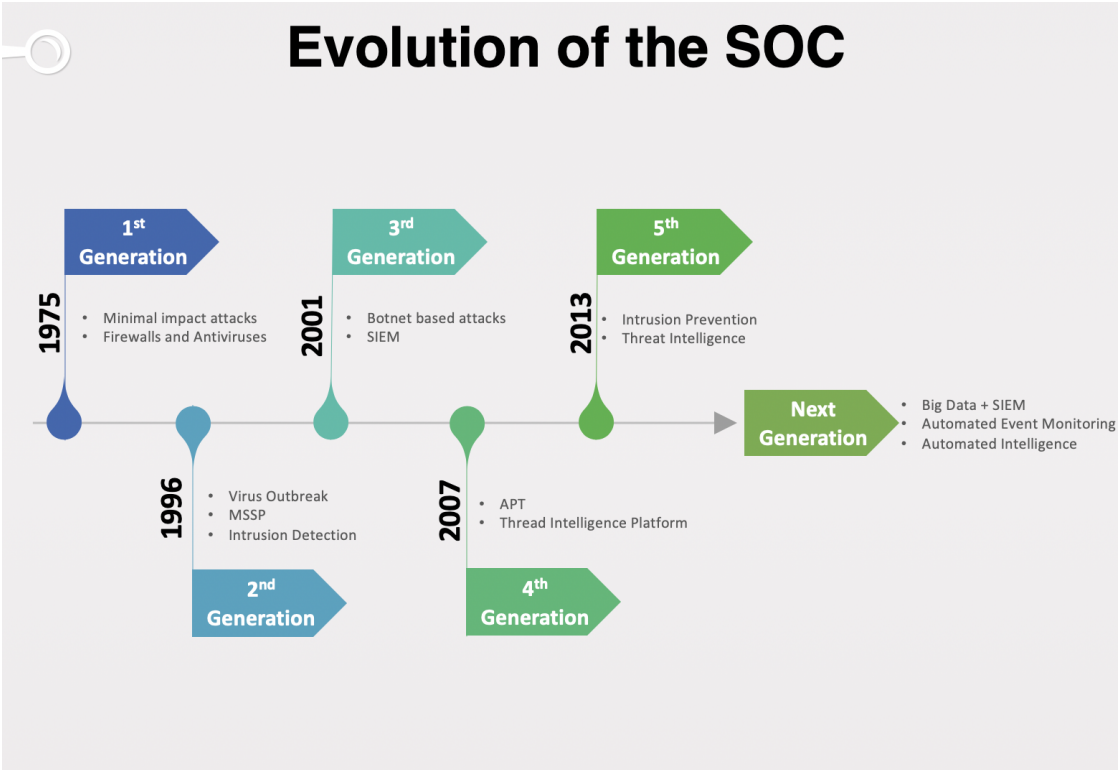


Figure 2.8: Evolution of the SOC

Previously, it was clarified why the SOC was established and how it evolved historically from the end of the 20th century until today. On this note, the following sections aim to answer the list of questions below:

- What are the main components of a SOC and how do they work together as a whole system?
- What are the limitations faced by the current generation?
- What are some of the alternatives to an in-house SOC?

Since different industries focus on specific areas for protection, currently, there isn't a particular standard for how a SOC is built. To clarify, an electrical utilities organization will focus on understanding what is happening in their environment in order to prevent attacks that might affect their ability to deliver electricity to their customers [34]. However, as is illustrated in Figure 2.9 there are three main components that are diagonal to every SOC and those are its Staff, Technologies and Processes [61, 13].

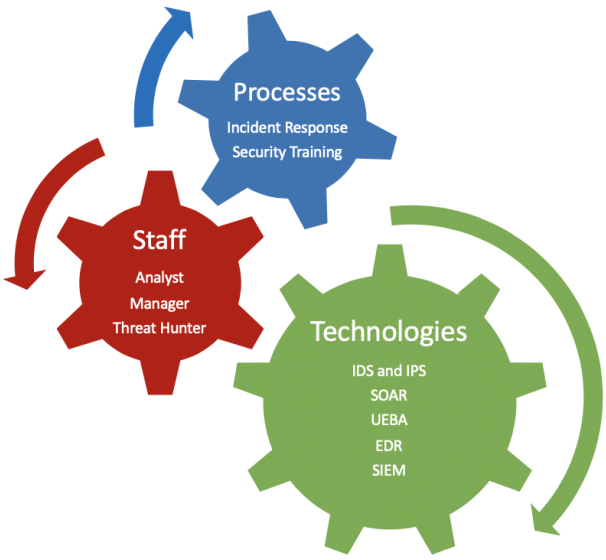


Figure 2.9: The 3 main components of the SOC.

2.2.2 Staff

Despite the fact that humans do not possess the ability to process data at the rate computers are programmed, they are still key for the successful operation of a SOC. In fact, machines lack common sense and cannot make intuitive informed decisions that can be achieved by the human brain. For this reasons, a team of experts is vital for the smooth operation of a SOC. A SOC team continuously monitors and analyzes the security infrastructure of an organization for any potential cyber threats. To clarify, the SIEM system,

which collects logs and events from hundreds of security tools and organizational systems, generates actionable security alerts, which the SOC team can analyze and respond to. Finally, in a typical SOC, staff members are divided into 4 tiers which are illustrated in Figure 2.10[65].

- The **Tier 1 Security Analyst** is a triage specialist and is the first human line of defense. Tasked with reviewing the latest alarms and alerts and identifying whether they are justifiable as security threats or rather classified as a false positive. Finally, in case if an alert that raises awareness cannot be solved at this stage it is escalated to the Tier 2 Specialist for further investigation.

- The **Tier 2 Security Analyst** is an incident responder responsible for the review of trouble tickets generated by the Tier 1 Analyst. Conducts a more in-depth analysis through leveraging Threat Intelligence capabilities, in order to single out infected systems and to evaluate the extension of an attack. Once again, if the tier 2 analyst cannot fully understand the incident and still has questions he is not able to answer, the incident is escalated to the tier 3 analyst.

- The **Tier 3 Expert Security Analyst** is a threat hunter who reviews asset discovery and vulnerability assessment reports. In charge with handling critical incidents escalated by tier 1 and 2 analysts. Leverages Threat Intelligence techniques to identify threats lurking within the network and runs penetration tests in order to find vulnerable entry points and attack vectors.

- The **Tier 4 SOC Manager** supervises, maintains and manages the SOC team. Reviews incident reports and develops crisis communication plans. Evaluates SOC performance through key performance indicators, like average incident detection time and average time till remediation.

2.2.3 Technologies

The SOC employs a wide variety of security tools for successfully monitoring an organization's systems and network infrastructure. According to [68], the technological component of the SOC can be segregated into **Data Collection**, **Analysis & Detection** and **Presentation**.

Over time, organizations have to evaluate the performance of the implemented SOC, as a way to assess and improve its productivity, defense capabilities and situational awareness. As we have already seen when the topic of risk assessment was touched upon, conducting a proper assessment goes through a definition of the context of the evaluation. Therefore, each organization should define which devices to monitor, what data to collect from them and in what format will this information be stored. Different Security Operation Centers implement a wide range of security tools, each being its own

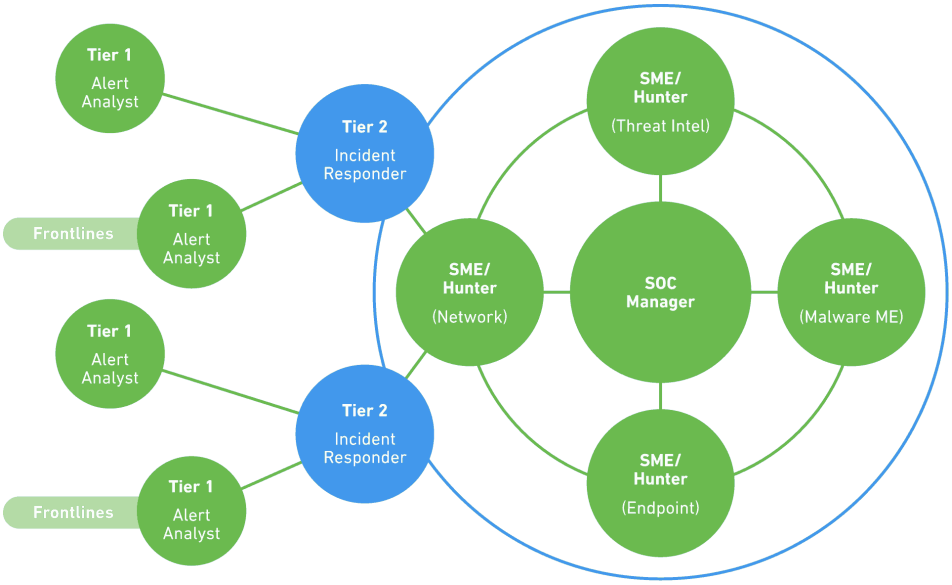


Figure 2.10: The different roles of the Staff arranged in tiers [63].

data source. Hence, common sources of data are: event data gathered from Intrusion Detection/Prevention Systems (IDS/IPS), sensors deployed over the network, anti-virus software, vulnerability scanners, firewalls, information collected from identity and access management software and even intelligence collected from external TI Feeds and TIPs[68]. Afterwards, through the use of a system such as the SIEM, data is aggregated, thoroughly normalised and correlated in a format that is useful for review and analysis. On this note, it is important to define the SIEM in the context of the SOC as it is the central technological component. Originally, the Security Information and Event Manager (SIEM) was actually two separate systems, the security information management (SIM) and the security event management (SEM) systems. Whereas the SIM provided log management capabilities, such as real-time log monitoring and analysis of different types of data, the SEM worked as a correlations engine that looked closely at specific types of events in order to find suspicious incidents. The fusion of the two systems gave birth to the SIEM, providing the visibility and log information of the SIM with the event correlation capabilities of the SEM. Finally, since it aggregates all events and log data from every endpoint and network device, it provides great visibility into the system and it is an effective tool for threat analysis detection.

The second component is also key for a successful SOC implementation. As large amounts of data is being gathered and correlated inside the SIEM, in order to effectively detect potential security incidents, automated detection tools were devised such as the Endpoint Detection and Response (EDR) and the User and Entity Behavioural Analytics (UEBA). Despite already having threat analysis detection, other technologies were introduced to aid the SIEM. Firstly, the EDR thrives to find threat patterns in data monitored and collected from endpoint devices. Finally, UEBA builds a model of "normal traffic"so

that it can pinpoint abnormal events through artificial intelligence and machine learning.

Finally, the presentations layer helps the SOC staff with communicating data regarding SOC performance over time to the upper management. On top of this, clear presentation of data enhances and facilitates decision making purposes, as trends inside the data are easier to detect. More information surrounding the topic of data visualization can be consulted in the next section.

2.2.4 Processes

SOCs rely on processes, protocols and policies in order to operate effectively, promptly mitigating any potential cybersecurity threats. This component unites the technological and human components of the SOC, since it defines the actions that analysts should take, for example in case of a security breach through an Incident Response Plan. In fact, procedures commonly employed by a typical SOC are, for example, a Incident Response and Handling Procedure.

Incident Response protocols are paramount for the quick detection of incidents and in order to minimize damage. To illustrate, the NIST SP 800-61-r2: Computer Security Incident Handling Guide is an example of a procedure developed by the National Institute of Standards and Technology (NIST) which "seeks to assist organizations in mitigating the risks from computer security incidents by providing practical guidelines on responding to incidents effectively and efficiently. It includes guidelines on establishing an effective incident response program, but the primary focus of the document is detecting, analyzing, prioritizing, and handling incidents"[48].

2.2.5 Challenges and Limitations

According to previous information, it can be inferred that the SOC can be an extremely competent and powerful tool providing specialists with ways to protect an institution from threat actors. Despite its benefits, according to [23], only around 26% of breaches in the last 12 months were detected by the SOC of several organizations, showcasing there is still room for improvement. To clarify, common limitations and challenges faced by current SOC teams are:

- **Large Amount of Data** - The amount of data that analysts are gathering is only increasing. With the large amount of security tools, which are constantly generating log data, the process of analyzing all this information can be quite time-consuming. To illustrate, even simple security tools like firewalls, anti-viruses, IDSs and IPSs provide valuable logging messages [26]. Despite their benefits, the process of trying to discern real threats from benign ones is challenging. Thus, if too much information is fed to the analysts they can quickly become overwhelmed and cannot accurately classify data between all the existing noise. Figure 2.11 illustrates how the amount of data gathered correlates to its value, taking into account an analyst's

ability to process it. As the amount of events per day grows the value of data increases until it reaches a point where analysts cannot keep up with the permanent inflow of data and are overwhelmed, introducing inefficiency.

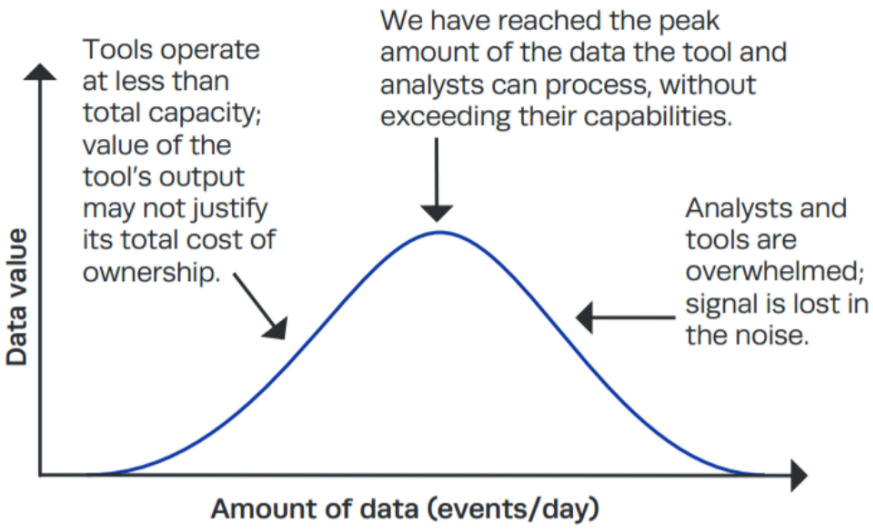


Figure 2.11: Correlation between the amount of data gathered by an analyst with its value [29].

- **False Positives** - The increase in data gathered by security tools triggers more alarms, in turn leading to more false positives alerts generated by the SIEM. A false positive alert reflects an incorrectly classified event as a potential threat when in reality it is harmless. Despite the fact that SIEMs employ precise rules in order to effectively pinpoint potential anomalies and atypical behaviour in the network, if this same sets of rules are not well defined they can produce an exorbitant number of false positives. Therefore, a potential tool that is supposed to help the SOC team ends up becoming a hindrance. To clarify, a study conducted in 2019 infers that 25% of an analyst's time is wasted chasing this incorrectly classified events [10]. Finally, a potential solution for this trend would be to automatically address low-level alerts via the inclusion of a SOAR platform, thus leaving more severe alarms to the meticulous care of the SOC team.

- **Lack of Specialized Professionals** - In order to swiftly answer to any possible threats that could strike at any moment's notice, the SOC line-up has to be on high alert 24x7. On top of this, the previous points highlight the stress associated with their position. In fact, according to [10, 56], this job is so taxing and psychologically demanding that "eight in 10 teams experienced measurable churns"in 2018. Furthermore, the lacking number of experts in the field is also one of the mains causes for this issue, as the same source infers that "skills shortage is typically cited as the number one challenge in maintaining SOC efficacy".

- **Widening Attack Surface** - Every year the number of cyber attacks is growing at an alarming speed. To illustrate, Figure 2.12 showcases how the growth in number of records breached tripled from 2018 to 2019, where it reached a 8.5 billion in number of occurrences [77]. One of the main reasons behind this growth is the constant widening in the attack surface, where phishing emails, exploiting vulnerabilities on mobile apps and leveraging Internet of Things (IoT) devices constitute a few examples. In fact, just to showcase the wide number of attack vectors available just from IoT devices, the chart below exhibits its growth on a year basis.

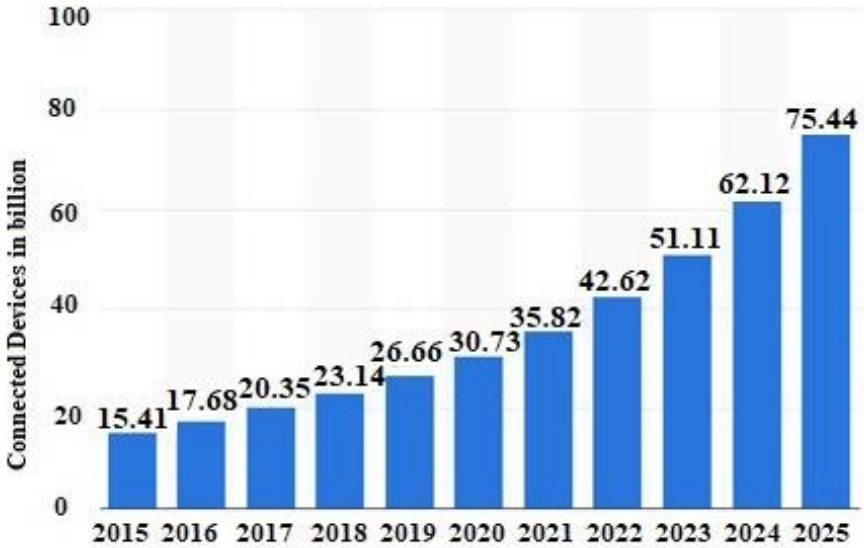


Figure 2.12: Internet of Things (IoT) connected devices from 2015 to 2025 (in billions) [2].

- **Lacking Data and Network Visualization** - Collecting and establishing proper organization of data is an important step, enabling analysts to accurately understand the current cybersecurity situation. As it has been stated, with the large amount of data gathered there are high chances that this intelligence will generate more noise than actually be helpful. Therefore, by employing competent data visualization tools, like business intelligence software, which has the potential of being infinitely customizable, it allows cybersecurity teams to visualize the important data in a way that makes the most sense to them. Additionally, having limited access to the quantity and quality of devices connected to the network jeopardizes the efficiency of the SIEM and complicates the analysts' decision making, due to lack of information. To clarify, according to [3] "the top reason for SOC ineffectiveness, according to 69 percent, is lack of visibility into network traffic". This statement corroborates the information above, since then vast amount of data generated by perimeter security tools is difficult to analyze unless it is visualized in an organized and centralized fashion.

2.2.6 Services

Despite providing measures to counter threat actors, in-house SOCs are extremely expensive infrastructures to build and maintain reaching up to 2.86 million dollars in annual fees [59]. In fact, not every organization has the funds to spare for the construction of such a facility, not to mention, the recruitment of a team of experts to maintain its operation. Therefore, there are some alternatives, for small to midsize enterprises (SMEs) to hire a similar service at a smaller cost.

The first, which is one of the most common security operations, is hiring a **CSIRT**. Whereas a SOC is broader in a protection oriented scope, the CSIRT "is assigned the responsibility for coordinating and supporting the response to a computer security event or incident"[52]. In other words, the team focuses on effective and quick incident response. Additionally, in case it is working under a SOC, they can also leverage threat intelligence to detect threats. Ultimately, if a smaller organization has as its top priority a quick and swift incident response in order to minimize damage caused by cybersecurity threats, the CSIRT constitutes a viable solution.

Secondly, **SOC-as-a-Service (SOCaaS)** is the ideal service for companies with a restricted budget who still want to have access to real-time threat detection and responding capabilities. SOCaaS is a cloud-hosted multi tenant software-based service that outsources a SOC's main components, in other words, its Staff, Technologies and Processes. In fact, analysts, incident respondents and the SIEM are all located and operated offsite by the service provider. Finally, this solution can improve the security posture of an organization since it includes features such as 24x7 network monitoring and log data collection, threat detection and incident response [13].

Lastly, the **MSSP** outsources security services for Small and medium-sized enterprises (SME). Despite the fact that both SOCaaS and MSSPs offer similar services, there are some key differences between the two. To illustrate, whereas a SOCaaS primarily focuses on the monitoring component, a MSSP on top of offering a continuous monitoring of network traffic and log management, it provides remote device management of security products like firewalls and intrusion detection/prevention systems [13]. Finally, this service is important in the context of this thesis since EY offers MSSP services to its clients, including the one in which the deployment of the dashboard solution will occur.

To sum up, no security service offers everything and addresses every need. Each one is tailored for a specific approach and enterprises have to choose the most appropriate solution according to their requirements and budget.

2.3 Situational Awareness Dashboard

As it has been previously stated, the financial sector is the biggest victim of attacks conducted by threat actors, being the most targeted industry for several years in a row [77]. Therefore, financial institutions need to invest more into detection and response, thus

aiming to reduce two main KPIs: the time to detect (TTD) and the time to recovery (TTR) [5]. Despite the fact that in past generations, fame was the main motivating factor behind attacks, nowadays money is what drives threat actors, and accordingly, banks is where the money is at. Yet, the concept of money is not limited to its physical or, in this case, digital definition. Instead, it can be translated into the theft of financial records, a customer's confidential data or employee records. However, why are attackers so successful despite constant technological innovations? Previously, one main obstacle was highlighted as one of the sources that inhibit a SOC from being efficient, that being: a lack of visibility [5] due to the exorbitant amount of data that is collected by cybersecurity tools. Thus, implementing a Dashboard that would aggregate relevant data into intuitive metrics regarding the performance of the SOC and that highlights its most vulnerable areas, would tackle the visibility problem, as well as helping to introduce a much needed Situational Awareness component. But what is a Dashboard and how can it help organizations?

A Dashboard is a type of visual interface that provides the user with quick to understand information relevant to a particular context. Furthermore, in the cybersecurity context of an organization, this concept would take the form of a graphical-user-interface which would convey to the user different metrics. Typically, these metrics are used by SOC analysts as a means to provide actionable information for decision making purposes. Finally, the Dashboard should be implemented on two foundations, a visualization component and an intelligence component. The following sections will review each one: Data Visualization and Metrics. Finally, this section is closed with two Situational Awareness Dashboards used in real life examples.

2.3.1 Data Visualization and Business Intelligence

Business Intelligence (BI) refers to the "analytical, technology supported process which gathers and transforms fragmented data of enterprises and markets into information or knowledge about objectives, opportunities and positions of an organisation"[75]. In other words, business intelligence covers the strategies and technologies employed for data analysis and subsequent visualization purposes.

In this digital era of Information Technology, the exponential amount of data generated by security tools is introducing a problem for primitive SOC instances without mechanisms to deal with the emerging concept of Big Data. As we have seen, analysts cannot keep up with the amount of data being generated. Coincidentally, Business Intelligence Software tackles some issues including this one as well as proportioning a set of benefits listed below [22]:

- **Fast and accurate reporting** - Through the usage of templates and automation of KPI calculations, the staff can implement real time reports that showcase relevant and accurate data for further inspection. Additionally, this eliminates manual tasks further improving efficiency and time consumption.

- **Enhanced visibility** - Offers needed visibility into business performance through the establishment of metrics and KPIs. In turn, this level of visibility into data builds a more situational aware archetype as different data sources are aggregated for a fuller picture of what is happening to ones business.
- **Improved decision making** - Enhances and shortens the decision making process through the support of actionable intelligence.
- **Enhances business productivity and operational efficiency** - Since there exists real time monitoring of the SOC's performance this makes the process of establishing goals and plans to reach objectives more efficient and intuitive. In the same way, it helps to pinpoint under-performing areas to cut costs or improve.

Therefore, implementing a business intelligence system should be a top priority for every organization. In order to develop a business intelligence system, different components are necessary. According to [25], those components are:

- Operational data sources or databases of structured and unstructured data.
- A process of collecting data from the various sources, as well as being capable of transforming it into a normalized format and storing it.
- A data warehouse that works as the central database where parsed information is stored.
- Analytical tools/software that translates the data stored into actionable intelligence.

To sum up, BI is a competent data analytics tool that offers various enhancements to business operations ranging from improved decision making to enhancements on business productivity. Furthermore, in order to build a BI system a set of components are needed to transform raw business data into a normalized and unified format for analysis. Finally, BI software tools are able to translate that intelligence into a visual format for a more intuitive representation through the use of metrics. But what metrics are interesting and how can we define useful and actionable metrics in a cybersecurity context? The next section aims to answer these questions.

2.3.2 Metrics: KPI and KRI

Metrics are quantifiable measures used to track and assess business processes. In order to effectively assess the cybersecurity landscape and identify patterns and trends of malicious activity, a selection of adequate metrics is essential. In fact, if too many and/or unnecessary metrics are being monitored and displayed this could lead to the introduction of noise and obstruction of visibility, thus jeopardizing the decision making process. Therefore, when selecting the most appropriate metrics, quality should be valued over

quantity and the first step should focus on identifying the most critical security operation functions performed by the SOC [41].

Manfred *et al.* [68], describes four types of typical SOC metrics: General, People, Technical and Governance and Compliance.

1. **General** - Can be divided into **Coverage** metrics, related to the amount of assets being monitored, and **Key Performance Indicators (KPI)** which are used to measure the performance of the SOC over time.

2. **People** - Metrics used to improve the performance of analysts inside the SOC by measuring their workflow.

3. **Technical** - Divided into Threat, Vulnerability, Risk, Alert, Incident and Resiliency metrics. **Threat metrics** refer to the threat level of vulnerabilities. **Vulnerability metrics** are interesting since they give oversight into weak spots attacker might exploit. **Risk Metrics** or Key Risk Indicators (KRI) are frequently calculated in real time and are important to assess the current security posture of the organization, contributing to the overall situational awareness. **Alert metrics** are related to the alerts generated by the SIEM or other security tools. **Incident metrics** refers to occurrences that aim to harm the organization, therefore keeping metrics that summarize this information is important. Finally, **Resiliency metrics** are associated with the ability to continue business operations after an attack.

4. **Governance and Compliance** - Segregated into **Compliance** metrics, which check if the SOC is compliant with all regulatory guidelines and standards, and **Maturity** metrics, which are used to track the maturity level of the SOC at any moment.

According to [41], the SMART methodology is a model that verifies if a metric is suited to be used in a business context and be used for decision making purposes. To clarify, following this rule of thumb any metric should be considered Simple/Specific, Measurable, Actionable/Achievable, Relevant and Time Based.

1. **Simple/Specific** - A quality KPI should not be overly complicated to calculate and measure. For visibility purposes clarity is key, and analysts should be able to understand its meaning clearly.

2. **Measurable** - Not only should an optimal KPI be quantifiable, but also the methods used to measure it should be consistent.

3. **Actionable/Achievable** - It should serve its purpose, meaning through the information it provides it should lead analysts to make decisions. Otherwise its purpose should be reevaluated.

4. **Relevant** - If a designated KPI does not align with the functions of the SOC it is ineffective.

5. **Time Based** - Effective KPIs should be relevant over long periods of time with results being interesting to review either not too frequently nor too rarely.

To sum up, as different organizations employ different security strategies and goals the set of KPIs to measure varies according to the circumstance. Therefore, each team of analysts is tasked with selecting proper metrics to fulfill their needs. To illustrate, a few examples of KPIs employed in SOCs are as follows: Mean Time to Detect (MTTD), Mean Time to Resolve (MTTR), Number of Incidents per Analyst, Number of False Positive Alerts and Time between Threats [41].

2.3.3 Related Work

This section showcases two examples of Situational Awareness Dashboards both applied in the Healthcare Industry. Unfortunately, due to the sensitive and private nature of these types of systems, no concrete examples of these platforms, in a SOC context, were found. However, KPIs recommendations can be used to partially illustrate what these Dashboards would provide, like for example, the ones mentioned in the end of section 2.4.1.

Amidst the COVID-19 pandemic which is currently being experienced globally, a visual representation of intelligence can help researchers identify patterns, understand the current situation being lived across a designated country and predict how this pandemic situation may evolve into the future. On this note, the following examples showcase two different countries which are implementing SA Dashboards, giving them access to the landscape of the Coronavirus outbreak.

2.3.3.1 COVID-19 Situational Awareness Dashboard: Canada

The Public Health Agency of Canada in collaboration with other entities developed a Situational Awareness Dashboard. To illustrate, the image below depicts a glimpse of the charts and metrics provided (Figure 2.13).

For instance metrics provided are: Total Cases by Providence, Time Charts of the Number of Cases/Deaths and the Number of Cases by Age and Sex.

2.3.3.2 COVID-19 Situational Awareness Dashboard: Portugal

Once again related to the COVID-19 Pandemic, the "Direção Geral de Saúde"which represents the healthcare authority inside the Portuguese government developed the following dashboard displaying the evolution of the COVID-19 pandemic (Figure 2.14).

For instance metrics provided are: Total Cases by Region, Total Number of Cases/Deaths/Recoveries and a chart with the evolution of the suspected cases from the beginning of the outbreak until the present.

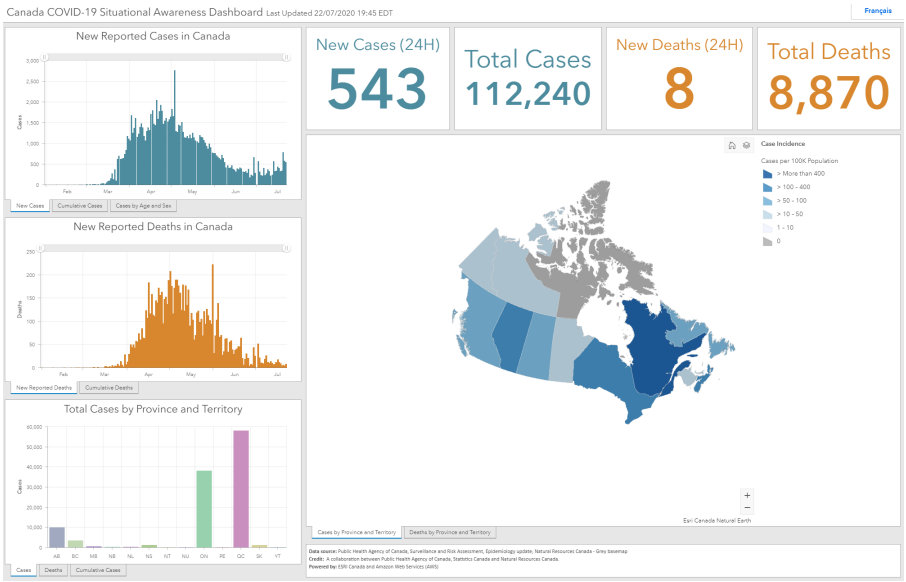


Figure 2.13: Canada COVID-19 Situational Awareness Dashboard [7]

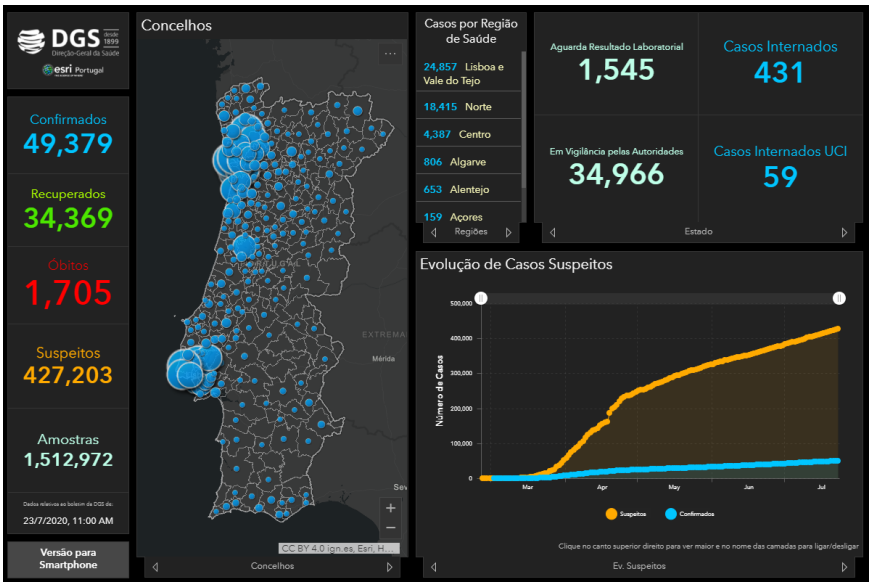


Figure 2.14: Portugal COVID-19 Situational Awareness Dashboard [49]

2.4 Final Thoughts

Over the years, the concept of cybersecurity evolved from simple firewall and anti-virus solutions to the current generation, where complex SOC infrastructures, operated by teams of experts, are tasked with real-time monitoring, detection and analysis of incidents. Alternatively, services like SOCaaS and MSSPs became available to small and medium sized businesses in case they did not possess the budget to acquire a costly infrastructure like the SOC. Despite their inherent usefulness, all these systems still faced many challenges such as having to deal with overwhelming amounts of data and alerts

generated by the SIEM which in turn created noise that would hinder visibility, specifically affecting its accessibility to reliable and meaningful information. Thus, in order to protect individuals from the invisible but constant threat that are cyber attacks, improving a SOC's visibility into intelligence constitutes an essential solution. Reaching this goal involves acquiring, (i) a perception of the organization's security posture and its threat environment through CTI, (ii) a comprehension of the security risk faced at any given moment and (iii) projecting this data into a viewable format. These three elements are the foundations behind establishing a cyber situational awareness landscape.

In fact, the third and last point is particularly important as it is the one that enables analysts to have a constant and informative visual representation of the risks and security posture of a designated entity, ensuring it is able to perform informed decisions through a Situational Awareness Dashboard. In fact, to successfully build such a system, a high attention to detail should be given to what KPIs should be tracked and how data should be displayed so as it is concise, intuitive and noise free. The following steps would have to be taken: gathering data from internal sources (organization) and external sources (OSINT), parsing and centralizing intelligence, implementing the dashboard with its graphs and charts and finally analyzing this data for monitoring purposes. As a result, this platform smooths the decision making process and improves monitoring by letting one have access to the bigger picture, ultimately helping to diagnose problems and allowing informed business decisions to be made. Thus, the next chapter aims to briefly describe the architecture of the solution that was developed during the development phase of this thesis, as well as some alternatives to the current architecture. Finally, on the topic of data visualization, according to [33], "graphical representations help you immediately identify outliers, detect malicious activity, uncover misconfigurations and anomalies, and spot general trends and relationships among individual data points".

A picture is worth a thousand log entries.

3

System Architecture

The following chapter's structure is divided into 5 main sections. The first section is used to introduce the main architectural components of both systems that were developed (3.1). Afterwards, the second(3.2) and third(3.3) sections are used to describe each of the three mains components of both systems, those being the ingestion, storage and visualization layers. Furthermore, these sections aim to introduce the software that was implemented, as well as some alternatives that were investigated and experimented upon before the sedimentation of each system architecture. Finally, the fourth and final section (3.4) of this chapter is used to sum up what was learned through experimenting with different software, during the implementation of both systems.

3.1 Bird's High View of both Systems

In the previous chapter, several important subjects were addressed in order to properly understand the landscape of the work conducted in this thesis. Not only were some core concepts such as Cybersecurity, Threat Intelligence and Data visualization defined, but it was also described the environment and purpose of the solution.

This chapter will overview the architectural components of both systems that were developed through the course of this dissertation. The first system ended up being used to experiment upon several different types of software, which in turn made it easier to establish the architecture of the second system. Hence, all the iterations of the solution of Systems A and B will be examined with justifications as to why each solution was developed with their specific components.

In order to develop both solutions, a system architecture was orchestrated that would allow the Dashboard to be as self sustained and automated as possible. As such, the devised system architecture can be segregated into three different components:

- **Ingestion Layer** - This layer is in charge of all the data gathering and parsing tasks that are used as the building blocks for constructing the visualizations and KPIs that are present in the Dashboard. As such, the ingestion layer's main purpose is

to gather raw data and refine it into actionable intelligence. The Ingestion Layer is directly connected to the Storage Layer.

- **Storage Layer** - The storage layer encompasses the Database, which is used to store all the data that extracted by the previous layer. Hence, on top of being connected to the Ingestion Layer, the storage layer has a direct link to the Visualization Layer, in order for the Dashboard to access the data that enriches its visualizations and metrics.

- **Visualization Layer** - Finally, the visualization layer is composed by the data visualization software incorporated in the solution, which as we have seen is connected to the Storage Layer.

Now that the components of both systems have been introduced, the following sections are used to describe each one in fine grained detail.

3.2 System A

System A's main purpose of development was to fulfill the main goal of this thesis, developing a Situational Awareness Dashboard. This Dashboard would allow users to grasp the cybersecurity situational picture of the organization through a set of strategically picked metrics, KPIs and KRIs showcased through different types of visualizations. Additionally, this set of visualizations would not only gather resources from internal intelligence, but also threat intelligence feeds and third-party calculated security scores to strengthen its model. Ultimately, due to the fact that the Contractor Agreement, between *EY* and the client ended, this system never left the mockups/testing phase.

Before diving into the system architecture it is important to introduce how the first architectural model came to be, since it paved the way towards reaching the final model. Hence, after a brief introduction the final model will be presented and each component detailed.

During the research phase of the dissertation, The *ELK stack* was one the first tools of implementation that were investigated. The *ELK stack* is a bundle of three widely used open source projects developed, managed and maintained by Elastic: *Elasticsearch*, *Logstash* and *Kibana*(Figure 3.1). Each one of these three products reflects the components that were described earlier with *Elasticsearch* being associated with the Storage Layer, *Logstash* the Ingestion Layer and *Kibana* the Visualization Layer.

As such, an architectural model composed by the *ELK stack* was the first iteration during System A's development. Nonetheless, two of the three components of the *ELK Stack* were changed for other alternatives. This matter is explored later on in this chapter.

Figure 3.1: ELK Stack: Elasticsearch, Logstash, Kibana

3.2.1 Architecture

As illustrated in Figure 3.2, the final architectural model of System A is composed by the following components: *Python Scripts* are in charge of the Data Ingestion Component, *Elasticsearch* is the core of the Storage Component, in other words, it is the database that stores the information and *Power BI* is the Visualization software used to build the Dashboard.

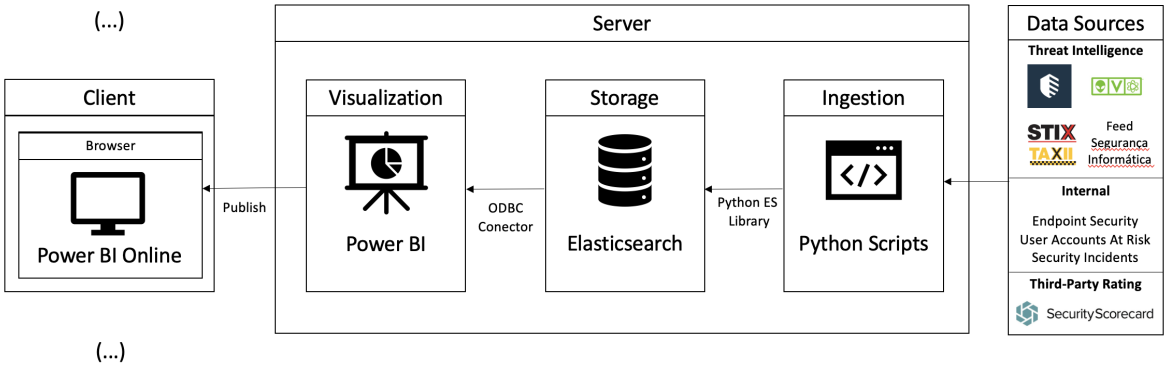


Figure 3.2: System A - Architecture Model

The process of establishing this architectural model started with the *ELK Stack* as its starting point, in other words, with *Logstash*, *Elasticsearch* and *Kibana*. Afterwards, a different iteration of the solution was experimented upon, one which involved changing the data ingestion tool from *Logstash* to a scheduled call on Python Scripts, that would load the data from different sources to the database. Finally, the final iteration of the solution substituted the Visualization Software from *Kibana* to *Power BI*.

The next subsections introduce the different software tools experimented upon on each component, coupled with the pros and cons of each which, ultimately, lead to the establishment of the final architectural model showcased above.

3.2.2 Ingestion Layer

As it has been described, the ingestion layer is tasked to perform all the ingestion of raw data, which is further enriched and converted into actionable intelligence. Different sources were used to extract data about external threats (through Threat Intelligence) and internal risks. Furthermore, a security score calculated by a Third-Party organization, was used to back up the previous information and serve as another measure to calculate the level of risk the designated entity is susceptible to. Further information regarding this topic can be consulted in the next chapter (Chapter 4).

The following sections aim to describe each software tool which was once incorporated in the Ingestion Layer.

3.2.2.1 Logstash

"*Logstash* is a light-weight, open-source, server-side data processing pipeline that allows you to collect data from a variety of sources, transform it on the fly, and send it to your desired destination"[60]. Typically, Logstash instances are connected to *Elasticsearch* due to its inherent compatibility. Overall, *Logstash* has high versatility of ingestion, pre-built filters and flexibility to output data to one's designated repository.

Regarding ingestion versatility, this tool offers the ability to ingest data from many available sources through a flexible plugin architecture of over 200 plugins which users can leverage to customize their data pipelines [60]. Additionally, *Logstash* has an API for the development of plugins, in case there is no alternative that satisfies a user's criteria.

This tool also allows data to be filtered through pre-built filter mechanisms that smooth out the data transformation process. For instance, filtering capabilities range from parsing dates and timestamps fields, to even being able to contextualize fields containing IP addresses with geographical information [30].

Concerning output flexibility, *Logstash* offers a wide range of output plugins to send event data into. As an example, the user has the flexibility to output data to Databases, such as *Elasticsearch* and *MongoDB*, send events to a generic HTTP or HTTPS endpoint or even write events to disk in a delimited format [31].

To sum up, below is a list of the pros of implementing *Logstash*:

- **Open-source** - Since all the of the software components of ELK are free and open-source, it has a low financial barrier to entry.

- **Configuration** - A straightforward configuration process which keeps configuration files in a plain text format.

- **Integration** - Tight integration into Elastic's remaining products, therefore minimal setup is required when using those tools.

- **Versatility** - It lets one write simple rules to programmatically load and transform data in real-time from different sources, making it possible to change data on the fly.
- **Flexibility** - Filter plugins are powerful for extracting and enriching input data.
- **Speed** - Makes editing configuration files directly a possibility, so that once a change is made, reloading the service reflects those modifications in one's *Logstash* instance.
- **Documentation** - Despite being a recently developed software, documentation is extensive and organized.

In the same way, below is a list of the cons of implementing *Logstash*:

- **Performance and resource consumption** - *Logstash* runs on the Java Virtual Machine (JVM). This means that Logstash will always use the maximum amount of memory one allocates to it. Memory usage grows linearly as the JVM keeps allocating to the heap until it runs out of space. Afterwards, it runs a garbage collection process, which frees up some space of the heap. Hence, as times goes by performance starts to decrease since the application may attempt to put more data into the heap despite the fact that there is insufficient room for it.
- **Community Support** - As a recently developed software, adoption is still mature, and finding solutions to some problems online is still somewhat challenging.
- **Complexity** - Despite being relatively straightforward to configure, as more filters and data transformation operations are added, the complexity of the configuration file also increases, more so if there are multiple configuration files for different data pipelines. This factor inevitably complicates the readability and future revision of the solution.
- **Testing and Debugging** - *Logstash* does not offer any type of debugging capabilities besides the existence of online *grok* testing tools. *Grok* is one of most used data filter plugins, which helps users parse log files.
- **Learning Curve** - One of the biggest shortcomings of *Logstash* is closely tied to its wide array of different plugins. With so many different modules to ingest, filter and output data it ends up overwhelming users who have never experienced working with the software beforehand. Hence, there is the presence of a time-consuming learning curve.

3.2.2.2 Python Scripts

Another option that was considered throughout the development of the solution, was to extract and transform data through a set of programmatically designed scripts, written

in Python. These scripts would be designed to extract data from REST APIs, files or other sources and be scheduled to execute on a daily basis, in order to add/update the database with new information.

To sum up, below is a list of the pros of opting for a scripted route:

- **Simple** - Scripting is a generally simple task to achieve, since knowing how to write code in a programming language is the only technical requirement.
- **Cost** - As long as the organization has IT personal who have programming knowledge, scripting is free and has no software or licensing costs associated with it.
- **Automation** - Scripts are good candidates for automated tasks, such as file management procedures, as they can create/update/delete files and folder. Additionally, they can be used to trigger other complex tasks, such as, checking for the existence of a file inside a folder and copying the contents of it to a local database.
- **Control** - Unlike other types of software, which usually only give the user limited control over the procedures by setting up a set of parameters, scripts are a type of solution that give the user granular control over the tasks being performed.
- **Flexibility and Versatility** - Scripts are flexible solutions because they are highly extensible and versatile. Users can develop complex solutions in different programming languages, which can be passed down and changed by other members of the team.
- **Testing and Debugging** - Since users have a higher degree of control over the solution, they can more easily test it for errors in a controlled environment and debug it for errors on its execution. Users also have full control over errors being able to create exceptions and even record in a separate file a summary of the operations performed.

In the same way, below is a list of the cons of opting for a scripted route:

- **Complexity over time** - Despite being relatively easy to implement for simple task automation, over time scripts end up becoming a complex solution. In fact, as time goes by and the size of the solution increases, more files, more methods, more variables and more lines of code, inevitably increase the complexity, readability and revision of the script.
- **Maintenance** - In the same way complexity increases over time, maintaining the script is also a cumbersome task for analysts.
- **Time consuming** - Implementing scripts to automate simple tasks is a straightforward non time consuming process. Nonetheless, over time the process of maintaining and changing the script can be time consuming.

3.2.2.3 Logstash vs Python Scripts

As we have seen, Logstash's event processing pipeline has three different stages: *inputs*, followed by *filters* and ending with *outputs*. *Inputs* generate events, *filters* modify them, and *outputs* ship them to another repository. Each data pipeline is orchestrated by writing a configuration file with its set of inputs, filters and outputs. Therefore, in order to implement the solution in hand, different configuration files would need to be built for each input, with its own set of filters in order to transform events and store them in the database. Despite of *Logstash's* inherent benefits in the context of our solution some of its shortcomings made it interesting to explore another approach to the process of data ingestion. In fact, the realization that its straightforward configuration, versatility and flexibility, did not out weight shortcomings like its over time complexity, the lack of community support and a steep learning curve (both fueled by the fact the software is immature in its production life cycle) enticed the idea of trying to experiment building the ingestion layer around another approach like scripting.

After experimenting with using scripts to extract data from some Threat Intelligence APIs it was concluded that, due to previous experience with programming, the process was faster than writing *Logstash's* configuration files. Additionally, troubleshooting errors and dealing with execution bugs would end up being easier to solve through scripts.

To sum up, both approaches would result in an automated data ingestion component. Nonetheless, the attribute that ultimately had the most weight when it came to deciding which approach to take was *Logstash's* steep learning curve compared to an existing knowledge in regards to scripting and programming overall.

3.2.3 Storage Layer

As it has been described, the ingestion layer is composed by the database that was used to store all the relevant information that is used to enrich the dashboard. Due to the fact that originally, the ELK Stack was the architecture that was going to be used, the Storage Component was chosen to be *Elasticsearch*.

3.2.3.1 Elasticsearch

"*Elasticsearch* is a distributed, open-source search and analytics engine built on Apache Lucene and developed in Java. It started as a scalable version of the Lucene open-source search framework then added the ability to horizontally scale Lucene indices. *Elasticsearch* allows you to store, search, and analyze huge volumes of data quickly and in near real-time and give back answers in milliseconds. It's able to achieve fast search responses because instead of searching the text directly, it searches an index. It uses a structure based on documents instead of tables and schemas and comes with extensive REST APIs for storing and searching the data. "At its core, you can think of *Elasticsearch* as a server that can process JSON requests and give you back JSON data."[18]. In order to better

understand how *Elasticsearch* stores and organizes data, below is a list of some of the core concepts behind this framework [17]:

- **Documents** - *Elasticsearch* stores complex data structures that have been serialized as JSON documents. Hence, a document can be defined as the basic unit of information, expressed in JSON, indexed in *Elasticsearch*. Documents can take the form of simple text files or even data objects structured in JSON containing primitive data types such as integers and strings.

- **Indices** - An index is a collection of documents with a similar data structure. For instance, an index entitled "Users"can be used to store information regarding different users with a similar data structure. Hence, each index has a name that is used to performs indexing/deletion/update/search operations against the documents stored inside it.

- **Inverted Index** - The inverted index is a mechanism inherent to each index which serves as a quick look-up of where to find search terms in a given document. An inverted index is a hashmap-like data structure that directs you from a word to a document. In other words, it is a data structure that stores a mapping from a document's piece of data to its locations in a document or a set of documents. Hence, this is the mechanism behind *Elasticsearch's* extremely fast search capabilities in large data sets.

- **Node** - Nodes are single servers which are part of a cluster. There are three types of nodes who overall cover an array of indexing, searching and management tasks. The **Master Node** manages the cluster and all the addition and removal operation of other nodes and indexes. The **Data Node** is tasked with all data-related operations such as search and aggregation. Finally, the **Client Node** manages node operation by forwarding requests to the appropriate type of node, in other words, it forwards data-related requests to Data Nodes and cluster requests to the Master Node.

- **Cluster** - A Cluster is an aggregation of Nodes which are connected together.

- **Shards** - An *Elasticsearch* index can be subdivided into different shards. These shards are hosted on nodes inside a cluster and are used to distribute different documents of the same index across multiple shards spread out multiple nodes inside a cluster. This functionality ensures **redundancy** protecting the data against hardware failure and increasing query capacity by the number of nodes in the cluster.

- **Replicas** - *Elasticsearch* index shards can be replicated once or more, creating different copies of the same primary shard. Hence, this feature further extends the redundancy functionalities provided by the shards mechanism.

To sum up, the main reason behind the decision of introducing *Elasticsearch* as one of the core architectural components was its integration in the ELK Stack. Nonetheless, as a non-relational database it is suited for the implementation of the solution.

As a side note, it is important to clarify how Python interacts with *Elasticsearch*, in order to perform all data-related operations. *Elastic* developed a low-level Python client for *Elasticsearch*, called *elasticsearch-py*. Through this client, it is possible to automate all index creation and document insertion operations.

3.2.4 Visualisation Layer

As it has been described, the visualization layer is composed by the data visualization software incorporated in the solution. Due to the fact that originally, the ELK Stack would be the architecture that was going to be used, *Kibana* was originally integrated in the Visualization Component, later substituted by *Power BI*. The following sections aim to introduce both frameworks and their respective functionalities, culminating in a final section that justifies why *Power BI* was the chosen architectural component.

3.2.4.1 Kibana

Kibana is a data visualization and management tool for *Elasticsearch* that provides real-time data visualization. Hence, it allows users to visualize *Elasticsearch* data through the use of histograms, line graphs, pie charts, maps and other data visualization techniques. Through its real-time data visualization organizations can be constantly aware of the status of their business operations.

When it comes to its features, *Kibana* offers different functionalities, such as, data visualization, data exploration, alerting, security, monitoring, management, share and collaborate and even machine learning. The following list entails on each topic mentioned above, with the aim to clarify the benefits and capabilities of implementing *Kibana*:

- **Data Visualization** - *Kibana* offers different ways to build one's ideal Dashboard with different data visualizations techniques. For instance, inexperienced users can take advantage of *Kibana Lens* easy-to-use and intuitive interface that simplifies the data visualization process through a drag-and-drop experience, whereas veteran users can leverage *Kibana's* more advanced data visualization tools.

- **Data Exploration** - Regarding data exploration, *Kibana* offers *Discover* and *Dashboards*. On the one hand, *Discover* enables users to explore and access every document in every index that matches the selected index pattern. Additionally, through this feature users can perform queries and filter search results. On the other hand, through the *Dashboards* functionality users can display a collection of visualizations which are updated in real time and create drilldowns between multiple dashboards to enhance the experience.

- **Alerting** - This feature enables alerts which identify users when some designated changes occur the data that is stored in *Elasticsearch*.

- **Security** - *Elastic Security* enables the SOC Staff with the ability to detect, investigate, and triage threats.

- **Monitoring and Management** - The *Elastic Stack* offers a variety of monitoring and management tools, such as: full stack monitoring, user and role management, license management and much more.

- **Share and Collaborate** - *Kibana* allows to share visualizations with other team members and offers the ability to share a link to a Kibana dashboard, or embed the dashboard in a web page as an *iframe*.

- **Machine Learning** - Through *Elastic's* machine learning features *Kibana* can automatically model the behavior of one's *Elasticsearch* data in real time.

Despite of the functionalities listed above, after experimenting with *Kibana* during the development phase of the mockup of System A, the following obstacles were encountered:

- **Premium Features** - Despite having a relatively low entry barrier (free), some of the most interesting features such as Machine Learning are blocked behind a subscription based service.

- **Inability to create visualizations using data from various indices** - In *Kibana* every single visualization created is tied to a particular index. This presented a major drawback since there were some metrics, such as the risk value extracted from the different Threat Intelligence sources, that could not be calculated since the *Indicators of Compromise* collected from each source were stored on its own independent index.

- **Third-Party Plugins** - *Kibana* provides the possibility of installing Third-Party Plugins to add some new type of visualization or interface element. Despite appearing to be a good solution for introducing some type of lacking feature, adding plugins to *Kibana* can be a very tedious process due to errors with version mismatching. Additionally, upgrading a plugin to a new version most likely will raise some incompatibility issues.

- **Lack of Control over Dashboard UI** - *Kibana's* dashboard building experience is conducted in two stages: making the visualizations and adding them to the Dashboard. The second stage is rather simple since the user only has to drag, drop and resize the chosen visualization. Despite its ease to use, this feature is a huge handicap for users that want to have full customization control over their Dashboards.

- **Dashboard Interactivity** - After experimenting with *Kibana*, compared to the other data visualization software (Power BI) there was a lack of interactive features present on the Dashboard.

- **Lack of Customizability** - Another point where there is a lack of customizability is on visualizations. For instance, color formatting is very limited not to mention in chart and graphs there is no *RGB* color selection, it is only possible to select one color out of a few samples out of a pallet.

- **Community Support** - As a recently developed software, adoption is still immature, and finding solutions to some problems online is still somewhat challenging.

All in all, Kibana is a potent open-source data visualization software with robust analytics, search and security features. Nonetheless, over the course of experimenting with this data visualization tool the list of obstacles encountered above led to the search for another alternative for the visualization component. Hence, after some investigation an alternative business intelligence software was found:*Power BI*.

3.2.4.2 Power BI

Power BI is a cloud-based business intelligence service suite developed by Microsoft. Analogous to other Business Intelligence services, its main purpose is to convert raw data into meaningful information by using intuitive visualizations so that one can easily analyze data and make important business decisions based on it. All in all, *Power BI* is a business intelligence tool which offers a bundle of data visualization tools such as software services, apps and data connectors.

Below is a list of some of the advantages and features of implementing *Power BI*:

- **Cost** - A major advantage of using *Power BI* is that it is a relatively inexpensive and affordable product. The Free License of Power Bi allows users to access almost all of the most important features of the product.

- **Simple** - *Power BI* is known for its simplicity and user-friendliness. In fact, a few hours working with the software were enough to start understanding how to connect data sources to Power BI and start building interesting visualizations.

- **Easily Accessible** - *Power BI* can be accessed from one's desktop computer, laptop, tablet or phone. This way teams can constantly access dashboards and reports anywhere and be engaged with each other.

- **Documentation** - Regarding documentation, Microsoft does a great job explaining every feature of Power BI and provides step-by-step guides on how to perform some complicated tasks.

- **Community Support** - Since Power BI is a tool that is widely used by organizations, there is an active community of other users who are willing to help. Nonetheless, there is a high probability the question someone is looking to get answered has already been answered by someone else who may have encountered the same issue. All in all, Power BI has a thriving community eager to help.
- **Wide Selection of Visualizations** - One of the main features of *Power BI* is its diverse visualizations. Hence, it becomes possible to create appealing dashboards and reports. Finally, it is also possible to create custom visualizations further enhancing and letting one personalize his Power BI experience.
- **Wide range of different ways to connect data** - Power BI offers users a range of structured or unstructured data sources to import data from.
- **Customizable Dashboards and Informative Reports** - In Power BI there are two types of top level visualizations: Dashboards and Reports. On the one hand, the Dashboard is single page and contains tiles, each representing a visualization created from a report. Therefore, Dashboards in Power BI should be used as a way to view some of the most important KPIs of one or more reports. On the other hand, Reports are a combination of multiple visual elements. Additionally, Reports can have up to several pages/tabs and are much more interactive in terms of data visualization, for instance, with the ability to apply filters on visualizations.

In the same way, below is a list of the obstacles and disadvantages of working with *Power BI*:

- **Premium Features** - Despite having a relatively low entry barrier (free), in order to have access to the ability to share Power BI reports inside or outside organizations, users must have access to a paid Power BI Pro subscription. Moreover, Power BI has a limit per data set of 2 GB of data for free accounts.
- **Rigid Formulas** - DAX stands for Data Analysis Expressions and is compromised of a collection of functions, operators and constants used to calculate new values. In other words, DAX is used to transform or create new information based on the data stored in Power BI's data sets. Despite of its powerful nature and ease to use in simple use cases, DAX is not the easiest language to work with. As the complexity of operations increases DAX formulas end up becoming long and complex.

3.2.4.3 Kibana vs Power BI

As we have seen, whereas *Kibana* is an open-source data visualization and management software built to complement *Elasticsearch* and the remaining ELK Stack, *Power BI* is more suited towards business intelligence and is more versatile than its counterpart, in regards to connectivity to different data sources.

Additionally, Power BI's reports are superior to Kibana's Dashboards due to their bigger degree of flexibility and customizability. Regarding pricing, Power BI is also superior in that category since the difference between the Pro and the Free version is negligible for single entities (Pro version gives permission to share Dashboards and Reports). On this note, in terms of collaborative work Power BI is once again, a notch ahead of Kibana, because of top of its ability to export data in different formats, such as CSV, and it allows users to publish data directly to websites.

To sum up, after experimenting with *Kibana* changing the data visualization tool to *Power BI* showed how superior the second is to the first. Therefore, it was decided that the visualization software that would be present in the development phase would be *Power BI*.

As a side note, it is important to clarify how *Power BI* has access to the data residing inside of *Elasticsearch*. This connection is made through an ODBC Connector. Briefly, "an ODBC driver uses the Open Database Connectivity (ODBC) interface by Microsoft that allows applications to access data in database management systems (DBMS) using SQL as a standard for accessing the data. ODBC permits maximum interoperability, which means a single application can access different DBMS"[70]. Elastic offers a compliant ODBC driver for *Elasticsearch*, therefore making the process of connecting *Elasticsearch* data to *Power BI* a possibility.

3.3 System B

Both Systems A and B had the same goal in mind, the development of a Situational Awareness Dashboard that would allow users to grasp the cybersecurity situational picture of the organization, through a set of strategically picked metrics.

System B was developed to fulfill a lacking visual component in another client of the financial sector. This client did not have access to a unified view of the performance and risks of the SOC. The key difference in this System compared to the previous one, was the involvement of the client. Whereas System A's client was not engaged in the development was that Dashboard, the client associated with System B was motivated with the solution and was engaged and cooperative.

Before introducing the final architecture and its components, it is important to take note that System B's Model was developed based on the experience that was gained from working with System A. Hence, this is the reason why two of the three architectural components remained the same. The next section will cover the final architectural model of System B.

3.3.1 Architecture

As illustrated in Figure 3.3, System B's architecture is composed by the following components: **Python Scripts** are in charge of the *Data Ingestion Component*, **MongoDB** is

the core of the *Storage Component*, in other words, it is the database that stores the information and, last but not least, **Power BI** is the *Visualization* software used to build the Dashboard.

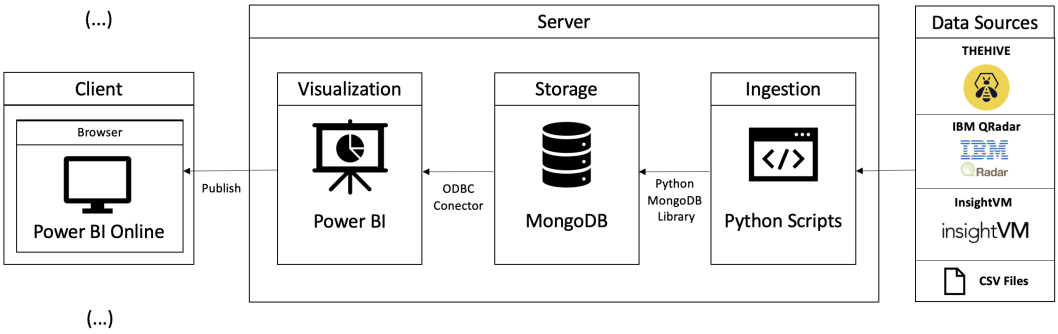


Figure 3.3: System B - Architecture Model

As it has been stated, the process of establishing System B's architectural model was mainly based on the experience gained from the work conducted during the development of System A. System B's architectural model ended up being slightly similar with a key difference in the Storage Layer, where *Elasticsearch* was replaced with *MongoDB*. This change will be explained and justified in the Storage Component section further ahead. Hence, the final architectural model of System B is the following: *Python Scripts* are in charge of data ingestion, *MongoDB* is the database and *Power BI* is the visualization software.

The next subsections introduce the different software tools experimented upon on each component, coupled with the pros and cons of each which, ultimately, led to the establishment of the final architectural model showcased above.

3.3.2 Ingestion Layer

After developing System A it was concluded that through the use of programmatically written scripts, coupled with a system scheduler that would run the scripts in a monthly/daily/hourly fashion the system could be developed in a controlled and flexible manner. Hence, after verifying that System A was functional it was decided that a set of *Python Scripts* would be in charge of System B's Ingestion Component.

3.3.3 Storage Layer

As it has been described, the ingestion layer is composed by the database that is used to store all the relevant information used to enrich the dashboard. As we have seen in System A's Architecture section, an ODBC connetor was necessary to connect the data stored inside of *Elasticsearch* with *Power BI*. Despite the existence of this technology, what was initially unknown to the user was that in order to take advantage of the ODBC Client

it was necessary to subscribe to Elastic's service (which was done for a 30 day trial during System A's development). Hence, since developing a solution to the client that would not be self sustained without having a financial cost was out of the question, the idea of using *Elasticsearch* was dropped.

After researching for other common unstructured databases, an interesting candidate was found: the popular non-relational database for unstructured data *MongoDB*. The reason why searching for another unstructured database was preferred was because of the possibility of changes being made to the schema of data. In other words, so that it would be possible to insert a new field to the existent data mapping, in order to for example, add a new visualization to the dashboard.

The following section aims to introduce *MongoDB* as the database replacing *Elasticsearch*, as well as to clarify some core concepts and advantages it poses over *Elasticsearch*.

3.3.3.1 MongoDB

MongoDB is a document-oriented NoSQL database used for the storage of high volume data. Typical SQL databases, tabulate data in a fixed row and column format, thus storing data in a structured fashion. These types of relational databases access data using SQL (Structured Query Language). In contrast, a NoSQL database does not store data in a tabular fashion. NoSQL databases can store data in four different ways: document, key-value, column stores, and graphs. In the case of *MongoDB*, data is stored in a document oriented style [74].

Now that the basics of MongoDB have been clarified, let us overview some of its core concepts.

- **Documents** - In the same way as *Elasticsearch*, *MongoDB* stores data as JSON documents. Hence, a document can be defined as the basic unit of information, expressed in JSON, indexed in *MongoDB*. Documents can take the form of simple text files or even data objects structured in JSON containing primitive data types such as integers and strings.

- **Collections** - A collection is a collection of documents. Unlike the case of tables in relational databases, a collection in MongoDB does not enforce a schema, being far more flexible.

- **Replica Sets** - In *MongoDB*, databases have at least two more copies of the data associated with. These copies are made upon creation of the database and are entitled as replica sets.

To sum up, both *Elasticsearch* and *MongoDB* could easily fill the role of the Storage Component. Nonetheless, since *Power BI* was the chosen visualization software and the requirements to connect to a data source were an ODBC connection, the database with a free ODBC client was the one that was chosen: *MongoDB*.

As a side note, in the same way Elasticsearch has a Python client called elasticsearch-py, PyMongo is a Python distribution containing tools for working with MongoDB. Through this client, it is possible to automate all collection creation and document insertion operations.

3.3.4 Visualisation Layer

Analogous to the Ingestion Layer, the decision to choose Power BI as the architectural component was made after working with System A. In fact, after developing System A and working with both technologies (Kibana and Power BI) it was concluded that the latter was far superior than the former, due to the reasons stated in System A's overview of the frameworks.

Additionally, due to the fact that the client had access to the *Office 365 Suite*, which includes licensing for Power BI Pro, it was possible to use its much needed premium features such as, more storage space for data sets and the ability to share reports through various members of the organization.

Finally, after verifying that System A was functional it was decided that *Power BI* would be in charge of System B's Visualization Component.

3.4 Conclusions

Over the course of the research phase, several different frameworks were investigated for the sake of aiming to find a suitable set of frameworks the would enable a sustainable solution to be built.

Upon starting the development of System A, it was initially planned to implement it through the support of the ELK Stack. As such, the first iteration of this system's architecture was initially compromised by the software bundle: *Logstash*, *Elasticsearch* and *Kibana*. Later on, first *Logstash* and later on *Kibana*, were respectively substituted by a set of programmatically written scripts and Power BI, another data visualization software. Firstly, *Logstash* was replaced due to an inherent lack of experience working with the framework compared to scripting. Lastly, *Kibana* was replaced, because the former introduced a lot of limitations and obstacles that were easily solved by changing the tool to *Power BI*.

As for System B, overall the architectural model did not suffer many changes but one, the replacement of the Storage Component from *Elasticsearch* to *MongoDB*. This decision came to be for two reasons. Firstly, the accessibility of *Elasticsearch's* ODBC Client was tied to the ELK Stack's premium features, and finally, it would not make sense to implement *Elasticsearch* since most of its design is based on the implementation of this framework along the remaining ELK Stack, or in this case, with *Kibana*.

To sum up, throughout the research and development phases of the work conducted in this dissertation, different software was experimented and trialed upon in order to find

the ones that would best fit the solution being developed. Just because a certain entity developed a framework that seems to align with one's objectives, does not mean that it is the only or best solution to use. Hence, by taking the ELK Stack as a starting point it was possible to build a skeleton of the architecture. Despite of its mutations over the course of time, the same basis was retained showcasing how the solution grew over time in spite of retaining the same foundation.

4

Implementation: System A

This chapter is divided into 4 different sections. The first section(4.1) is used to introduce System A and to explain why this chapter is rather small in size when compared to the one in chapter 5, which describes System B's implementation. Afterwards, the second section(4.2) jumps straight to the different types of metrics that can be consulted in the dashboard. The third section(4.3) briefly introduces the different data sources accessed and that are used to enrich the dashboard visualizations. In the fourth section(4.4) the dashboard is presented to the reader, as well as its different visualizations. Finally, in the last section(4.5), a brief summary of the chapter is performed, as well as a description of what is unique to System A.

4.1 Preamble

As we have seen in the previous chapter, Systems A and B both have different objectives, architectures and developments life cycles. Whereas System B was developed with some prior knowledge and experience due to the time spent working on the first system, System A was developed in a phase where there did not exist any previous experience working on the subject. In fact, System A ended up becoming more of a proof of concept rather than a fully fledged and independent solution, like System B. Hence, it can be stated that System A is not the main system but can be considered one of the stepping stones that led to the development of the main situational awareness dashboard: System B.

Furthermore, it is important to mention that, even if System A left the mock-up stage it would always be a difficult solution to bring to production. Since the majority of System A's metrics are calculated based on information the was gathered from services that require a subscription, this solution would not be ideal. In fact, as it will be showcased ahead, only two out of the four of the threat intelligence feeds were open/free, not to mention how the third party security rating service was also a subscription based service. Finally, the internal data also faced the obstacle that the client had not given access to the credentials required to access its information.

To sum up, the main idea that this section aims to convey is that, since System A

was part of the learning experience, this component should be mentioned and described. Despite this, it should not be given the same attention to detail regarding its implementation process, as it is given to System B. Therefore, this chapter will be much shorter than the one which describes System B's implementation, only focusing on the parts that are unique to it:

- **Metrics** - Both systems have different data to visualize, therefore their metrics also differ. Hence, System A's metrics will be reviewed in the next section.
- **Data sources** - If both systems have different types of metrics, their data sources will inevitably be different as well.
- **Visualizations** - Different systems, different dashboards, different metrics, different visualizations. Hence, this chapter also reviews System A's visualizations and dashboard interface.

4.2 Metrics

This section aims to list the different metrics incorporated in System A's dashboard. Due to a limited access to data and a lack of knowledge and perception into what metrics could be interesting to explore in a SOC context, in this solution the visualizations that ended up being developed were very much dependent on the information that was capable of being accessed. This aims to justify the low number of metrics and the reduced size of the dashboard compared to System B.

4.2.1 Threat Intelligence Metrics

In regards to the Threat Intelligence KPIs, these metrics provides the SOC Staff with the possibility of identifying incoming attacks to the organization that have not been yet found. Furthermore, by providing visibility into the most predominant trends such as, threat actors and categories of attacks the organization can strengthen their preventive security posture by focusing into defending against such threats.

All in all, five different metrics were built:

- **Top 3 Threat Actors** - Measures the most predominant threat actors involved in the attacks related to the indicators of compromise gathered.
- **Top 3 Categories** - Measures the most predominant categories related to the indicators of compromise gathered.
- **Top 3 Targeted Entities** - Measures the most predominant entities which were victims or are in som way related to the indicators of compromise gathered.

- **Percentage of Data from each Source** - Measure the percentage and number of indicators of compromise gathered from each source. Provides perception into the relevance of the data sources when compared to each other.
- **Table of IOCs** - Data table which showcases every indicator of compromise so that viewers can analyze them. It can be filtered by the headers in ascendant or descendant order.

4.2.2 Internal Metrics

Moving on to the Internal Metrics, they were divided into three groups. The first represents KPIs related to machines, their exposure level and status of the Antivirus software:

- **Distribution of Machines by Risk** - Measures the distribution of machines by their risk level. This value is affected by the exposure level and status of the antivirus.
- **Distribution of Machines by Exposure Level** - Measures the distribution of machines by their level of exposure.
- **Number of Machines** - Number of machines being monitored by the application.
- **Disabled Antivirus** - Number of machines with a disabled antivirus.
- **Antivirus Not Reporting** - Number of machines with an antivirus solution which is not reporting.
- **Antivirus Not Updated** - Number of machines with an antivirus solution which is not up to date.
- **Table of Machines** - Data table which showcases every machine and its main attributes. Can be used by analysts to filter data and search for the status of specific machines.

The second refers to users and suspicious activities performed in their accounts:

- **Top Suspicious IP Addresses** - Measures the most predominant suspicious activities from specific IP Addresses.
- **Distribution of Suspicious Activities by Location** - Measures the most predominant suspicious activities from a specific location.
- **Number of User Accounts** - Number of user accounts being monitored by the application.
- **Compromised Accounts** - Number of user accounts which have been compromised.
- **Accounts At Risk** - Number of user accounts at risk of being breached.

- **Accounts At No Risk** - Number of user accounts with no risk associated.
- **Table of User Accounts** - Data table which showcases every user account and its main attributes. Can be used by analysts to filter data and search for the status of specific users.

Finally, the last set of metrics measures different information related to security incidents:

- **Distribution of Security Incidents by Severity** - Measures the number of security incidents by each severity level.
- **Top 3 Incident Categories** - Measures the top 3 most predominant categories of incidents.
- **Number of Incidents** - Calculates the number of security incidents.
- **Open Incidents** - Calculates the number of open security incidents.
- **Resolved Incidents** - Calculates the number of resolved security resolved incidents.
- **True Positives Incidents** - Calculates the number of true-positive security resolved incidents.
- **False Positives Incidents** - Calculates the number of false-positive security resolved incidents.
- **Table of Incidents** - Data table which showcases every security incident and its main attributes. Can be used by analysts to filter data and search for the status of specific incidents.

4.2.3 Risk Metrics

Before introducing the risk metrics it is important to explain how the rating process is performed. Every KRI present in the dashboard is illustrated by meter visualizations which measure risk with a score from 0 to 100. The higher the score, the bigger the risk. Finally, this risk score is divided into 4 different zones: from **0** to **25** risk is considered *Low*, from **25** to **50** risk is *Moderate*, from **50** to **75** it is *High* and from **75** to **100** it is considered *Critical*.

The KRIs can be divided into two sets. Whereas the first corresponds to the low level risk components of the third party security rating, the rest are top level representations of risk, such as, the threat intelligence risk component.

The first set of risk metrics being analyzed are the ones in respect to the Third Party Security Rating. *Security Scorecard* helps organizations understand the cyber health of their ecosystem across 10 risk factor groups. It's important to mention that each one of this values is represented in a security rating, the higher the better. Hence, the values

are mirrored in order to extract a risk factor from them. For instance, a security rating of 75 is translated to a risk factor of 100 - 75 = 25. Finally, the risk factor groups are listed below [53]:

- **Web Application Security** - Represents the the idea of protecting a Web Application's assets from threat actors. Hence, it measures how resilient a web application is to threats such as, Cross-site Scripting (XSS) or SQL injection attacks.

- **Endpoint Security** - This factor aims to measure how well secure and protected are the organization's endpoint devices (laptops, desktops, mobile devices) that access that company's network.

- **Network Security** - Represents the protection of the network infrastructure from attacks from threats, such as, unauthorized access. Hence, the service aims to actively search for existing vulnerabilities, such as "open access points, insecure or misconfigured SSL certificates, or database vulnerabilities and security holes that can stem from the lack of proper security measures".

- **IP Reputation** - The application "ingests millions of malware signals from all over the world. The incoming infected IP addresses are then processed and attributed to corporate enterprises through our IP attribution algorithm. The quantity and duration of malware infections are used as the determining factor for these calculations, providing a data point for the overall assessment of an organization's IP Reputation, along with other assessment techniques."

- **DNS Health** - Security Scorecard "measures multiple DNS configuration settings, such as OpenResolver configurations as well as the presence of recommended configurations such as DNSSEC, SPF, DKIM, and DMARC."

- **Cubit** - The Cubit Score is a Security Scorecard proprietary threat indicator. It "measures a collection of critical security and configuration issues related to exposed administrative portals".

- **Information Leak** - The application "identifies all sensitive information that is exposed as part of a data breach or leak, keylogger dumps, pastebin dumps, database dumps, and via other information repositories. SecurityScorecard maps the information back to the companies who own the data or associated email accounts that are connected to the leaked information, assessing the likelihood that an organization will succumb to a security incident due to the leaked information."

- **Hacker Chatter** - "Continuously collects communications from multiple streams of underground chatter, including hard-to-access or private hacker forums. Organizations and IPs that are discussed or targeted are identified."

- **Patching Cadence** - The application also measures how "diligently a company is at patching its operating systems, services, applications, software, and hardware in a timely manner".

- **Social Engineering** - The service also tracks if "employees are using their corporate account information for services, for example, social networks, service accounts, personal finance accounts, and marketing lists that can be exploited."

All in all, Security Scorecard measures the ten risk factors listed above. Hence, those values are extracted for the case of System A's organization and incorporated into the global risk score.

Below is a list of the remaining KRIs integrated in System A's dashboard:

- **Third-Party Risk Rating** - Measures the aggregated value of the 10 risk factors listed above. This value is also extracted from *Security Scorecard*.

- **Machines Risk** - One of the three risk metrics of the internal component. This KRI performs an average of the risk associated with every machine.

- **User Accounts Risk** - One of the three risk metrics of the internal component. This KRI performs an average of the risk associated with every user.

- **Security Incidents Risk** - One of the three risk metrics of the internal component. This KRI performs an average of the risk associated with every unresolved security incident.

- **Internal Risk** - Measures the aggregated value of the three KRIs listed above which are part of the internal risk component. This value is calculated means of an average between all three values.

- **Threat Intelligence Risk** - Measures the risk associated with the threat intelligence component of the solution. Each indicator of compromise (IOC) has a risk score associated with it. The KRI is calculated through averaging the risk of every IOC.

- **Global Risk Index** - Measures the overall risk of system A. The value of the KRI is generated by means of an average between the Threat Intelligence Component, the Internal Component and the Third-Party Risk.

To sum up, all the different KPIs and KRIs have been described above. Hence, the next chapter cover how the data ingestion process is conducted. In addition, it also explains how the risk of each security incident, user account, machine and indicator of compromise is calculated.

4.3 Data Sources

Moving on to the topic of data sources, the list below points out each type of data source incorporated in the system:

1. **Threat Intelligence Feeds** - Five different sources were used to extract information regarding potential threat indicators: Feed de Segurança Informática, IBM X-Force Exchange, OTX Alienvault and a TAXII Server.

2. **Internal Data** - Regarding the information collected towards building the visualizations in respect to the internal components of the organization, since the client did not provide access to this information, for testing purposes the information and extracted in CSV format and loaded into the database.

3. **Third-Party Security Rating** - Finally, the Third-Party Security Rating that was used to further enrich the dashboard was from "Security Scorecard". Unlike some of the Threat Intelligence Feeds there was no mechanism for accessing the scores through a RESTful API. Hence, the information was also extracted in CSV format and loaded into the database.

All in all, data is extracted through a set of different data sources by exporting data from various CVS files, accessing endpoints in RESTful APIs and connecting to a TAXII Server to access STIX formatted data. The next section will explain how the process of extracting relevant Threat Intelligence occurs.

4.3.1 Threat Intelligence

As it has been stated, there are four different Threat Intelligence sources used in System A. These four sources were segregated into two different types: STIX formatted data and Non-STIX formatted data. Whereas the former represents STIX data gathered from TAXII servers, the latter corresponds to indicators of compromise gathered from various applications.

Overall, the data mapping of each indicator of compromise after it has been extracted, parsed and transformed is:

- **id: String** - Unique identifier associated with each IOC.
- **timestamp: Date** - Date where the IOC was first detected.
- **indicator: String** - The Indicator of Compromise (IOC).
- **source: String** - Name of the data source where the IOC was extracted from.
- **type: String** - Type of the IOC (URL/IPv4/IPv6).
- **category: String** - Category associated with the IOC (Malware/Phishing).

- **status: Boolean** - Status of the IOC (Active/Inactive).
- **target: String** - Name of the target entity associated with the IOC.
- **malware_type: String** - The type of the malware
- **malware: String** - The name of the malware.
- **threat_actor: String** - The threat actor who orchestrated the attack.
- **likelihood_qual: String** - Qualitative value associated with the likelihood of occurrence of the organization falling victim to the IOC (Guaranteed/High/Moderate/Low).
- **likelihood_quant: Double** - Quantitative value associated with the likelihood of occurrence of the organization falling victim to the IOC (1.0/0.75/0.5/0.25).
- **impact_qual: String** - Qualitative value associated with the impact of the IOC in case the organization falls victim to it (Critical/High/Moderate/Low).
- **impact_quant: Integer** - Quantitative value associated with the impact of the IOC in case the organization falls victim to it (100/75/50/25).
- **risk_qual: String** - Qualitative value of the risk associated with the IOC (Critical/High/Moderate/Low).
- **risk_quant: Integer** - Qualitative value of the risk associated with the IOC (0-100).

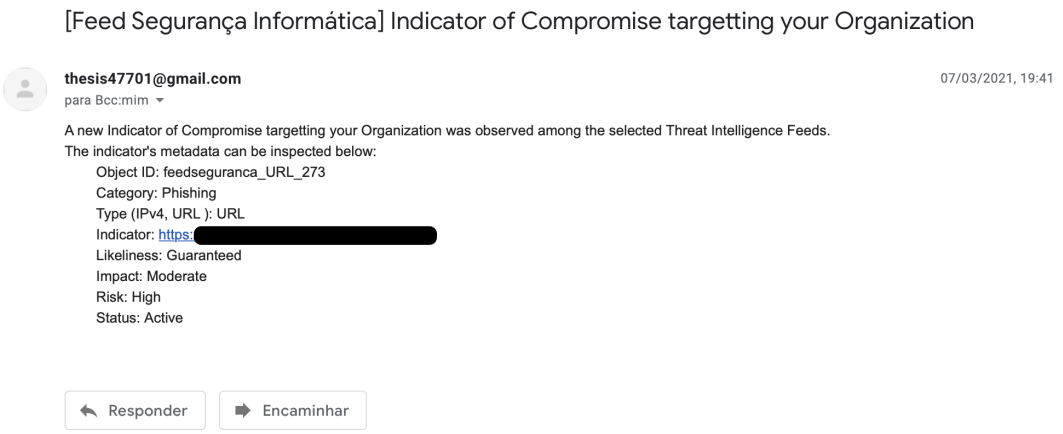
Of note, it is also important to mention that for every Indicator of Compromise which was verified that had a **Guaranteed** likelihood or **Critical** risk, an alerting mechanism was built that would automatically send an email to any designated email inbox. This feature was built as a component of the scripts and is triggered during the data ingestion process. Figure 4.1 showcases an example of such an email containing an IOC targeting Client A.

Now that the data mapping of each indicator of compromise has been listed, the following sections aim to describe each data source's endpoints, as well as how the likelihood, impact and risk values were calculated for each.

4.3.1.1 Non-STIX formatted Sources

The data ingestion process is performed through three different non-STIX formatted data sources, those being: Feed Segurança Informática, IBM X-Force Exchange and OTX Alienvault. This section aims to describe each data source.

First and foremost, *Feed de Segurança Informática* is a threat intelligence feed which compiles phishing and malware indicators of compromise targeting Portuguese entities. Hence, since the client behind the development of this system was a Portuguese financial institution, this data source was ideal. This feed provides a free RESTful API which users


[Feed Segurança Informática] Indicator of Compromise targetting your Organization

thesis47701@gmail.com 07/03/2021, 19:41
para Bcc:mim

A new Indicator of Compromise targetting your Organization was observed among the selected Threat Intelligence Feeds.
The indicator's metadata can be inspected below:
Object ID: feedseguranca_URL_273
Category: Phishing
Type (IPv4, URL): URL
Indicator: https:[redacted]
Likeliness: Guaranteed
Impact: Moderate
Risk: High
Status: Active

Responder Encaminhar

Figure 4.1: Example of an email triggered by an alert containing information regarding an IOC which targeted Client A.

can access by submitting a request for an API token. The primary endpoint which was used is described in Table 4.1.

HTTP Method	URI	Description
GET	https://feed.seguranca-informatica.pt/api.php? token_u={token}& from=last-year& format=json	Returns a list of IOCs from the last year in JSON

Table 4.1: Feed de Segurança Informática Endpoint

The second data source being described is *IBM X-Force Exchange*. This platform is "a cloud-based threat intelligence platform that allows you to consume, share and act on threat intelligence. It enables you to rapidly research the latest global security threats, aggregate actionable intelligence, consult with experts and collaborate with peers"[24]. This feed provides a paid (free for one month) RESTful API which users can access by creating an account and generating an API Token. The primary endpoints which were accessed are described in Table 4.2.

HTTP Method	URI	Description
GET	/xtfi/mw/url	Returns a list of URLs related with malicious or malware hosting websites
GET	/xtfi/phishing/url	Returns a list of URLs related to phishing websites
GET	/xtfi/mw/ipv4	Returns a list of IPv4 addresses related with malicious or malware hosting websites
GET	/xtfi/mw/ipv6	Returns a list of IPv6 addresses related with malicious or malware hosting websites

Table 4.2: IBM X-Force Exchange Endpoints

The final Non-STIX formatted data source is *OTX Alienvault*. It is a "threat data platform that allows security researchers and threat data producers to share research and investigate new threats"[47]. Users can access threat intelligence data, compromised by IOCs, through subscribing to **Pulses**. Pulses are defined as "collections of indicators of compromise (IoCs), reported by the OTX community, which other community members review and comment on"[47]. This feed also provides a paid (free for one month) RESTful API which users can access by creating an account and generating an API Token. This way users can extract the IOCs related to the pulses they are subscribed to. The user subscribed to one pulse which was used to ingest IOCs. This pulse was from **PhishTank** and contained a dynamic list of verified/online banking phishing URLs. Finally, the primary endpoint used to access IOCs contained on their associated pulses is described in Table 4.3.

HTTP Method	URI	Description
GET	/pulses/{pulse_id}	Returns a list of IOCs inside the associated pulse

Table 4.3: OTX Alienvault Endpoints

4.3.1.2 TAXII Servers: STIX 2.0

Moving on to the Threat Intelligence sources in STIX format. As it was expressed in the chapter covering the State of the Art, STIX can be a rich format since it has many data objects which can heavily enrich Indicators of Compromise. Nonetheless, it ends up becoming a complex data model since keeping track of every data node connected to the original is a time consuming task. Therefore, to simplify the solution, only data objects which were of the type IOC and two other data objects which have relationships with it (the *Malware* and the *Threat Actor* objects) were searched for. Unfortunately, both the process of searching for free TAXII servers and having those servers include these type of data objects was a very difficult task to accomplish. Only one server was found, developed by *MITRE* [4], and only a few data objects of the type Indicator of Compromise were found during the short time frame where the solution was used to ingest data. Hence, not a lot of relevant data in STIX format was able to be extracted. Not because of the format, which has inherently tremendous potential, but because of the difficulties encountered searching for open TAXII servers.

4.3.1.3 Risk Assessment

Now that every TI feed has been described, it is important to detail how the process of calculating risk was performed. According to NIST's "Guide to Conducting Risk Assessments"[43], the risk factor can be calculated from the product of the likelihood of an incident occurring, times its impact. Hence, in order to measure the risk of an IOC these two factors had to be measured.

Regarding an IOC's likelihood, the following though process was undertaken to measure this component.

- **Guaranteed/1.0** - If the IOC has been sighted targeting the client's organization.
- **High/0.75** - If the IOC has been sighted targeting a Portuguese financial institution.
- **Moderate/0.5** - If the IOC has been sighted targeting a Portuguese institution.
- **Low/0.25** - If the IOC has been sighted targeting a financial institution.

Measuring this value was a difficult task. Some sources which provided information regarding the targeted entity made it easier to measure than others, like IBM X-Force Exchange which did not provide any information regarding the targeted entity. Hence, in case no information was provided two verifications were performed. In case the IOC was a URL, its HTML was scraped. Afterwards, both the HTML content and the URL were checked for the existence of matching keywords. These keywords were a list of Portuguese financial institutions. On the other hand, if the IOC was an IP it was verified if this IP address belonged to the sub net of the client's digital footprint.

Regarding an IOC's impact, an additional tool was used in order to measure the impact of any given indicator. IBM X-Force has a database of IOCs which have been score using their own risk scoring methodology methodology. This value ranges from 1 to 10, 1 having no risk associated with it and 10 the highest risk. Hence, since there was no likelihood associated with this index, it was considered as the measure which calculates the impact the IOCs in our solution. Therefore, every IOC that is ingested in our solution is ran through IBM X-Force's API. The following methodology was used to calculate the impact of each IOC according to IBM X-Forces risk score:

- **IBM Risk > 8** - The IOC has critical impact, in other words, an impact score of 100.
- **8 > IBM Risk > 5** - The IOC has high impact, in other words, an impact score of 75.
- **5 > IBM Risk > 3** - The IOC has moderate impact, in other words, an impact score of 50.
- **IBM Risk < 3** - The IOC has low impact, in other words, an impact score of 25.

Now that the process to calculate each risk component has been described, the following risk matrix (Figure 4.2) was built, which crosses each likelihood with impact level calculating each final risk score:

As it can be seen in the risk matrix, there are four different levels of risk. The list below describes each one, as well as how each should be interpreted:

- **Critical** - The indicator represents a devastating threat that is highly likely to target or may already have targeted the client's organization. It case this threat has not been dealt with already, it should be done with immediate priority.

Risk Assessment	Impact			
Likelihood	Low (25)	Moderate (50)	High (75)	Critical (100)
Guaranteed (1.0)	Moderate (25)	High (50)	Critical (75)	Critical (100)
High (0.75)	Low (19)	Moderate (38)	High (57)	Critical (75)
Moderate (0.5)	Low (13)	Moderate (25)	Moderate (38)	High (50)
Low (0.25)	Low (6)	Low (13)	Low (19)	Moderate (25)

Figure 4.2: Risk Matrix: Indicators of Compromise

- **High** - The indicator represents a dangerous threat that has a high to guaranteed chance of attacking our organization. Therefore, it is highly advised to take preventive actions against this threat as soon as possible.
- **Moderate** - The indicator could cause moderate damage to the organization. Therefore, it has a moderate priority of resolution.
- **Low** - The indicator does not appear to have the potential to cause significant damage or the probability of targeting the organizations. Therefore, it has a low priority of being taken preventive measures against it.

4.3.2 Internal

Moving on to the internal data ingestion process, it can be divided into three data sets. The first accumulates data regarding the organization's machines AV status. The second information regarding suspicious activities in users accounts and categorizes each user account according to its risk status. The last data set has information regarding security incidents. Finally, in order to simplify this section, the data mapping of each source will be skipped, concentrating its focus on how risk was measured for each one. Unlike the risk calculated in the previous section, the likelihood times impact formula is not applied in this situation. The overall process of calculating risk is based on the AV status of the machines, if the user account is at risk of being breached and the severity of security incidents.

As it was mentioned at an earlier stage of the chapter, since the client did not end up providing access to the APIs of each tool that stores this information, this information had to be extracted manually in CSV format.

4.3.2.1 Machines' Anti-Virus Status

On the topic of each machine's AV status, this information was extracted from one the organization's security tools: Microsoft Defender for Endpoint. One of the features that this tool provides on top of its the ability to detect the status of a machine's AV, is a qualitative risk level. Hence, the list below translates each risk level of the tool to our risk level:

- **High** - It was considered that the risk in our solution was **100** or **Critical**
- **Medium** - It was considered that the risk in our solution was **75** or **High**.
- **Low** - It was considered that the risk in our solution was **50** or **Moderate**.
- **Informational** - It was considered that the risk in our solution was **25** or **Low**.

4.3.2.2 User Access Control

Moving on to the topic of each user account activities, this information was extracted from one the organization's security tools: Azure Active Directory Entity Protection. One of the features that this tool provides is the ability to measure each user account's risk level as well as identifying if it has administrative privileges. These were the two core entries that helped calculating our risk score.

- **Confirmed Compromised** - It was considered that the risk in our solution was **100/Critical** if the user had administrative privileges and **85/Critical** if not.
- **High** - It was considered that the risk in our solution was **75/Critical** if the user had administrative privileges and **60/High** if not.
- **Medium** - It was considered that the risk in our solution was **50/High** if the user had administrative privileges and **35/Moderate** if not.
- **Low** - It was considered that the risk in our solution was **25/Moderate** if the user had administrative privileges and **10/Low** if not.

4.3.2.3 Security Incidents

Last but not least, on the topic of each security incidents, this information was extracted from one the organization's incidents ticketing platform: BCM Remedy. One of the features that this tool provides is the ability to categorize each security incident by severity. This was the primary indicator used to measure risk. It is important to note that incidents that have been resolved or are considered false-positives are not considered for the calculation of the dashboards risk indicator. Coincidentally, Remedy categorizes incidents in four types, the same that are considered in System A: Critical, High, Medium and Low. Therefore, these categories remained the same, changing the keyword "Medium"to "Moderate"and associating the scores 100, 75, 50 and 25 with each one.

4.3.3 Third-Party Security Rating

The "Security Scorecard"service enables users who are not subscribed to the service, to create an account and provides visibility into the organization's security rating which is associated with the email's domain. Hence, after creating an account the user had access to the organization's security rating and its ten risk factor groups ratings.

Furthermore, the "Security Scorecard"service provides a RESTful API that enables users to export data regarding an institution's security rating. Despite the existence of this feature, it is lodged behind a subscription service. Hence, there was no possibility of using it and incorporating it in the data ingestion process through scripts. Nonetheless, this services provides users with the possibility of exporting data in CSV format which was the route that was taken. Below is the data mapping of the CSV data that was exported from the application:

1. **Timestamp: Date** - Indicates the date of the security score which is calculated on a daily basis.

2. **Score: Integer** - The overall security rating which takes into account the 10 different risk factor groups.

3. **Each Risk Factor Group Rating: Integer** - This entry represents the ten different risk factor group ratings that are measured by the service.

Each security rating is a score from 0 to 100. The higher the security rating, the better. Hence, in order to extract a risk factor that would align with the scale implemented in the dashboard, each rating was mirrored. In other words, if a certain component has a security score of 65, then the risk of said factor is 100 - 65 = 35. These mirrored ratings, which are now translated into risk, are the ones that are able of being visualized in the situational awareness dashboard and that contribute to the global security risk index.

To sum up, despite the fact that "Security Scorecard"does not provide a free RESTful API for automated data extraction, it provides a way of manually exporting security ratings in CSV format. Furthermore, each set of security ratings was translated into a risk score, so that the SOC Staff could, more intuitively, have a perception onto risk through the use of the developed situational awareness dashboard.

4.4 Visualizations

Before introducing with slide of the dashboard as well as its corresponding visualizations, it is important to mention that the data that can be visualized in the dashboard is real. During the month of December the data ingestion tools were ran every day filling the database with threat intelligence. Regarding the internal component and third-party rating service, the data that enriches the visualizations of each slide had to be manually exported in CSV format from their respective data sources.

The visualizations that constitute System A's Situational Awareness Dashboard are divided into four different categories/tabs the user can explore:

1. **Overview** - Top level perspective of the dashboard. Provides a brief overview of the most important metrics and risk scores of each component.

2. **Threat Intelligence** - Represents the information gathered from the different Threat Intelligence sources. In fact, the viewer can explore every indicator of compromise ingested in this slide as well as other information represented in other KPIs.

3. **Internal** - Represents three different types of information gathered from security tools employed by the SOC of the organization. Provides the viewer with the ability to explore the status of every user account's suspicious behaviours, machines' AV status and information regarding security incidents.

4. **Third-Party Rating** - This final slide, gives the user a quick perception of the security risk of the organization which was extracted from the *Security Scorecard* service. Provides perception into 10 different risk factors and how each evolves over time.

4.4.1 Overview

As it has been stated, the first slide of the dashboard gathers essential metrics of each component of the dashboard. Figure I.1 illustrates this panel.

First and foremost, four meters can be visualized at the left side of the slide, each representing a different risk score. Furthermore, at the right of each meter the viewer can visualize how the risk score of the KRI to its left evolves over time.

At the right side of this slide, another set of visualizations can be consulted. The three boxes at the top have metrics regarding the number of machines with different AV statuses, the number of user accounts at risk and with no risk associated and other metrics regarding the resolution status of security incidents. The three remaining KPIs provide the viewer with useful metrics regarding the indicators of compromise gathered from the various threat intelligence sources. In fact, they provide information into who the top 3 threat actors, categories and entities related with the indicators of compromise.

Finally, all the different KPIs can be filtered through the time slicer in the upper right corner of the screen. All the metrics listed above are described in the "Metrics"section of the current chapter.

4.4.2 Threat Intelligence

Moving ahead to the second panel, viewers can grasp different KPIs related to the IOCs gathered from the different Threat Intelligence Feeds.

First and foremost, there is the KRI that measure the risk score of this component, followed by three visualizations that are also present on the "Overview"panel: the top 3

threat actors, categories and targeted entities. Furthermore, there is KPI related to the percentage of data gathered from each Threat Intelligence Feed.

At the bottom of the screen there is a data table with information related to every IOC, such as: its date of collection, type, category, status, quantitative risk value, the indicator of compromise, malware name (if applicable), malware type (if applicable), target entity and threat actor behind the attack. All the different metrics can be filtered through the time slicer in the upper right corner of the screen.

Finally, Figure I.2 illustrates this panel.

4.4.3 Internal

This panel can be divided into three rows of metrics, each related to a different internal component.

The first row incorporates metrics related to the security status of the different endpoints. From the left to the right, the first KRI is a meter that measures the current risk value of the organization's machines. It is followed by two piecharts. Whereas the first provides an overview of the distribution of risk per endpoint, the second gives perception to the how many endpoints exists for each exposure level. Furthermore, four different metrics are displayed, that measure the total total of machines, which have its AV disabled, not reporting and not up to date. Finally, the data table to the right side gives information regarding each machine, such as: its unique identifier, qualitative risk value (priority of treatment), exposure level, quantitative risk value, domain, date where it was last seen reporting, antivirus disabled status, antivirus not reporting status and antivirus not updated status.

Moving on to the second row, the first KRI measures the current risk value of the organization's user accounts. This meter is followed by a bar chart which counts the number of suspicious IP Addresses related to suspicious login activities. Additionally, a geographical map illustrates the location of each access performed by those suspicious IP Addresses. Furthermore, four different metrics are displayed, that measure the total number of user accounts, which of those are compromised, are at risk and have no risk associated. Finally, the data table to the right side provides information regarding each user account, such as: the email address associated with it, a qualitative risk value (priority of treatment), status of the account (Compromised/At risk/No risk), quantitative risk value, the IP address and respective country associated with the suspicious activity and the status of the user account (Active/Deleted).

Lastly, the third row provides metrics related to security incidents. The first KRI is a meter that measures the current risk value of this component. It is followed by a bar chart that calculates the number of security incidents by their severity. Afterwards, another bar chart demonstrates the top three incident categories. Furthermore, to the right of the bar chart four different metrics are displayed which measure the number of security incidents, which of those remain open/resolved and the number of true positive and false

positive incidents registered. All these different metrics can be filtered through the time slicer in the upper right corner of the screen.

Finally, Figure I.3 illustrates this panel.

4.4.4 Third-Party Rating

Last but not least, there is the Third Party Rating which measures risk, according to a Third Party Security Rating service. As it has been stated in a previous section, the risk rating is divided into ten different factor groups. Hence, this panel illustrates this information through a set of eleven different meters: one fo the global third party rating and the remaining ten for each risk factor group. Furthermore, two line graphs illustrate how the Third Party Risk Rating, as well as the score of its components, evolve over time. All these different metrics can be filtered through the time slicer in the upper right corner of the screen. Finally, Figure I.4 illustrates this panel.

4.5 Conclusion

To sum up the contents of this chapter, it aimed to describe every step of the implementation process of System A, from detailing every data source to describing each metric and its respective visualization.

The development phase of System A refers to the first of two stages that took place during the time working in this thesis. Unlike System B, which will be described in the next section, this solution did not leave the testing phase mainly because the Contractor Agreement between EY and the Client reached an end. Nonetheless, the system was implemented in a testing environment which is described in this chapter.

Despite the fact that this solution never left testing, it ended up fulfilling its purpose which was of letting the student gain experience in the SOC of an institution, designing interesting metrics to explore, researching risk assessment methodologies, mounting a system architecture and unifying all of the different components to build an independent system.

5

Implementation: System B

This chapter is divided into four different sections. The first section(5.1), contains an overview of how System B was setup, in other words, the environment it is lodged in, how each component was built and the way each one is connected to each other. Afterwards, the second section(5.2) introduces every metric and how it is calculated, whereas the third section(5.3) aims to describe each one of the data sources that the system uses to enrich the dashboard. In the fourth section(5.4), every page of the dashboard is presented in detail, with a description of every action and filtering option available to the user. Finally, in the last section(5.5), a brief summary of the chapter is performed, as well as a description of what is unique to System B.

5.1 System Setup

As we have seen in Chapter 3, the architecture surrounding this system is centralized around a server. This server contains all the 3 core components of the solution. Hence, it is in charge with:

- Extracting, parsing and transforming data
- Lodging the database that stores all the relevant information
- Publishing the Power BI Reports and their corresponding data set

The server that was provisioned by the client was running Windows 10, which was an important requirement as *Power BI* is exclusive to Microsoft devices. Afterwards, there was a setup phase which covered all the installation and configuration tasks that needed to be performed in order to start development. Below is a list of all the steps that were undertaken during this phase:

1. Python + PyMongo library, MongoDB-as-a-Service and Power BI were installed.
2. Installed the *MongoDB ODBC Driver for BI Connector* and created a System Data Source Name.

3. Installed and configured the *on-premises data gateway* developed by Microsoft.

The first step is self explanatory, the installation of Python, MongoDB and Power BI refer to each component of the system architecture. Additionally, by installing the PyMongo client, it is possible to perform operations on the database through Python.

After setting up the core components, in order to be able to connect the database with the visualization software, it is necessary to install the ODBC Driver for MongoDB and create a System DSN. According to Microsoft, a DSN "is the name that applications use to request a connection to an ODBC Data Source. In other words, it is a symbolic name that represents the ODBC connection. It stores the connection details like database name, directory, database driver, UserID, password, etc. when making a connection to the ODBC"[72]. Through Figure 5.1, one can see that after installing the MongoDB ODBC Driver it is possible to create a System DSN. Configuring the DSN requires at least a name, the IP address of the server where MongoDB is running, the port number of the specific process and the name of the database being connected to.

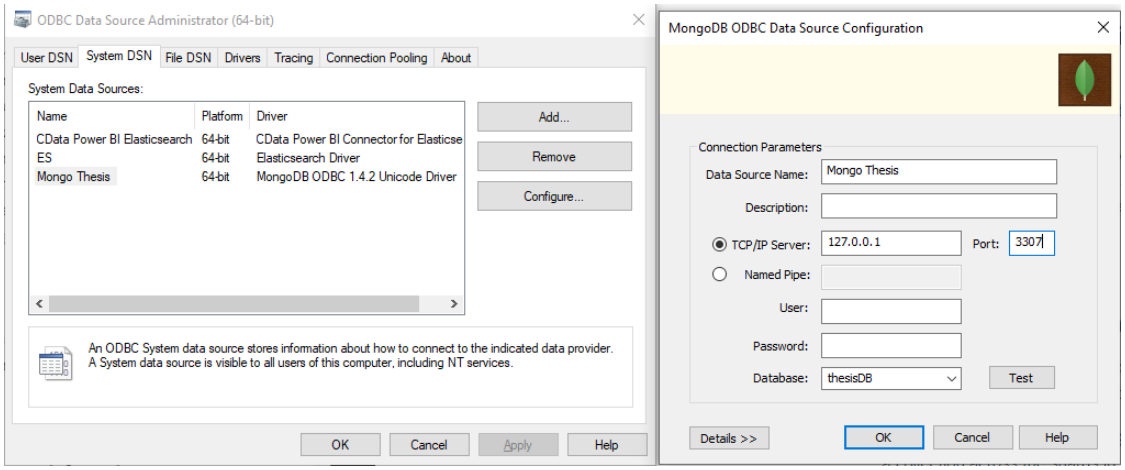


Figure 5.1: Setup of MongoDB's System DSN

Since the machine where MongoDB is running is the same machine of where the DSN is being created, the address should be **127.0.0.1**. This address is the loopback Internet protocol, used to establish an IP connection to the same machine. In regards to the port number, MongoDB's documentation states that the process is running on port number **3307**. The remaining fields, Data Source Name and Database, are self explanatory.

The final step performed during the setup of the system was to install and configure the Microsoft on-premises data gateway. This gateway allow users to keep their reports up to date by connecting them to on-premise data sources without the need to manually move one's data.

Now that the system setup has been described, the next section will list and catogrize all the metrics that were incorporated in the solution.

5.2 Metrics

Before tackling each metric it is important to point out that, during the development of the solution, several metrics were not able to be delivered with rich visualizations due to a missing data component that the client could not fulfill. In other words, several KPIs that were proposed to be constructed did not have the data to support building its visualizations. Nonetheless, the system was setup so that the visualizations associated with the KPIs with no available data, were prepared to display actionable intelligence when the appropriate data was injected into the database. Doing this, required to build a temporary mapping for that data. This way, if one day the client would have access to the data necessary to build the KPIs, they could do so. By doing this, it facilitates future work that would have to be performed on the solution, like changing the document schema, even though the scripts will always end up having to be adjusted.

Additionally, prior to the involvement of the student in the implementation of the Situational Awareness Solution, the list of metrics to be displayed in the dashboard had already been discussed and settled between EY and the client. In fact, this same set of KPIs can be segregated into four different types: **Financial**, **Incident**, **Alerts and Vulnerabilities** and **Other Metrics** with an additional group of KRIs or **Risk Metrics**.

Each KPI has a unique identifier, description, a type, frequency of collection, target audience and formula of calculation. In fact, it is important to highlight that not every KPI is present in every Dashboard, for example the *Operational* dashboard, monitored by the SOC Staff, does not contain any information regarding financial metrics, nor do they contribute for the global risk index. Nonetheless, in order to clarify the subject, the following sections will present tables that list every metric, along with the properties mentioned above. On a final note, it was decided that no DAX formulas would be showcased in the document. Explaining every DAX formula of every measure, would lead to a deep level of detail which would worsen the clarity of the document. Explaining a DAX formula implies explaining every function used by DAX, not to mention since some data is calculated through auxiliary tables those tables would have to be described as well. Hence, the formulas of calculation will be presented in a simplified manner.

5.2.1 Financial Metrics

The first set of metrics being covered are the financial metrics. These metrics help organizations successfully manage their budget and resource allocation. Successfully keeping track of these key indicators is important, in order to achieve your goals and grow your business.

Figure 5.2 illustrates the set of financial metrics that were incorporated in system B. At first sight, one can quickly understand that none of the financial metrics had their visualizations finalized. The reason behind this was briefly referenced in the beginning of this section. The client did not provide or have access to the necessary information to

complete the KPIs.

	Financial Metrics					
#ID	Name	Description	Type	Frequency of Collection	Target Audience	Formula
KPI.FIN.01	Incident Financial Estimation	Estimate the hour/person cost of emloyees involved in the incident response times the number of hours spent on the resolution	Governance	Daily	Top Management/CISO	(Hour Rate Analist * Number of hours needed to resolve the security incident)
KPI.FIN.02	Financial and Reputacional incidents	Number of security incidents causing financial loss, business disruption or public embarrassment	Governance	Daily	Top Management/CISO	(Number of Financial or Reputational Incidents / Total number of incidentes) * 100
KPI.FIN.03	Security Team Expenses	Comparison of expenses / investments incurred by the security team compared to the team's budget	Governance	Daily	Top Management/CISO	-

Figure 5.2: List of Financial Metrics (FIN)

- **KPI.FIN.01 (Incident Financial Estimation)** - Despite having access to the amount of hours needed to resolve an incident, the hour rate per analyst was not provided by the client. Thus, the visualization was not completed. By default, the hourly rate that was placed for each analyst was 0 so that the visualization could appear empty.

- **KPI.FIN.02 (Financial and Reputational Incidents)** - The lack of a field or flag that could distinguish incidents with financial or reputational loss against normal incidents, made this KPI also not possible to finalize. Nonetheless, the field "Efeitos"was added to the "Incidentes"collection so that in the future it could be used to complete the visualization.

- **KPI.FIN.03 (Security Team Expenses)** - The client never provided a list of the security team's expenses and budget, therefore the KPI's visualization could not be finalized.

Finally, as previously mentioned, the target audience of the financial metrics is compromised by the Top Management and CISO dashboards, not including the Operational one.

5.2.2 Incident Metrics

Moving to Incident Metrics, these metrics give insight about how the SOC is performing. This is important because it helps businesses determine the efficiency of their security practices and whether they're meeting specific goals.

Figure 5.3 illustrates the set of incident metrics that were incorporated in system B. Contrary to the previous section, due to the fact that the client was able to provide rich sources of information about security incidents, only four metrics out of the fourteen KPIs could not be completed.

	Incident Metrics					
#ID	Name	Description	Type	Frequency of collection	Target Audience	Formula
KPI.IHR.01	Reactive Counter-Measures	Percentage of incidents with containment measures applied	Technical	Daily	Top Management/CISO/Operational	(Number of incidents with counter measures in place/ Total number of incidents) * 100
KPI.IHR.02	Proactive Counter-Measures	Percentage of prevention measures implemented without any associated incidents	Technical	Daily	Top Management/CISO/Operational	(Number of proactive measures not implemented in any security incident/Number of proactive measures) * 100
KPI.IHR.03	Time to Confirmation	Average amount of time since a security incidente is opened until it is classified as a true positive	Technical	Daily	Top Management/CISO/Operational	(Summation of the differences between the date the incident was opened to the date it was confirmed as a true positive / Number of true positive incidents)
KPI.IHR.04	Time to Containment	Average amount of time since a security incident is confirmed as a true positive until it is classified as a true positive	Technical	Daily	Top Management/CISO/Operational	(Summation of the differences between the date the incident was opened to the date it was contained / Number of true positive incidents)
KPI.IHR.05	Time to Closure	Average amount of time since a security incidente is confirmed as a true positive till until it is closed	Technical	Daily	Top Management/CISO/Operational	(Summation of the differences between the date the incident was confirmed to the date it was closed / Number of true positive incidents)
KPI.IHR.06	Confirmed Incident Matrix	Numerical matrix which measures how the number of confirmed incidents has evolved over time	Technical	Daily	Top Management/CISO/Operational	Counter of the number of confirmed incidentes for the current and previous months
KPI.IHR.07	False Positive Incidents Matrix	Numerical matrix which measures how the number of false positive incidents has evolved over time	Technical	Daily	Top Management/CISO/Operational	Counter of the number of false positive incidentss for the current and previous months
KPI.IHR.08	Unscheduled Downtime	Total unscheduled downtime of each system or machine	Technical	Daily	Top Management/CISO/Operational	Summation of the total amount of unscheduled downtime per machine
KPI.IHR.09	Critical Assets Integrated in the	Percentage of Critical Assets Integrated in the SIEM	General	Daily	Top Management/CISO/Operational	(Number of critical assets integrated in the SIEM / Number of critical assets) * 100
KPI.IHR.10	Table of Critical Assets Integrated in the SIEM	Table that verifies if a critical asset is integrated in the SIEM	General	Daily	Top Management/CISO/Operational	Verifies if the asset is integrated in the SIEM
KPI.IHR.11	Repeated Incidents	Number of machines with the same type of OS struck by a security incident	Technical	Daily	Top Management/CISO/Operational	Counter of the number of machines with the same type of OS struck by security incidentes
KPI.IHR.12	Escalated Incidents	Number of incidents that were escalated to the SOC Manager	Technical	Daily	Top Management/CISO/Operational	(Number of escalated incidentes / Total number of incidents) * 100
KPI.IHR.13	Unresolved Incidentes since last extraction	Total number of unresolved incidents since the last extraction	Technical	Daily	Top Management/CISO/Operational	Number of unresolved incidents since last extraction / Total number of incidents since last extraction
KPI.IHR.14	Unresolved Incidentes	Total number of unresolved incidents	Technical	Daily	Top Management/CISO/Operational	Number of unresolved incidents / Total number of incidents

Figure 5.3: List of Incident Handling and Response Metrics (IHR)

- **KPI.IHR.02 (Reactive Counter-Measures)** - This metric could not be calculated due to a lack of information regarding incident proactive measures.

- **KPI.IHR.08 (Unscheduled Downtime)** - There was no type of security incident that would fit this category. Hence, there was an inability to access information regarding the duration of an unscheduled downtime on the organization's machines.

- **KPI.IHR.11 (Repeated Incidents)** - Since the security incident data schema does not pass the information regarding the OS of the affected machine (if it applies to the type of incident that occurred), it is not possible to finalize the visualization.

- **KPI.IHR.12 (Escalated Incidents)** - The same goes for the KPI that measures the number of incidents escalated to the SOC Manager. Due to the absence of a flag that would indicate if a given incident has been escalated for further investigation, it is not possible to calculate this KPI.

5.2.3 Alerts and Vulnerabilities Metrics

Regarding Alerts and Vulnerabilities, these metrics further improve visibility into how the SOC is performing. Whereas alerts have a bigger focus on the SIEM's performance and checking if rules are triggering alerts properly, vulnerabilities metrics are vital for enforcing protection to prevent the possibility of being attacked in the future.

Figure 5.4 illustrates the set of alert and vulnerability metrics that were incorporated in system B. Once again, the client was capable of providing rich sources of information about alerts and vulnerabilities. Hence why almost all of them were able to be finalized.

Alerts and Vulnerabilities Metrics						
#ID	Name	Description	Type	Frequency of Collection	Target Audience	Formula
KPI.VUL.01	Vulnerability Distribution	Vulnerability distribution per asset group	Technical	Daily	Top Management/CISO/Operational	Number of vulnerabilities per asset group
KPI.ALR.01	Alerts investigated per analyst	Number of days na analyst needs to investigate na alert	People	Daily	Top Management/CISO/Operational	Summation of the number of days necessary to close every incident assigned to a specific analyst / Number of alerts closed by the analyst
KPI.ALR.02	Escalated Alerts	Percentage of escalated alerts	Technical	Daily	Top Management/CISO/Operational	(Number of escalated alerts / Total number of alerts) * 100
KPI.ALR.03	False Positive Alerts	Percentage of false positive alerts	Technical	Daily	Top Management/CISO/Operational	(Number of false positive alerts/ Total number of alerts) * 100
KPI.ALR.04	Alert Distribution by Rule	Calculates the number of alerts that were triggered by each SIEM rule	Technical	Daily	Top Management/CISO/Operational	Number of alerts that were triggered by each SIEM Rule
KPI.ALR.05	Alert Distribution by Category	Top 10 Alerts by Category	Technical	Daily	Top Management/CISO/Operational	Number of alerts by Category

Figure 5.4: List of Alerts and Vulnerabilities Metrics (ALR and VUL)

- **KPI.ALR.02 (Escalated Alerts)** - Identically to KPI.IHR.12, this performance indicator measures the number of alerts escalated to the SOC Manager. Due to the absence of a flag that would indicate if a given alert has been escalated for further investigation, it is not possible to measure this KPI.

5.2.4 Other Metrics

Finally, the remaining set of metrics did not follow a specific category, thus they were grouped in the "Other"set of metrics. These metrics cover subjects that range from user satisfaction, to the regulatory of security assessments.

Figure 5.5 illustrates a list of the remaining KPIs that were incorporated in system B. Due to the fact that these KPIs are more directed towards governance and compliance, they are not as easily exportable like the technical KPIs. Additionally, since none of the information associated with this set of KPIs was given, these visualizations could not be finalized and appear as empty in the dashboard.

Other Metrics						
#ID	Name	Description	Type	Frequency of Collection	Target Audience	Formula
KPI.OTH.01	Services with Security Requirementes	Number of IT services that comply with the security requirements	Compliance	Daily	Top Management/CISO/Operational	(Number of IT services that comply with the security requirements / Total number of services) * 100
KPI.OTH.02	Access Management	Time required to create, remove or change accesses, compared to the thresholds established by the organization	Compliance	Daily	Top Management/CISO	-
KPI.OTH.03	Security Assessments	Frequency of Security Assessments	Compliance	Daily	Top Management/CISO/Operational	Counter of the frequency of security assessments on an yearly basis
KPI.OTH.04	User Satisfaction	Level of user satisfaction regarding the quality and efficiency of information management	Compliance	Daily	Top Management/CISO	Summation of the level of satifaction of each user / Total number of users who answered the survey
KPI.OTH.05	Business Process Incidents	Number of business process incidents caused by missing information	Compliance	Daily	Top Management/CISO/Operational	(Number of business process incidents caused by missing information / Total number of business process incidents) * 100
KPI.OTH.06	Attack Prevented by Business Unit	Measures the number of attacks by business unit that were successfully contained before causing damage to the organization	Compliance	Daily	Top Management/CISO/Operational	Number of attacks prevented by business unit
KPI.OTH.07	Phishing Awareness Training	Measures the number of associates who completed their Phishing Awareness Training	Compliance	Daily	Top Management/CISO	(Number of associates with a completed Phishing Awareness Training / Total number of associates) * 100
KPI.OTH.08	Endpoint Security	Measures the percentage of endpoint devices with EDR or Hardening protection in place	Compliance	Daily	Top Management/CISO/Operational	(Endpoint devices with full protection (Hardening and EDR) / Total number of endpoint devices) * 100
KPI.OTH.09	Log Storage	Measures the amount of space available in the Log Management tool	Compliance	Daily	Top Management/CISO/Operational	(Total Storage - Occupied Storage)
KPI.OTH.10	Number of Threat Intelligence Sources	Number of Threat Intelligence sources integrated in the SIEM	Compliance	Daily	Top Management/CISO/Operational	Number of Threat Intelligence sources integrated in the SIEM

Figure 5.5: List of Other Metrics (OTH)

Finally, this is another set of metrics where not every one is visible in all three dashboards. In fact, only seven out of the ten KPIs are visible to the *Operational* team.

5.2.5 Risk Metrics

In the previous sections, various KPIs with different categories have been analyzed. Nonetheless, it was stated that the solution would incorporate a risk management component. For this a set of KRIs, that would gauge risk, needed to be devised. Luckily, almost all KPIs had an already in place mechanism of thresholds that would help the user interpret the visualizations being showcased. Hence, by taking advantage of the thresholds one could develop one KRI for every KPI that would evaluate if its value is low, medium or high by quantifying it on a specific scale. Finally, by grouping/averaging every KRI value one could measure the risk level the security team is exposed to. This logic was what gave birth to the risk management mechanism in place.

Almost every visualization (with a few exceptions) has two threshold values that help users interpret them and take action if necessary. These two values place KPIs is one of three different risk zones depending on its value: low, medium and high.

For instance, taking the average time to confirmation metric (KPI.IHR.02). The two threshold values are: 10 minutes between low and medium, and 45 minutes between medium and high. In other words, if the metric has a value smaller than 10 its risk is low. If it is between 10 and 45 its risk is medium. If it is higher than 45, then it has high risk. Hence, depending on the value of the KPI, it is logical what the qualitative evaluation of risk would be (low, medium or high). Despite this, a quantitative formula was developed where the risk score varies from 0 to 100. If the KPI is in the low risk zone, the KRI can vary from 0 to 25 score. If the KPI is in the medium risk zone, the KRI can vary from 26 to 75 score. Finally, if the KPI is in the high risk zone, the KRI can vary from 76 to 100 score.

With the exception of the KPIs listed below, every remaining performance indicator had a KRI associated with it which is used to measure risk.

- **KPI.IHR.06**, **KPI.IHR.07** and **KPI.IHR.11** do not have a respective KRI. This decision was made because all 3 KPIs represent informative metrics which show the evolution of specific values through time. Hence, no risk can be extracted from them through the same methods used in the remaining ones.

- **KPI.IHR.10** and **KPI.IHR.13** do not have a respective KRI because it would be redundant. Whereas the former already has KPI.IHR.09, the latter has KPI.IHR.14. Each has a respective KRI which serve the same purpose.

- **KPI.ALR.04** and **KPI.ALR.05** do not have a respective KRI. Both indicators provide a perception of how the distribution of alerts is made by category and rule. Hence, these metrics are purely informative and no risk can be extracted from them through the same methods used in the remaining examples.

Figures

]5.7, 5.8 and 5.9 showcase every KRI. Each is identified by its category, a unique identifier (mirroring the associated KPI), a name, a target audience and a formula of calculation that identify the thresholds that apply to that metric.

	Risk Metrics		
Risk Component	**Name**	**Target Audience**	**Formula**
Financial	Incident Financial Estimation (KRI.FIN.01)	Top Management/CISO	KPI<300: (Score 0-25) 300<=KPI<450: (Score 25-75) 450<=KPI < 750: (Score 75-100)
Financial	Financial and Reputacional incidents (KRI.FIN.02)	Top Management/CISO	KPI < 5%: (Score 0-25) 5%<=KPI <10%: (Score 25-75) KPI >= 10%: (Score 75-100)
Financial	Security Team Expenses (KRI.FIN.03)	Top Management/CISO	Expenses <= Budget: (Score 0-25) Expenses > Budget: (Score 25-100)* *Score is 100 when the expenses are double the budget

Figure 5.6: Risk Metrics of the Financial Component

	Risk Metrics		
Risk Component	**Name**	**Target Audience**	**Formula**
Incident	Reactive Counter-Measures (KRI.IHR.01)	Top Management/CISO/Operational	KPI > 60%: (Score 0-25) 40% < KPI =< 60%: (Score 25-75) KPI <= 40%: (Score 75-100)
Incident	Proactive Counter-Measures (KRI.IHR.02)	Top Management/CISO/Operational	KPI < 5%: (Score 0-25) 5%<=KPI<10%: (Score 25-75) KPI >=10%: (Score 75-100)
Incident	Time to Confirmation (KRI.IHR.03)	Top Management/CISO/Operational	KPI<10m: (Score 0-25) 10m<=KPI<45m: (Score 25-75) 45m <= KPI < 55m: (Score 75-100)
Incident	Time to Containment (KRI.IHR.04)	Top Management/CISO/Operational	Weighted average of the time to containment scores of the critical (75% weight on the score) and non-critical incidents (25% weight on the score)
Incident	Time to Closure (KRI.IHR.05)	Top Management/CISO/Operational	Weighted average of the time to closure scores of the critical (75% weight on the score) and non-critical incidents (25% weight on the score)
Incident	Unscheduled Downtime (KRI.IHR.08)	Top Management/CISO/Operational	KPI < 45m: (Score 0-25) 45m<=KPI<450m: (Score 25-75) 450m <= KPI < 495m: (Score 75-100)
Incident	Critical Assets Integrated in the SIEM (KRI.IHR.09)	Top Management/CISO/Operational	KPI >= 90%: (Score 0-25) 60% <= KPI < 90%: (Score 25-75) KPI < 60%: (Score 75-100)
Incident	Escalated Incidents (KRI.IHR.12)	Top Management/CISO/Operational	KPI < 5%: (Score 0-25) 5%<=KPI<10%: (Score 25-75) KPI >= 10%: (Score 75-100)
Incident	Unresolved Incidents (KRI.IHR.14)	Top Management/CISO/Operational	KPI < 5%: (Score 0-25) 5%<=KPI<10%: (Score 25-75) KPI >= 10%: (Score 75-100)

Figure 5.7: Risk Metrics of the Incident Component

Last but not least, there exist five more Key Risk Indicators in System B:

- **Financial Risk** - This KRI aggregates the financial risk scores in a single value, through a simple average.
- **Incident Risk** - This KRI aggregates the incident risk scores in a single value, through a simple average.
- **Alerts and Vulnerabilities Risk** - This KRI aggregates the alerts and vulnerabilities risk scores in a single value, through a simple average.

	Risk Metrics		
Risk Component	**Name**	**Target Audience**	**Formula**
Alerts and Vulnerabilities	Alerts Investigated per Analyst (KRI.ALR.01)	Top Management/CISO/Operational	KPI < 1d: (Score 0-25) 1d <= KPI < 2d: (Score 25-75) KPI >= 2d: (Score 75-100)
Alerts and Vulnerabilities	Escalated Alerts (KRI.ALR.02)	Top Management/CISO/Operational	KPI < 5%: (Score 0-25) 5%<=KPI<10%: (Score 25-75) KPI >= 10%: (Score 75-100)
Alerts and Vulnerabilities	False Positive Alerts (KRI.ALR.03)	Top Management/CISO/Operational	KPI < 5%: (Score 0-25) 5%<=KPI<10%: (Score 25-75) KPI >= 10%: (Score 75-100)
Alerts and Vulnerabilities	Volume of Vulnerabilities (KRI.VUL.01)	Top Management/CISO/Operational	KPI < 500: (Score 0-25) 500 <= KPI < 1000: (Score 25-75) KPI >= 1000: (Score 75-100)

Figure 5.8: Risk Metrics of the Alerts and Vulnerabilities Component

	Risk Metrics		
Risk Component	**Name**	**Target Audience**	**Formula**
Other	Services with Security Requirements (KRI.OTH.01)	Top Management/CISO/Operational	KPI >= 90%: (Score 0-25) 70% <= KPI < 90%: (Score 25-75) KPI < 70%: (Score 75-100)
Other	Access Management (KRI.OTH.02)	Top Management/CISO	KPI <= 1d: (Score 0-25) 1d < KPI <= 5d: (Score 25-75) KPI > 5d: (Score 75-100)
Other	Security Assessments (KRI.OTH.03)	Top Management/CISO	KPI >= 2: (Score 0-25) 1 <= KPI < 2: (Score 25-75) KPI < 1: (Score 75-100)
Other	User Satisfaction (KRI.OTH.04)	Top Management/CISO	KPI >= 4: (Score 0-25) 2.5 <= KPI < 4: (Score 25-75) KPI < 2,5: (Score 75-100)
Other	Business Process Incidents (KRI.OTH.05)	Top Management/CISO/Operational	KPI < 10%: (Score 0-25) 10% <= KPI < 40%: (Score 25-75) KPI >= 40%: (Score 75-100)
Other	Attacks Prevented by Business Unit (KRI.OTH.06)	Top Management/CISO/Operational	KPI >= 60%: (Score 0-25) 40% <= KPI < 60%: (Score 25-75) KPI <= 40%: (Score 75-100)
Other	Phishing Awareness Training (KRI.OTH.07)	Top Management/CISO	KPI >= 90%: (Score 0-25) 60% <= KPI < 90%: (Score 25-75) KPI < 60%: (Score 75-100)
Other	Endpoint Security (KRI.OTH.08)	Top Management/CISO/Operational	KPI >= 90%: (Score 0-25) 60% <= KPI < 90%: (Score 25-75) KPI < 60%: (Score 75-100)
Other	Log Storage (KRI.OTH.09)	Top Management/CISO/Operational	KPI >= 60%: (Score 0-25) 30% <= KPI < 60%: (Score 25-75) KPI < 30%: (Score 75-100)
Other	Threat Intelligence Sources (KRI.OTH.10)	Top Management/CISO/Operational	KPI >= 10: (Score 0-25) 5 <= KPI < 10: (Score 25-75) KPI < 5: (Score 75-100)

Figure 5.9: Risk Metrics of the Other Component

- **Other Risk** - This KRI aggregates the remaining risk scores in a single value, through a simple average.

- **Global Risk Index** - The final indicator is the global risk index which is calculated through a simple average of every KRI. This decision was made so that every KRI had the same weight on the final risk score.

After reviewing every different metric that was integrated in the dashboard, the following section aims to help the reader understand how the information that brings these KPIs and KRIs to life was extracted.

5.3 Data Ingestion

Moving on to the topic of data sources, the list below points out each type of data source incorporated in the system:

1. **RESTful APIs** - The client provided access credentials for three different applications: TheHive, IBM QRadar and InsightVM. Through these three RESTful APIs it was possible to gather data that could be used to calculate a portion of the whole set of KPIs.

2. **CSV Files** - The data stored in static CSV files is either used to calculate a KPI or to support other data towards the calculation of a KPI.

As we have seen the data ingestion process is handled through a set of Python scripts that manipulates the data sources listed above. The data ingestion process is conducted in the following manner: a selected Python script, which triggers functions that export and store data from each data source, is scheduled to be ran on a timely basis (every 24 hours). In fact, this mechanism was implemented through the use of Windows' Task Scheduler functionality which lets users create tasks that automate procedures.

In order to understand how the data ingestion process is performed, all the files that support the data ingestion process will be described. In fact, Table 5.1 aims to condense this information in a clear format.

Now that the user has a grasp the main components of the data ingestion process, the following activity diagrams aim to describe it in a step by step format.

Figure 5.10 illustrates how the ingestion process is handled in a top level perspective. It shows the order of operations called by the main script. The process starts by checking if every collection that is going to be used to store data in is created. Afterwards, it checks through the various CSV files in order to check for new data. Finally, it accesses several endpoints of different applications and extracts data from these sources.

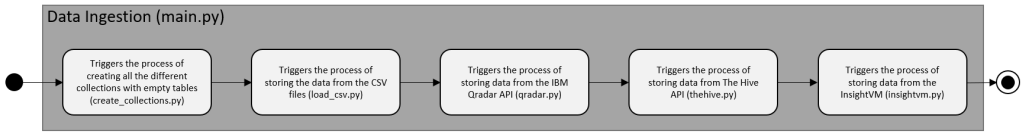


Figure 5.10: Activity Diagram of the data ingestion process performed in *main.py*

Figure 5.11 illustrates the steps performed when the main script triggers the *create_collections.py* file's main function.

Figure 5.12 illustrates the steps performed when the main script triggers the *load_csv.py* file's main function.

Figure 5.13 illustrates the steps performed by each file that is responsible for accessing data through RESTful APIs.

File Name	Description
main.py	This file is scheduled, by Windows, to be ran on a daily basis and is the one that triggers all the data ingestion functions calls.
create_collections.py	This file verifies if all the collections exists, in case they don't it creates an empty collection. This is important because if for some reason an error occurs and the collection does not exists, Power BI will display an error message instead of saying that there is no data in a visualization.
load_csv.py	This file is in charge with verifying for the existence of the CSV files and ingesting their data.
thehive.py	This file extracts data regarding the latest incidents, transforms it and ingests it to MongoDB.
qradar.py	This file extracts data regarding the latest security alerts, parses and transforms it and stores it into MongoDB.
insightvm.py	This file extracts data regarding the latest vulnerabilities found on the organization's devices, transforms it and stores it in MongoDB.
config.ini	This configurations file separates some key information from the rest of the files. Variables like the IP address, port number and credentials to access the three APIs listed above, are stored separately in this file.
log.txt	This text file is used to document the daily data ingestion process, so that the team can verify if there were any error, or exceptions which were triggered during the daily data ingestion process.

Table 5.1: Files in charge of the data ingestion process

To sum up, different data ingestion methods had to be arranged depending on the way the client could feed information. Thus, two primary methods came to be: ingestion through making HTTP requests to endpoints of an application's RESTful API and through accessing information stored in CSV files. On this topic the following section aims to describe all three RESTful APIs incorporated in the solution.

5.3.1 RESTful APIs

"A REST API (also known as RESTful API) is an application programming interface (API or web API) that conforms to the constraints of REST architectural style and allows for interaction with RESTful web services"[69]. In other words, it is a way for developers to access the resources of an application through an exposed interface. APIs usually have a documentation set that describes its set of endpoints. Hence, below is a detailed guide of what endpoints were accessed and what data was extracted for each RESTful API.

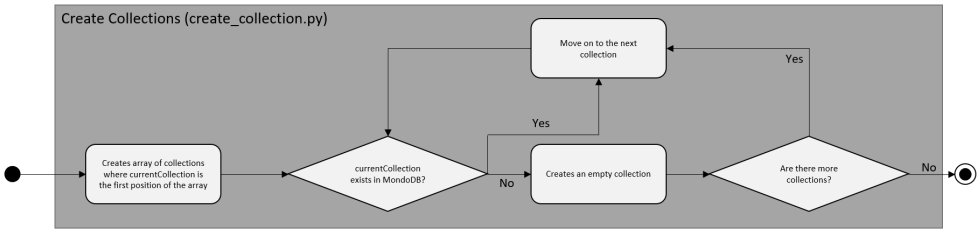


Figure 5.11: Activity Diagram of the data ingestion process performed in *create_collection.py*

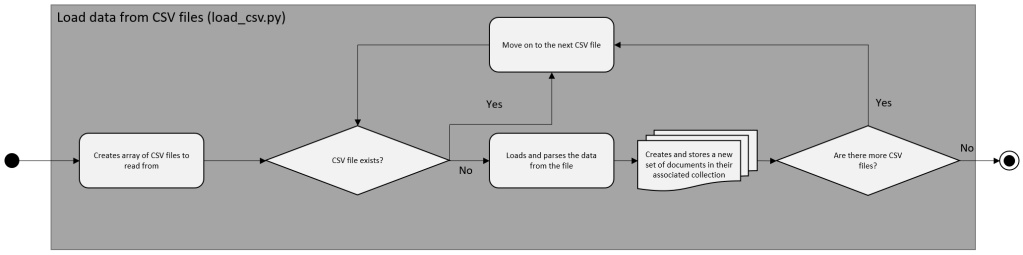


Figure 5.12: Activity Diagram of the data ingestion process performed in *load_csv.py*

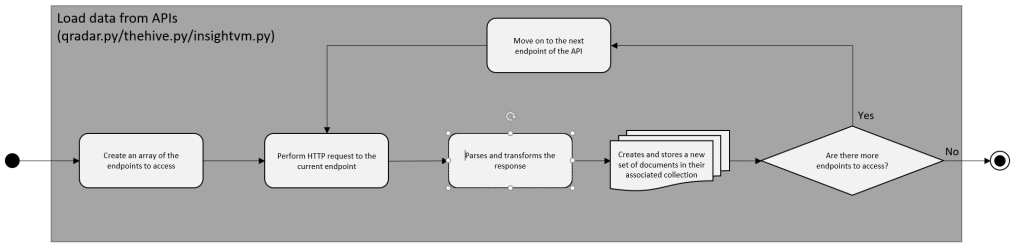


Figure 5.13: Activity Diagram of the data ingestion process performed in *thehive.py*, *qradar.py* and *insightvm.py*

5.3.1.1 TheHive: Security Incident Response Platform

TheHive is "a scalable, open source and free Security Incident Response Platform, designed to make life easier for SOCs, CSIRTs, CERTs and any information security practitioner dealing with security incidents that need to be investigated and acted upon swiftly."[66] Basically, TheHive is a ticketing platform where members of the SOC team can collaborate in order to conduct investigations simultaneously.

Being the main platform used by the client to handle security incidents, this API represented the main source of data regarding KPIs related to security incidents.

After consulting its documentation, two endpoints were identified as the ones to export relevant information to incorporate in the dashboard. Table 5.2 below highlights each one:

HTTP Method	URI	Description
GET	/api/case	Returns a list of security incidents
POST	/api/case/{caseId}/task/_search	Returns the tasks associated with that specific security incident

Table 5.2: TheHive Endpoints

As it can be verified above, two different data types were identified which contained interesting information to include in the data set, the data objects **Case** and **Task**. Whereas the Case object represents a security incident, each Task object is associated with a Case and contains information about in which stage of remediation the incident is in.

Due to the fact that TheHive's documentation data mapping of each object partially mismatches what the client's implementation of TheHive provides, only the schema of the object that is stored in the database will be shared.

First, the security incident object is stored in the "Incidentes (MongoDB)"collection and has the following properties:

- **_id: Integer** - ID of the incident.
- **Category: String** - Category of the incident.
- **Status: String** - Status of the incident, if it is open or resolved.
- **Created by: String** - User who created the incident.
- **Creation Date: Date** - Incident creation date.
- **Open Date: Date** - Date when the incident was first opened.
- **Close Date: Date** - Date when the incident was closed.
- **OS Machine: String** - OS of the machine that was struck by the security incident. This property is used as a placeholder field since the TheHive does not pass this information. It could be used to calculate KPI.IHR.9.
- **Effects: String** - Identifies if the security incident has any type of reputational or financial impact to the organization. This property is used as a placeholder field since the TheHive does not pass this information. It can be used to calculate KPI.FIN.2.
- **Unit: String** - Identifies the security unit affected by the security incident. This property is used as a placeholder field since the TheHive does not pass this information. It can be used to calculate KPI.OTH.6.
- **Cause: String** - Identifies the cause behind the security incident. This property is used as a placeholder field since the TheHive does not pass this information. It can be used to calculate KPI.OTH.5.

- **Description: String** - Description of the incident.
- **Title: String** - Title of the incident.
- **Assigned to: String** - User who is assigned to the incident.
- **Severity: Integer** - Incidents have three levels of severity 1, 2 and 3. Respectively, these values translate to a low, medium and high severity.
- **Resolution Status: String** - Gives information about the resolution of the incident, if it a False-Positive/True-Positive/Indeterminate/Other.
- **Escalated: Boolean** - True or false if the incident has been escalated to the SOC Coordinator for further investigation. This property is used as a placeholder field since the TheHive does not pass this information. It can be used to calculate KPI.IHR.10.

Finally, the task object is stored in the "Tarefas (MongoDB)"collection has the following properties:

- **_id: Integer** - ID of the task.
- **Creation Date: Date** - Task creation date.
- **Open Date: Date** - Date when the task was first opened.
- **Close Date: Date** - Date when the incident was closed.
- **Status: String** - Status of the task: *Cancel*, *Completed* or *In Progress*.
- **Created by: String** - User who created the task.
- **Description: String** - Description of the task.
- **Title: String** - Title of the task.
- **Assigned to: String** - User who is assigned to the task.
- **Stage: String** - Stage of the incident remediation process. There are 5 stages: Identification, Containment, Eradication, Recovery and Lessons Learned.
- **Associated Incident: Integer** - ID of the incident associated with the task.

5.3.1.2 IBM QRadar: SIEM

IBM QRadar is the SIEM solution which is integrated in the SOC of Client B. As a SIEM, QRadar provides real-time visibility into the organization's IT infrastructure, triggering security alerts in case suspicious activity is detected.

Being the main platform used by the client to trigger security alerts, this API represents the main source of data regarding KPIs related to security alerts.

After consulting its documentation, three endpoints were identified as the ones to export relevant information to incorporate in the dashboard. Table 5.3 highlights each one:

HTTP Method	URI	Description
GET	/api/siem/offenses	Returns a list of security alerts
GET	/api/siem/offense_types	Returns a list with each type of security alert
GET	/api/siem/offense_closing_reasons	Returns a list of offense closing reasons
GET	/api/asset_model/assets	Returns a list of assets incorporated in the SIEM

Table 5.3: IBM QRadar Endpoints

Due to the fact that QRadar's documentation is private, it it not possible to provide visibility into the response of each request. Nonetheless, a brief explanation of each endpoint, as well as the schema of all the data that was gathered through these endpoints will be detailed below.

Firstly, the offenses endpoint returns the majority of the information needed towards building the security alerts KPIs. As such, each security alert object stored in the "Alerts (MongoDB)"collection, has the following schema:

- **_id: Integer** - ID of the alert.
- **Description: String** - Description of the alert.
- **Category: String** - Category of the alert
- **Assigned to: String** - User the alert is assigned to.
- **Status: String** - Status of the alert: Open or Closed.
- **Creation Date: Date** - Alert creation date.
- **Close Date: Date** - Date when the alert was closed.
- **Closing Reason: Integer** - ID of the reason why the alert was closed, needs to be matched with data from another endpoint.
- **Severity: Integer** - Severity of the alert on a scale from 1 to 10.
- **Rules: String** [] - Array containing a list of the type of rules that were behind the creation of the alert.
- **Event Counter: Integer** - Counter of the number of events that constitute the alert.

The second and third endpoints contextualize some indirect information passed by the offenses endpoint. Both the type of security alert and its closing reason is passed down with an id that needs to be matched with the help of the endpoints two and three.

Schema of the "Offense Type"data object:

- **_id: Integer** - ID of the offense/alert.
- **Type: String** - Category of the alert.

Schema of the "Offense Closing Reason"data object:

- **_id: Integer** - ID of the offense/alert.
- **Reason: String** - Alert closing reason.

Finally, this last endpoint is relevant towards calculating the percentage of critical assets incorporated in the SIEM. Hence, a list of assets is stored each represented by the following schema:

- **_id: Integer** - IP of the asset.

5.3.1.3 InsightVM: Vulnerability Management Tool

InsighVM is a vulnerability management tool, that leverages the Rapid7 Insight platform, a library of vulnerabilities which provides an efficient way to collect an organization's vulnerability data [51]. InsightVM's API supports the REST design pattern. Hence, it is integrated in System B through this interface.

As the vulnerability management tool used by the client, this API represents the main source of data behind the vulnerability KPIs.

After consulting its documentation, one endpoint was identified as the one to export relevant information from. Table 5.4 highlights the endpoint in question:

HTTP Method	URI	Description
GET	/api/3/scans	Returns a list of all the scan results performed on the organization's devices.

Table 5.4: InsightVM Endpoints

The schema of the documents that are stored in the "Scans (MongoDB)"collection has the following properties:

- **Asset counter: Integer** - Number of investigated assets
- **Group: String** - Name of the Asset Group
- **Initial Date: Date** - Date when the scan began

- **Final Date: Date** - Date when the scan was finished
- **Total Vulnerabilities: Integer** - Total number of vulnerabilities
- **Critical Vulnerabilities: Integer** - Number of critical vulnerabilities
- **Severe Vulnerabilities: Integer** - Number of severe vulnerabilities
- **Moderate Vulnerabilities: Integer** - Number of moderate vulnerabilities

5.3.2 CSV Files

Now that it has been described how one half of the data ingestion process is conducted, this section aims to describe how the remaining information is gathered.

The remaining data was decided to be imported through a set of CSV files for two reasons: because the application which stored that information did not have a RESTful API or the information did not exist. Hence, for the information which did not have an API it was exported in CSV format and stored in its specific folder for the scripts to analyze. For the information that did not exist, a placeholder set of CSV files were created, each with a specific data schema.

Table 5.5 represents the folder name, description of the data and the document schema that is stored in each CSV file.

5.4 Visualizations

During the implementation phase, specifically during the development of the situational dashboard, a set of test dashboards with arbitrary data was developed. Two main advantages were gained by this. Firstly, time was not being wasted while the client worked on creating the credentials for accessing the data sources. Secondly, through the development of a mock-up for the each of the three dashboards, the development team could present these set of dashboards to the client and have access to feedback at an earlier stage.

This set of mock-ups and testing dashboards are going to have a fundamental importance in this section. Showcasing the visualizations of the dashboard with real data would be a breach of confidentiality. Hence, the dashboards developed during the testing phase can be used to both guarantee confidentiality and to showcase how every visualizations would be presented (even the ones that could not be finalized due to a lack of information).

Finally, and before beginning to showcase the dashboard, it was decided that the set of images that will be displayed in the next couple of sections are in respect to the *Top Management Dashboard*. This was the most logical decision, since it is the one that is most rich in terms of metrics. As we have seen, the only difference between the *Top Management Dashboard* and the *CISO's* dashboard is the fact that the Monthly Overview is customized

Folder Name	Description per file	Document Schema
Critical Assets	List of the addresses of all critical assets (Crown Jewels)	{_id, Hostname, App, IP Address}
Log Management	Periodic extraction of used and total storage of the log management tool	{_id, Date, Total Storage, Used Storage}
Security Assessments	Lists every security assessment performed	{_id, Date, Assessment}
Staff Position and Cost/hour	Lists every member of the SOC staff by position and rate per hour	{_id, Username, Alternative ID, Position, Cost/hour}
Financial of Security Team	Lists every activity performed by the Security Team by cost and budget	{_id, Activity, Cost, Budget}
Threat Intelligence Sources	Lists every Threat Intelligence Source by date of addition	{_id, Date, TI Source}
Phishing Awareness Training	Lists all training performed by each user	{_id, Date, User, Training}
Access Management	Lists all access management activities performed Lists the SLA maximum value that each type of activity may reach	{_id, Type, Start Date, End Date} {_id, Type, Duration}
Endpoint Protection	Lists every device and its EDR and Hardening protection status	{_id, Machine, EDR, Hardening}
Security Measures	List of every Security Measure available	{_id, Measure}
Services with Security Requirements	Lists every service accompanied with a flag that checks if the service complies with the security requirements established	{_id, Date, Service, Requirement}
System Downtime	Lists every system downtime that has occurred	{_id, Machine, Start Date, End Date}
User Satisfaction	Lists the results of user satisfaction surveys	{_id, User, Type, Score}

Table 5.5: CSV Files Document Schema

for each one. The *Operational Dashboard* on the other hand has a few additional differences in respect to the other two. Since there are a couple of metrics which are not present in this dashboard, the risk index formula also changes depending on the metrics that are missing. All in all, since the *Top Management Dashboard* is the one that is most rich in terms of metrics, it is the one that will be used to showcase the solution.

5.4.1 Monthly Overview

The opening slide of the dashboard is the Monthly Overview. It provides a top level perception and a monthly summary of the most important metrics of each set, for each dashboard type. In fact, the Top Management and the CISO Dashboards have different metrics displayed. Additionally, the Operational Dashboard does not even have a Financial Metrics section.

As it can be seen in Figure II.1, the "Monthly Overview"panel can be divided into the following visual components:

1. **Header** - The header is present in every dashboard, since it represents the key way of navigating in the solution. Through the header's tabs, the user can not only quickly understand what he is seeing, but also navigate to any set of metrics at his disposal. The top left usually has the company logo. In this case, it was removed in order to maintain confidentiality.

2. **Risk Score** - This key component is also present in every dashboard's *Monthly Overview*. It provides the user immediate visibility over the level of risk. Thus, the global risk score is the first metric that was chosen to be incorporated in this slide.

3. **Metrics** - On top of the risk score, the *Monthly Overview* also various visualizations separated by their category. The *Top Management* has a bigger focus regarding financial visualizations and others that monitor the performance of the SOC Staff. The *CISO* gives a smaller importance to financial metrics deciding to incorporate "Other"metrics. Finally, the *Operational Dashboard* has no visibility in regards to financial metrics and therefore has more screen for showcasing more metrics of each category.

5.4.2 Risk Metrics

When the user interacts with the second tab on the header, he is transported to the KRI section. This section aggregates all the risk indicators, giving insight to their security scores and thresholds. The Risk Metrics section is divided into: Global Risk, Finance Risk, Incidents Risk, Alerts and Vulnerabilities Risk and Other Risk. One can inspect each individual section through the set of buttons on the left. The following sections aim to describe each risk component in detail.

5.4.2.1 Global Risk

Beginning with the "Global Risk", this is a rather simple and informative slide. On top of consulting the Global Risk Index, the user can read a text box which includes information such as, the components that affect the risk score, its formula of calculation and how does one interpret its value.

5.4.2.2 Finance

Moving along to the first component of risk, the financial one. Figure II.3, which represents this slide, has four different visualizations, three of which represent the financial KRIs and a big one which aggregates these three values into one.

Additionally, there is a level of interactivity that applies to the rest of the risk indicators. In order to check the threshold values of each KPI, by clicking on the title of the high level risk component, the smaller KRIs are substituted with their respective threshold values. This can be verified in Figure II.4.

5.4.2.3 Incidents

Moving along to the second component of risk, the incidents one. In Figure II.5, ten different visualizations appear, nine which represent the incidents KRIs and a big one which aggregates these nine values into one.

5.4.2.4 Alerts and Vulnerabilities

Moving along to the third component of risk, the incidents one. In Figure II.6, five different visualizations appear, four which represent the alerts and vulnerabilities KRIs and a big one which aggregates these four values into one.

5.4.2.5 Other

Moving along to the final component of risk which represents the remaining set of metrics. In Figure II.7, eleven different visualizations appear, ten which represent the remaining KRIs and a big one which aggregates these ten values into one.

5.4.3 Financial Metrics

When the user interacts with the third tab on the header, he is transported to the first set of KPIs, ones which refer to the financial component. This section aggregates all the financial performance indicators, giving access to their respective visualizations and a complete description. In fact, some KPIs have more than a single visualization. Each visualization is accompanied with a description of the visualization, source of information, formula of calculation of the KPI and how interpret its value in a risk point of view.

The Financial Metrics section is divided into: Incident Financial Estimation, Financial and Reputational Costs of Security Incidents and Expenses of Security Team. The user can inspect each individual indicator through interacting with the set of buttons on the left. The following sections aim to describe each of these components in detail.

5.4.3.1 Incident Financial Estimation

The first financial KPI is the Incident Financial Estimation, which has three different visualizations for the user to interpret. This KPI can be visualized in Figure II.8.

Before describing all three visualizations, its important to refer that almost all visualizations have a time filter in the upper left corner. This way, only incidents that have been created during that time period are considered in the calculation of the KPIs and visualizations. This gives the dashboards a bigger level of versatility, as users can consult data in specific time frames.

The first visualization a table with every incident, its ID, category, close date and cost. Users can order the table in an upward or downward fashion by clicking on any of the header's attributes. This way they can order the list of incidents from the most costly to the less costly or vice versa. Finally, the viewer can also consult the total cost of all the incidents in the final row.

The second visualization represents a bar chart which calculates the average incident response cost by category. Furthermore, the visualization has a green and yellow lines. These lines represent the thresholds used to evaluate the level of risk. The colour of the bars in the graph also change depending on the value of the KPI.

Finally, the third visualizations represents a line chart of the evolution of the incident response cost by both per category of incident. Furthermore, the visualization has a green and yellow lines. These lines represent the thresholds used to evaluate the level of risk.

The last two visualizations also have one button each which identifies the ID of the KPI. Upon clicking on each button, the other visualizations is substituted with a text box. This text box provides vital information, such as, a description of the visualization, source of information, formula of calculation of the KPI and how to interpret its value in a risk point of view. Figure II.9 showcases this informative text box for KPI.FIN.01.1.

5.4.3.2 Financial and Reputational Costs of Security Incidents

The second financial KPI is the Financial and Reputational Costs of Security Incidents, which has three different visualizations for the user to interpret. The visualizations behind this KPI can be consulted in Figure II.10.

The first visualization a table with every incident, its ID, category, close date and the the incident financial or reputational cost. Users can order the table in an upward or downward fashion by clicking on any of the header's attributes. This way they can order the list of incidents any way they prefer.

The second visualization represents a bar chart which calculates the percentage of incidents with reputational or financial costs per category. Furthermore, the visualization has a green and yellow lines. These lines represent the thresholds used to evaluate the level of risk. The colour of the bars in the graph also change depending on the value of the KPI.

Finally, the third visualizations represents a meter that measures the overall percentage of incidents with reputational or financial costs. The visualization's colour change according to the thresholds defined for this KPI. Furthermore, below the visualization is a text box which provides vital information regarding the description of the visualization, source of information, formula of calculation of the KPI and how to interpret its value in a risk point of view.

5.4.3.3 Expenses of the Security Team

The third and final financial KPI covers the Expenses of the Security Team, which are compromised by a single visualization with its respective descriptive text box. The visualization behind this KPI can be consulted in Figure II.11.

This visualization is a horizontal bar chart which serves to compare each security team's activity's expenses with the existing budget. Furthermore, below the visualization is a text box which provides vital information regarding the description of the visualization and source of information.

5.4.4 Incident Metrics

When the user interacts with the fourth tab on the header, he is transported to the second set of KPIs, one which refers to the incidents component. This section aggregates all the incident performance indicators, giving access to their respective visualizations and a complete description. In fact, some KPIs have more than a single visualization. Each visualization is accompanied with a description of the visualization, source of information, formula of calculation of the KPI and how to interpret its value in a risk point of view.

The Incident Metrics section is divided into: Incident Matrix, Time to Confirmation, Time to Containment, Time to Closure, Reactive/Proactive Countermeasures, Unscheduled Downtime, Critical Assets Integrated in the SIEM, Incident Repetition, Escalated Incidents and Unresolved Incidents. The user can inspect each individual indicator through interacting with the set of buttons on the left. The following sections aim to describe each of these components in detail.

5.4.4.1 Incident Matrix

The first incident KPI covers two Incident Matrices, each illustrated by a single visualization. Figure II.12 showcases both visualizations.

Whereas the top visualization corresponds to a matrix of confirmed incidents, the matrix below illustrates the false positive incidents. Both visualizations serve the same

purpose, they display for each category of incident and severity, the number of confirmed/false positive incidents. For each column of severity, two values are shown: the number of incidents registered in the current month and if there has been an evolution of that value compared to the previous month. In fact, a red upwards arrow informs that the number of incidents has increased, a green downwards arrow shows the number has decreased and a neutral yellow arrow indicates the number is equal to the previous month. This way the security team can monitor the evolution of false positives and confirmed incidents compared to the previous month.

5.4.4.2 Time to Confirmation/Containment/Closure

This subsection will summarize three different KPIs since they are inherently similar. The "Time to Confirmation"KPI has two visualisations, whereas the "Time to Containment"and "Time to Closure"KPIs have four visualizations. The reason for this is that these last two metrics differentiate critical incidents with non critical incidents, in order to understand if critical incidents take more time to contain and close. Figure II.13 illustrates the "Time to Confirmation"performance indicator.

The visualization on the left is a bar chart which indicates for each category of incident the average amount of time it takes to confirm it, in minutes. The visualization is also accompanied with two thresholds lines and each bar chart changes colours depending where its value is relative to the thresholds. Furthermore, the visualization is accompanied with a text box with a description of the visualization, source of information, formula of calculation of the KPI and how to interpret its value in a risk point of view. Finally, the visualization on the right serves the same purpose but it illustrates how these same values evolve over time. It is supported by the same text box with descriptive information. Both visualizations can be filtered by the time slicer in the upper right corner.

Moving along to the last two KPIs, each slide has four visualizations which showcase the same information listed above but for different metrics. As it has been described, the amount of visualizations is doubled since there is a differentiation between critical and non critical incidents (with different thresholds each). Furthermore, there exists two buttons which when clicked showcase two description boxes with information surrounding the KPIs. The visualizations can also be filtered through the time slicer in the upper right corner. The images listed below showcase both KPIs with and without their descriptive text boxes. Finally, Figure II.14 illustrates the "Time to Containment"KPI and Figure II.15 showcases this KPI after clicking on one of the description buttons. Figure II.16 illustrates the "Time to Closure"KPI and Figure II.17 showcases this KPI after clicking on one of the description buttons.

5.4.4.3 Reactive/Proactive Counter-Measures

This subsection will summarize two different KPIs. The "Reactive Counter-Measures"KPI has two visualisations, whereas the "Proactive Counter-Measures"only has one.

The visualizations on the left refer to the reactive counter measures. The visualization above is a bar chart which groups the percentage of incidents with reactive measures implemented by category. Like other incidents, the colour of the bars change depending on the thresholds which are indicated by the green and yellow lines. On the other hand, the visualization below showcase the same information over time. Both visualizations also have a button which, when clicked, showcases a descriptive text box.

Regarding the final visualization on the right, which is associated with the Proactive Counter Measures performance indicator, it is illustrated by a simple meter. The visualization is also supported by a descriptive text box.

Finally, Figure II.18 illustrates the "Reactive/Proactive Counter-Measures"KPI and Figure II.19 showcases this KPI after clicking on one of the description buttons.

5.4.4.4 Unscheduled Downtime

The "Unscheduled Downtime"KPI is illustrated by two different visualization. Whereas the first visualization showcases the summation of each different machine, the first showcases how each of those values has evolved over time, month by month. Figure II.20 showcases both visualizations.

The left visualization is a bar chart which showcases the summation of each different machine's downtime duration. Like other visualizations the colour of the bar change according to where the value sits when compared to the thresholds represented by the green and yellow lines.

Finally, the right visualization illustrates how the summation of all machines with an unscheduled downtime has evolved temporarily. Both KPIs can be filtered using the time slicer on a upper right corner.

5.4.4.5 Critical Assets Integrated on the SIEM

This slide has two different KPIs, each illustrated by a single visualization. Figure II.21 showcases both visualizations.

Starting with KPI.IHR.9, it is illustrated by a meter which measures the percentage of critical assets integrated in the SIEM. The colour of the meter changes according to the KPI's value and its defined thresholds. The descriptive box below gives insight into this information and more.

Moving to KPI.IHR.10, it is illustrated by a simple table which matches the asset's name with its IP and a column indicating if the critical asset is integrated in the SIEM or not.

5.4.4.6 Repeated Incidents

This performance indicator measures the number of incidents that happened on machines with the same Operating System. Figure II.22 showcases this visualization.

This KPI is illustrated by a Matrix which crosses the category of the incident with the system OS that it targeted. Like other KPIs represented by matrices, the two columns on the indicator showcase the number of incidents that manifested on the current month and how this number has evolved compared to the previous month.

5.4.4.7 Escalated Incidents

This performance indicator measures the percentage of incidents which were escalated to the SOC Manager for further investigation. Figure II.23 showcases this visualization.

KPI.IHR.12 is illustrated by a bar chart where each bar represents a different incident category. The colour of each bar changes depending on the metric's value and the values of the thresholds, represented by the green and yellow lines. Finally, the visualization can be filtered through the time slicer on the upper right corner.

5.4.4.8 Unresolved Incidents

Finally, the "Unresolved Incidents", composed by the different KPIs, is the last slide of the Incident Metrics. The first KPI measures the number of unresolved incidents since the beginning of the month, whereas the second represents the number of unresolved incidents since the SOC was implemented. Figure II.24 showcases both visualizations.

KPI.IHR.13 is illustrated by a bar chart with a specific bar for the percentage of resolved incidents, whereas the remaining bars represents each incident category's percentage of unresolved incidents. Furthermore, it is also supported by the typical informative text box which helps the user interpret the KPI.

KPI.IHR.14 is illustrated by a pie chart. This chart separates the percentage of unresolved incidents by category and showcases the percentage of resolved incidents. Finally, the time slier on the upper right corner is used to filter the data on this visualization.

5.4.5 Alerts and Vulnerabilities Metrics

When the user interacts with the fifth tab on the header, he is transported to the set of KPIs which refer to the alerts and vulnerabilities component. This section aggregates all the indicators which refer to alerts triggered by the SIEM and vulnerabilities scanned on the vulnerability management tool. Every visualization has a degree of interactivity, since users can click on different properties which filters every visualization being displayed. Additionally, each KPI is accompanied with a description of the visualization, source of information, formula of calculation and how to interpret its value in a risk point of view.

This section is divided into: Alerts Investigated by Analyst, Escalated Alerts, False Positive Alerts, Alert Distribution by Rule, Alerts by Category and Distribution of Vulnerabilities by Asset Group. Finally, the user can inspect each individual indicator through interacting with the set of buttons on the left. The following sections aim to describe each of these components in detail.

5.4.5.1 Alerts Investigated per Analyst

This KPI is illustrated by two visualizations. Whereas the first, showcases a distribution of the alerts by analyst, the second measures the average amount of days each analyst takes to analyze the alert. Figure II.25 showcases both visualizations.

The visualization on the left is a simple pie chart with the the distribution of alerts that have been resolved by analyst.

The visualization on the right is a table which matches each analyst with the average amount of days he takes to resolve an incident. Additionally, the values of the metric change colours depending on the thresholds defined.

Finally, the text box below both visualizations describes them and provides additional information about each visualization to help the user interpret them. Both visualizations can be filtered with the time slicer on the upper right corner of the screen.

5.4.5.2 Escalated Alerts

The "Escalated Alerts"performance indicator is illustrated by a meter visualization which calculates the percentage of incidents which were escalated to the SOC Manager for further investigation. The color of the bar inside the meter changes values dependent of the thresholds that were assigned to this KPI. Finally, below the visualizations is a descriptive text box. Figure II.26 showcases this visualization.

5.4.5.3 False-Positive Alerts

This KPI is illustrated by two visualizations. Whereas the first, showcases a meter with the percentage of false-positive security alerts, the second shows how this value evolves over time on a monthly basis. Figure II.27 showcases this visualization.

The visualization on the left corresponds to a meter which changes values according to the defined threshold values.

The visualization on the right indicates how the percentage of false-positives evolved over time. Additionally, the threshold values are also illustrated in the visualization thought the green and yellow lines.

Finally, both visualization have a descriptive text box which the viewer can read to help them interpret the correspondent visual element. The time slicer in the upper right corner can be used to filter the data.

5.4.5.4 Alerts Distribution per Rule

This KPI is illustrated by two visualizations. Whereas the first showcases a bar chart with the distribution of alerts triggered by security rule, the second shows how this value evolves over time on a monthly basis. Figure II.28 showcases this visualization.

Finally, both visualization have a descriptive text box which the viewer can read to help them interpret both visual elements. The time slicer in the upper right corner can

be used to filter the data. Since this KPI is not considered toward the risk component of the solution, no thresholds needed to be defined.

5.4.5.5 Alert Distribution by Category

This KPI is illustrated by a single visualization, a bar chart where each bar is a different alert category. Like the previous visualization, it has a descriptive text box which the viewer can read to help them interpret both visual elements. The time slicer in the upper right corner can be used to filter the data. Since this KPI is not considered toward the risk component of the solution, no thresholds needed to be defined. Figure II.29 showcases this visualization.

5.4.5.6 Vulnerabilities

This KPI is illustrated by two visualizations, a bar chart and a line graph. Both have a descriptive text box which the viewer can read to help them interpret both visual elements. Like the previous visualization, the time slicer in the upper right corner can be used to filter the data. Since this KPI is not considered toward the risk component of the solution, no thresholds needed to be defined. Figure II.30 showcases both visualizations.

KPI.VUL.1.1 is illustrated by the visualization on the left of the screen. This bar chart groups data into asset groups which further categorizes each bar per vulnerability severity. Each bar counts the number of vulnerabilities with a specific severity inside a specific asset group.

KPI.VUL.1.2 is illustrated by the visualization on the right of the screen. This line graph showcases how the number of vulnerabilities by severity evolves over time.

5.4.6 Other Metrics

When the user interacts with the final tab on the header, he is transported to the set of KPIs which refer to the remaining metrics. This section aggregates all the indicators which refer to compliance metrics or even other metrics which simply did not fit in the remaining categories. Every visualization of this has a degree of interactivity, since users can click on different properties which filters every visualization being displayed. Additionally, each KPI is accompanied with a description of the visualization, source of information, formula of calculation and how to interpret its value in a risk point of view.

This section is divided into: IT Services with Security Requirements, Access Management, Security Assessments, User Satisfaction, Business Process Incidents, Attacks Prevented by Business Unit, Phishing Awareness Training, Endpoint Security, Log Storage and Number of Threat Intelligence sources. Finally, the user can inspect each individual indicator through interacting with the set of buttons on the left. The following sections aim to describe each of these components in detail.

5.4.6.1 IT Services with Security Requirements

This KPI measures the percentage of IT services with security requirements. Figure II.31 showcases this visualization.

KPI.OTH.1 is illustrated by a bar chart visualization where each bar measures, for each month of the year, the percentage of IT security systems which fulfill security requirements. The colour of the bar changes dependant on the thresholds represented by the green and yellow lines. A description of the KPI can be consulted below the visualization. It is possible to filter the visualization through the time slicer in the upper right corner of the screen.

5.4.6.2 Access Management

This KPI measures the average amount of time the SOC takes to perform access management tasks. This value is also compared to a maximum acceptable value decided by the organization. Figure II.32 showcases this visualization.

This visualization is a horizontal bar chart where each set of bars corresponds to an access management procedure. Furthermore, for each procedure there are two bars. The first, measures the average time in days it took to conclude the procedure. Finally, the second bar indicates the maximum tolerable value of each procedure. Below the visualization is a text box which provides vital information regarding the description of the metric and its source of information.

5.4.6.3 Security Assessments

This KPI measures the amount of security assessments performed one, two and three years ago. Figure II.33 showcases this visualization.

This visualization is a matrix that maps each type of security assessment to the amount of times it was performed one, two and three years into the past. Like other metrics, it also has a set of thresholds which influence its risk score, as well as the colour of the values in the matrix. Below the visualization the viewer has access to a descriptive text box, which supports the interpretation of the matrix.

5.4.6.4 User Satisfaction

This KPI measures an average user satisfaction score regarding information management, collected from surveys. Figure II.34 showcases this visualization.

This visualization is a horizontal bar chart which calculates the average score the users gave to the quality and timeliness of management information. Like other metrics, it also has a set of thresholds which influence its risk score, as well as the colour of the bars. Below the visualization the viewer has access to a descriptive text box, which supports the interpretation of the visualization.

5.4.6.5 Business Process Incidents

This KPI measures the percentage of Business Process Incidents caused by a lack of information. Figure II.35 showcases this visualization.

The visualization that corresponds to this KPI is a meter which illustrates the percentage of business process incidents caused by missing information. The colour of this meter varies according to a set of thresholds defined previously. Below the visualization, the viewer can consult the source of the information that builds the visualization, as well a description, the formula and thresholds explaining how to interpret the KPI compared to these values.

5.4.6.6 Prevented and Contained Attacks by Business Unit

This KPI measures the percentage of prevented and contained attacks, segregated by business unit. Figure II.36 showcases this visualization.

The visualization that corresponds to this KPI is a bar chart which illustrates the percentage of prevented attacks by business unit. Each bar represents a different business unit and the colour of each bar varies with the values of the thresholds defined previously. Below the visualization, the viewer can consult the source of the information that builds the visualization, as well a description, the formula and thresholds explaining how to interpret the KPI compared to these values.

5.4.6.7 Phishing Awareness Training

This KPI measures the percentage of employers who have performed their Phishing Awareness Training. This KPI can be consulted through two different visualizations. Figure II.37 showcases both visualizations.

The first visualization that corresponds to this KPI is a meter which illustrates the percentage of employers who have performed their Phishing Awareness Training. The colour of the meter varies with the values of the thresholds defined previously.

The second visualization is a line chart which illustrates how the value of the KPI evolves over time. This visualization can be filtered through the time slicer on the upper right corner of the screen.

Finally, below both visualizations the viewer can consult the source of the information, as well a description, the formula and thresholds explaining how to interpret the KPI compared to these values.

5.4.6.8 Endpoint Security

This KPI measures the percentage of endpoint systems with total protection (EDR and Hardening). This KPI can be consulted through three different visualizations. Figure II.38 showcases each of them.

All three visualizations are meters which reflected different percentages: endpoints devices with total protection, endpoints devices with EDR implemented and endpoints devices with Hardening. Endpoint Detection and Response (EDR) is "is an integrated endpoint security solution that combines real-time continuous monitoring and collection of endpoint data with rules-based automated response and analysis capabilities"[73]. Hardening is the process of "collection of tools, techniques, and best practices to reduce vulnerability in technology applications, systems, infrastructure, firmware, and other areas. The goal of systems hardening is to reduce security risk by eliminating potential attack vectors and condensing the system's attack surface. By removing superfluous programs, accounts functions, applications, ports, permissions, access, etc. attackers and malware have fewer opportunities to gain a foothold within your IT ecosystem"[57]. Hence, after verifying if endpoint devices have both properties, this information is used to conclude if the system is secured.

Finally, below the visualizations the viewer can consult the source of the information, as well a description, the formula and thresholds explaining how to interpret the KPI compared to these values.

5.4.6.9 Available Storage

This KPI measures the percentage of available storage is the Log Management Tool. This KPI can be consulted through a single visualization. Figure II.39 showcases this visualization.

The visualization that corresponds to this KPI is a meter which illustrates the percentage of available storage. The colour of the meter varies with the values of the thresholds defined previously.

5.4.6.10 Threat Intelligence Sources

This KPI measures the number of threat intelligence sources integrated in the SIEM of the SOC. This KPI can be consulted through a single visualization, which can be consulted in Figure II.40.

The visualization that corresponds to this KPI is line chart which illustrates the evolution of the percentage of threat intelligence sources integrated in the SIEM. The two different green and yellow lines present in the visualizations represent the thresholds that help to measure the risk associated with this KPI.

5.5 Conclusion

To sum up the contents of this chapter, it aimed at describing every step of the implementation process of System B. Firstly, it starts by describing how the system was setup, what are its main components and how each one of them communicates with each other. Afterwards, a description of each KPI and KRI is performed, as well as their associated

formulas of calculation. Furthermore, we have the data ingestion process which is detailed step by step showcasing activity diagrams for each python script. These diagrams represent how each script, which is associated with a different data source, performs its part of the overall data ingestion process. Finally, each one of the dashboards' panels is described along with their respective visualizations.

All in all, unlike System A, this solution left the testing phase and transitioned to production where it was successfully completed, following the implementation described in this chapter.

6

Product Evaluation

This chapter is composed by two different section. The first section(6.1) briefly describes how the evaluation process of System B's dashboard is conducted in the next section. As described, this second and final section(6.2) showcases different evaluation scenarios. Each scenario represents a verification that checks if the solution was capable of fulfilling a different requirement.

6.1 Evaluation Process

Initially, the evaluation process that the solution was supposed to be subjugated to was divided into three types. Unfortunately, due to the fact that after delivering System B, the student lost access to the solution, therefore also losing the ability to monitor it over time, these evaluations became impossible to perform.

1. **Data Ingestion Process Performance** - The first evaluation would be focused around the performance of the solution, specifically, of the data ingestion process, verifying the time that it would take to perform the daily scheduled data extraction tasks over time.

2. **Risk Indicators Robustness** - The second type of evaluation would be an analysis over the robustness of the risk index formulas. This assessment would be done in order to verify if over time the equations used to measure each risk value needed to be altered so as to refine the calculation of each score.

3. **Thresholds Accuracy** - Finally, the last type of evaluation that could be been performed would have been surrounding the values of the thresholds of each KPI. Ideally, the thresholds of each performance indicator would be monitored over time, in order to verify if some type of intervention was necessary. This way, this assessment would help establish the thresholds values so that they could be as accurate as possible.

Since none of the scenarios listed above were achievable, a different approach was considered. Instead of building specific test scenarios with the purpose of measuring the

performance of System B, it was decided that the evaluation process would compare the initial set of contributions that were proposed to be implemented, with the features the solution ended up providing. Hence the next section will cover each objective, representing each one as an evaluation scenario.

6.2 Evaluation Scenarios

Before the development of System B, a set of contributions were established. They represented the features that the system would ideally achieve. The following subsections list every one of these scenarios, explaining if the goal was or was not successfully integrated.

6.2.1 Scenario 1 - Designing a Visualization Interface

One of the mains objectives of this thesis was aimed at providing a centralized framework that would deliver a degree of situational awareness around the performance and risk of the security environment surrounding the organization. This information is usually conveyed through the means of a dashboard. When properly designed, dashboards are powerful monitoring visualization tools, due to their inherent visual nature and ability to aggregate and centralize information at a glance.

This objective was fulfilled and delivered while working in the client behind System B. In fact, prior to the dashboard's development, the client has to manually build monthly reports. Hence, the solution in place successfully filled this gap, automating this process.

6.2.2 Scenario 2 - Developing Dashboards for Different Audiences

Effective data visualization is not only about trying to illustrate every possible piece of information, it also requires a degree of efficiency. If viewers from different backgrounds use the same dashboard, it will inevitably contain a heavy load of visualizations so as to please each audience. Hence, segregating one dashboard into several ones, for each different type of user, allows for an easier viewing experience. This differentiation not only provides relevant information for each user, but also an enhanced control over what is displayed. Furthermore, this allows for the possibility to present sensitive data to one target audience, that the other cannot have visibility upon. For instance, System B's Top Management Dashboard provides visualizations about user satisfaction survey results, which are not visible to the remaining SOC Staff.

This objective was fulfilled and delivered while working in cooperation with the client associated with System B. In fact, as it was described, three different dashboards were developed: one for the Top Management, one for the CISO and another for the SOC's Operational Team.

6.2.3 Scenario 3 - Establishing a Risk Assessment Plan

Successfully measuring risk requires identifying and prioritizing the risk factors that should be monitored over time. Additionally, coming up with a plan to translate risk into a quantitative and/or qualitative scale is also important.

This objective was fulfilled and delivered while working with the client associated with System B. In fact, both the risk categorization, as well as their formulas of calculation, were all developed during the course of this dissertation.

6.2.4 Scenario 4 - Developing Key Performance and Risk Metrics

One of the key ingredients towards building a dashboard is their data visualization component, which can take the form of pie charts, line graphs, bar charts, sunbursts and many other illustrations. In order to assemble a set of visualizations, it is paramount to first know what information is going to be displayed. Hence, designing a set of metrics that measure performance and risk is a vital step in the development of the solution.

This objective was fulfilled and delivered while working in System B. In fact, prior to the involvement of the student in the project, EY and the Client discussed what key performance indicators should be displayed in the final dashboard. Regarding the key risk indicators, they were later designed by the student during the development of the solution.

6.2.5 Scenario 5 - Building a System Architecture

In order to provide the client with a solution that would be as automated and self sustained as possible, a system architecture had to designed that would fulfill a set of requirements. First it would need to provide the customer with different options to automatically ingest and transform data. Afterwards, a storage medium to allocate all this information had to be implemented so that the information would be accessible by other architectural components, as well as the client (MongoDB Compass provides an intuitive GUI where users can explore and manipulate the different collections, their data, storage space, as well as other information). Last but not least, a data visualization software had to be incorporated in order to actually translate the data gathered and stored in the previous components into actionable intelligence.

This objective was fulfilled and delivered in System B. In fact, the final system architecture can be consulted in Chapter 3.3 and the steps that were performed to setup and implement the solution are described in Chapter 5.

6.2.6 Scenario 6 - Writing a User Manual

After designing, implementing and delivering a solution, it needs to be documented so that the client can be granted a perception of the implementation phase, in order to understand it, manipulate it and possibly expand it in the future.

This objective was fulfilled and delivered in System B. The contents of the user manual provide the following information:

1. The context behind the development of the solution, as well an explanation behind the segregation of dashboards by different target audiences.
2. A description of each of the system's components, how each one was configured and the communication mechanisms used so that each component could communicate with each other.
3. Categorization used in each metric group and a description of each one.
4. Description of every visualization of the dashboard.
5. How the data refresh mechanism was implemented and how it can be adjusted.
6. How to share and perform access management manipulation, in order to provide other users of the organization with read/write permissions to each dashboard.
7. How to revise the threshold parameters of each metric, as well as the risk formulas used to score the different components.

6.2.7 Scenario 7 - Designing a EY Dashboard

As it has been stated, the aim of this thesis is to develop a dashboard which provides organizations with a key situational awareness perception. Logically, this same dashboard can later be reused by EY, for presentation purposes in order to sell this development idea to other interested organizations.

Hence, after the delivery of the solution, a similar mockup dashboard was developed with some interface changes performed. These changes were mainly aimed at the colour palette and the logo. Figure 6.1 provides visibility into the "Top Management's: Monthly Overview"slide of the final product.

This objective was fulfilled and delivered in System B. In fact, EY uses the developed dashboard as a product and is prepared to sell it as a service to other interested clients.

6.2.8 Scenario 8 - Implementing Alert Mechanisms

Dashboards are used so that users can regularly overview a set of key information at a glance. Despite this, it is not common for dashboards to be designed to be consulted with 24/7 regularity. Hence, implementing alert mechanisms which trigger notifications or emails depending on specific conditions, provides an advantage for organizations which want to be constantly aware of its security state at all times.

Despite being a useful feature, it was not implemented in the final product. To explain why, Power BI provides dashboards and reports. As it has been previously described, a report was developed for System B's visualization component. Unfortunately, the Power

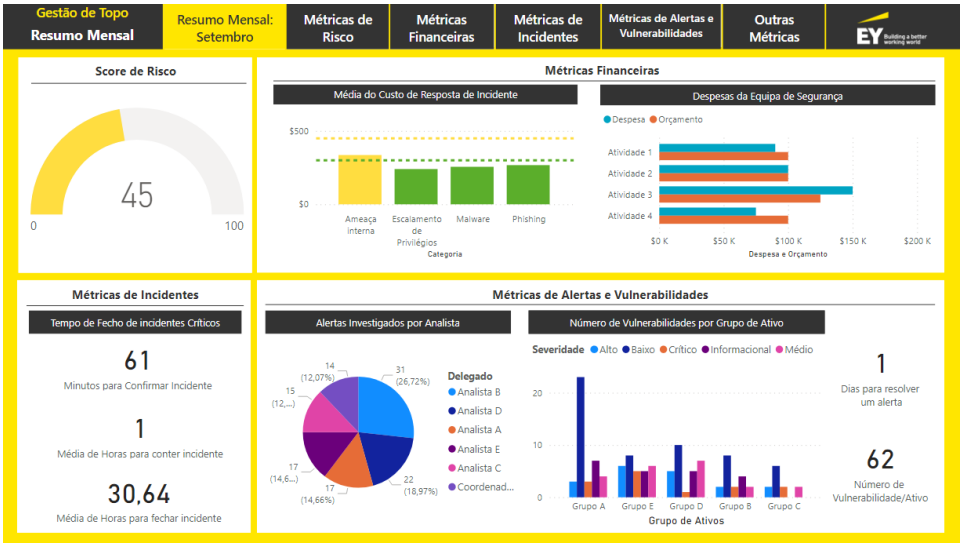


Figure 6.1: EY Top Management Dashboard: Monthly Overview

BI alert mechanism is exclusive to dashboards. Thus, since the report feature of Power BI does not have this mechanism it ended up not being implemented. An alternative way would be to create a mockup dashboard with all the risk metrics of the report and associate alerts for each one. Nonetheless, this feature was neither a priority or even requested by the client, not to mention the student did not have time to complete this procedure on time when the time frame to deliver the solution reached its end.

Finally, it is important to mention that despite not being implemented in System B, this feature was partially implemented on System A. In fact, a function which would be triggered when IOCs with specific risk scores were ingested in the database, was developed. This function sends an email to a specific email account created by the user with metadata related to the incident. An example of such an email, can be consulted in Chapter 4.

6.3 Conclusion

To sum up, ideally System B would have been subjugated to a set of different test scenarios. Nonetheless, since having access to the solution was a requirement of the test cases, a different approach was taken. It was decided that the evaluation process would compare the initial set of contributions that were proposed to be implemented, with the features the solution ended up providing.

All in all, despite not achieving one of the eight initial set of objectives, based on the product evaluation the student objectively claims that the main contributions and goals of System B's situational awareness dashboard have been achieved.

7

Conclusion

The following chapter's structure is divided into three different sections. The first section(7.1) summarizes the contents of this thesis, specifically it states what was developed, why was it needed and verifies if the system ended up meeting the requirements which had been initially defined. This is followed by the second section(7.2), which further complements the first, by listing the obstacles that were encountered during the development phase, in other words, the system's threats to validity. Finally, in the third section(7.3) and after summing up the contents of the thesis and detailing the mains obstacles that were found during its development phase, the chapter is closed with a list of enhancements that could be incorporated in the solution, in order to further refine it.

7.1 Conclusions

The aim of the work developed during this dissertation consisted in the design of a Dashboard which would centralize a set of key information related to the cybersecurity state of an organization. In fact, this dashboard aimed to measure two main concepts, those being the SOC's performance and risk. Hence, by providing such a degree of awareness on a daily basis, this would strengthen the entity's perception onto its main security liabilities, providing a situational awareness archetype.

During the research and development phases that took place while working with EY, two systems were developed. Whereas System A did not evolve into a self sustained and finalized product, System B on the other hand successfully achieved its purpose. In fact, taking into account the initial set of objectives (proposed in Chapter 1) which the solution aimed to attain, System B ended up realizing all its goals but one, successfully carrying out its main purpose. In fact, as described in Chapter 6, from the development of the system architecture which served as the foundation behind the solution, to the design of metrics and visualizations which would intuitively provide viewers with actionable intelligence, only the implementation of alerts mechanisms was not carried through.

Despite its low relevance in regards to the final product, the learning experience that went into implementing System A had a key role into successfully developing System

B. In fact, as described in Chapter 3, from an architectural point of view the student learned a lot by experimenting with different tools which ultimately, helped design what ended up being System B's architecture. Furthermore, System A had a strong Threat Intelligence focus as it collected, parsed and performed a risk evaluation into Indicators of Compromise. This component which provides an external perception of the threat environment surrounding an organization, was not carried through to System B. Ideally, the second system would benefit from this added TI component, as it would strengthen its risk perception.

All in all, based on the product evaluation the student objectively claims that the main contributions and goals of the situational awareness dashboard have been achieved.

7.2 Threats to Validity

This section aims to summarize the mains obstacles that were found during the development of Systems A and B. The list below sums up System A's threats to validity:

- **Expired Contractor Agreement** - Due to the fact that the Contractor Agreement, between *EY* and the client ended, this system never left the mockups/testing phase. Hence, the system could not reach a production stage and ultimately be finalized.

- **Subscription based Data Sources** - Amongst the five different data sources, three of them were subscription based, including the Third-Party Security Rating. This obstacle would inevitably doom any prospects of bringing the system to production, since the client would be required to subscribe to the services. This was a difficult problem to face due to the overall difficulty in finding free Threat Intelligence Feeds which provide rich IOC information.

The list below sums up System B's threats to validity:

- **Time Frame** - During the development phase of System B, there was a tight and strict time frame to comply with. This precise schedule coupled with the fact that the client delivered the data source's access credentials much later than it was promised, delayed the development of the solution. Hence, there was not the possibility to thoroughly test many key components of the dashboard that ideally would be trialed for adjustments, such as, the threshold values and the risk formulas.

- **Unsustainable Metrics** - Another obstacle that was found was a lack of data to build some of the metrics which inevitably made them unsustainable to illustrate. Despite the fact that the KPIs were both agreed upon by EY and the Client, there was missing information that the client could not provide, thus making it impossible to visualize some of them.

7.3 Future Work

Despite complying with most of the initial objectives, there are a lot of improvements which the solution could benefit from for posterity purposes. The list below summarizes some topics which could be integrated in the future:

- **More Metrics** - Despite having a fair amount of metrics, System B's dashboard contains room for improvement in regards to its Financial and Vulnerabilities components. For instance, some KPIs that could have been incorporated in the vulnerability management component are: Time to Detect, Time to Resolve, Time to Mitigate and Time to Patch Vulnerability.

- **Stronger External Component** - Unlike System A, System B is mainly focused on assessing the organization's internal security state, lacking access to an external perspective. Hence, it would only benefit System B to contain System A's Threat Intelligence and Third-Party Security Rating risk components, thus strengthening its risk scoring model.

- **Heavier Dashboard Differentiation** - The segregation that was performed in System B's dashboard experience was a division for three types of users. On the one hand, the Operational Dashboard had certain metrics that were not available for this particular user, justifying its existence. On the other hand, between the CISO and the Top Management Dashboard the key difference was the "Monthly Overview"Panel which ended up not being a heavily justifiable reason to separate both dashboards. Hence, in the future and with the evolution of the system, finding a way to better differentiate both dashboards would be ideal.

- **Over Time Evaluation** - This topic is related to the thresholds and risk formulas which were assigned to each different scoring component. Over time, a revision of those values should be performed in order to calibrate the scale of the scoring system, thus making it as realistic as possible.

- **Risk Model Integration** - System B ended up having a strongly qualitative form of measuring risk. Ideally, in order to have access to a more granular risk perception, the integration of a quantitative risk model, such as FAIR, would be an interesting idea to explore.

Bibliography

[1] L. Ablon. *Data Thieves: The motivations of CyberThreat Actors and Their User and Monetization of Stolen Data*. Retrieved in April 2021. 2018. URL: https://www.rand.org/pubs/testimonies/CT490.html.

[2] T. Alam. *A Reliable Communication Framework and Its Use in Internet of Things (IoT), Engineering and Information Technology (IJSRCSEIT)*. Retrieved in July 2020. 2018. URL: http://ijsrcseit.com/CSEIT1835111..

[3] *Analyst Research: Ponemon Institute Research: Improving the Effectiveness of the SOC*. Retrieved in July 2020. 2019. URL: https://www.devo.com/resources/ponemon-soc-effectiveness-report-2019/.

[4] *ATT&CK™ CONTENT AVAILABLE IN STIX™ 2.0 VIA PUBLIC TAXII™ 2.0 SERVER*. Retrieved in August 2021. URL: https://www.mitre.org/capabilities/cybersecurity/overview/cybersecurity-blog/attck%E2%84%A2-content-available-in-stix%E2%84%A2-20-via.

[5] *Building a Business driven Financial SOC*. Retrieved in July 2020. 2017. URL: https://storage.googleapis.com/stateless-www-cyberbit-com-liv/2017/06/Cyberbit-Building-a-Business-driven-Financial-SOC.pdf.

[6] *Business white paper 5G/SOC: SOC Generations*. Retrieved in July 2020. 2013. URL: http://www.cnmeonline.com/myresources/hpe/docs/HP_ArcSight_WhitePapers_5GSOC_SOC_Generations.PDF.

[7] *Canada COVID-19 Situational Awareness Dashboard*. Retrieved in July 2020. URL: https://experience.arcgis.com/experience/2f1a13ca0b29422f9b34660f0b705043/.

[8] D. Carfagno. *What Is a Security Operations Center, and Why Is It Important?* Retrieved July 2020. 2018. URL: https://www.blackstratus.com/what-is-a-security-operations-center-and-why-is-it-important/.

[9] *CBEST Intelligence-Led Testing Understanding Cyber Threat Intelligence Operations Version 2.0*. Retrieved in July 2020. 2014. URL: https://www.bankofengland.co.uk/-/media/boe/files/financial-stability/financial-sector-continuity/understanding-cyber-threat-intelligence-operations.pdf.

[10] E. Chickowski. *Every Hour SOCs Run, 15 Minutes Are Wasted on False Positives*. Retrieved in July 2020. 2019. URL: https://securityboulevard.com/2019/09/every-hour-socs-run-15-minutes-are-wasted-on-false-positives/.

[11] *CNSS Instruction No. 4009*. Retrieved in July 2020. 2010.

[12] *COVID-19 News: FBI Reports 300 percentage Increase in Reported Cybercrimes*. Retrieved in April 2021. URL: https://www.imcgrupo.com/covid-19-news-fbi-reports-300-increase-in-reported-cybercrimes/.

[13] B. N. Crystal Bedell Mark Bouchard. *Definitive Guide to SOC-as-a-Service: The Essential Elements to Advanced Threat Detection and Response*. Retrieved in July 2020. 2018.

[14] *Cyber Threat and Cyber Threat Actors*. Retrieved in April 2021. 2020. URL: https://cyber.gc.ca/en/guidance/cyber-threat-and-cyber-threat-actors.

[15] *CyCraft Classroom: MITRE ATTCK vs. Cyber Kill Chain*. Retrieved in April 2021. 2020. URL: https://medium.com/cycraft/cycraft-classroom-mitre-att-ck-vs-cyber-kill-chain-vs-diamond-model-1cc8fa49a20f.

[16] J. D. *A Dictionary of Physics*. 6th. Oxford University Press, 2009. ISBN: 9780199233991.

[17] *Elasticseach Architecture: 7 Key Components*. Retrieved in September 2021. URL: https://cloud.netapp.com/blog/cvo-blg-elasticsearch-architecture-7-key-components.

[18] *Elasticsearch: What It Is, How It Works, And What It's Used For*. Retrieved in August 2021. URL: https://www.knowi.com/blog/what-is-elastic-search/.

[19] M. Endsley. *Toward a Theory of Situation Awareness in Dynamic Systems*. Retrieved in July 2020. 1995. URL: https://www.researchgate.net/publication/210198492_Endsley_MR_Toward_a_Theory_of_Situation_Awareness_in_Dynamic_Systems_Human_Factors_Journal_371_32-64.

[20] *ENISA Threat Landscape 2020: Cyber Attacks Becoming More Sophisticated, Targeted, Widespread and Undetected*. Retrieved in April 2021. 2020. URL: https://www.enisa.europa.eu/news/enisa-news/enisa-threat-landscape-2020.

[21] R. Fiedler and M. Stamer. *MITIGATE: Multidimensional, IntegraTed, rIsk assessment framework and dynamic, collaborative risk manaGement tools for critical information infrAstrucTrurEs*. Retrieved in June 2021. 2018. URL: https://searchcio.techtarget.com/definition/OODA-loop.

[22] *Here Are 10 Key Benefits of Business Intelligence Software*. Retrieved in June 2021. 2020. URL: https://www.mastersindatascience.org/learning/benefits-of-business-intelligence/.

[23] *How does security evolve from bolted on to built-in? EY Global Information Security Survey 2020*. Retrieved in July 2020. 2020.

[24] *IBM X-Force Exchange: Overview*. Retrieved in August 2021. URL: https://www.ibm.com/products/xforce-exchange.

[25] J. Jaklič. "Assessing Benefits of Business Intelligence Systems – A Case Study". In: *Management: Journal of Contemporary Management Issues (mbuble@efst.hr); Vol.15 No.1* (Jan. 2008).

[26] N. A. Joseph Muniz and G. McIntyre. *Overview of Security Operations Center Technologies*. Retrieved in July 2020. 2015. URL: https://www.ciscopress.com/articles/article.asp?p=2455014.

[27] R. Lee. *2020 SANS Cyber Threat Intelligence (CTI) Survey*. Retrieved in July 2020. 2020.

[28] S. Lewis. *OODA Loop*. Retrieved in June 2021. 2019. URL: https://searchcio.techtarget.com/definition/OODA-loop.

[29] O. Lindstrom. *Next Generation Security Operations Center*. Retrieved in July 2020. 2018.

[30] *Logstash: Filter Plugins*. Retrieved in August 2021. URL: https://www.elastic.co/guide/en/logstash/current/filter-plugins.html.

[31] *Logstash: Output Plugins*. Retrieved in August 2021. URL: https://www.elastic.co/guide/en/logstash/current/output-plugins.html.

[32] Z. Long et al. *Collecting Indicators of Compromise from Unstructured Text of Cybersecurity Articles using Neural-Based Sequence Labelling*. Retrieved in April 2021. 2019. DOI: 10.1109/IJCNN.2019.8852142.

[33] R. Marty. *Applied Security Visualization*. Pearson Education, Inc, 2008. ISBN: 978-0-321-51010-5. URL: http://www.foo.be/cours/dess-20122013/b/AppliedSecurityVisualization.pdf.

[34] J. McCarthy. *Situational Awareness for Electric Utilities*. Retrieved in July 2020. URL: https://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.1800-7.pdf.

[35] *MISP Official Website*. Retrieved in July 2020. URL: https://www.misp-project.org/features.html.

[36] *MISP: Malware Information Sharing Platform*. Retrieved in July 2020. URL: https://github.com/MISP/misp-bookl.

[37] *MITRE ATT&CK*. Retrieved in April 2021. URL: https://attack.mitre.org.

[38] *MITRE ATT&CK FAQS: What is the relationship between ATTCK and the Lockheed Martin Cyber Kill Chain®?* Retrieved in May 2021. 2021. URL: https://attack.mitre.org/resources/faq/.

[39] *MITRE CVE: Frequantly Asked Questions*. Retrieved in May 2021. 2021. URL: https://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-150.pdf.

[40] *Modern Bank Heists 3.0*. Retrieved in April 2021. URL: https://www.vmware.com/resources/security/modern-bank-heists-2020.html.

[41] J. Moran. *Key Performance Indicators (KPIs) for Security Operations and Incident Response*. Retrieved in July 2020. URL: https://www.dflabs.com/wp-content/uploads/2018/03/KPIs_for_Security_Operations_and_Incident_Response-2.pdf.

[42] *NIST Special Publication 800-150: Guide to Cyber Threat Information Sharing*. Retrieved in May 2021. 2016. URL: https://attack.mitre.org/resources/faq/.

[43] *NIST Special Publication 800-30 Revision 1 Guide for Conducting Risk Assessments*. Retrieved in July 2020. 2012. URL: https://nvlpubs.nist.gov/nistpubs/Legacy/SP/nistspecialpublication800-30r1.pdf.

[44] *NISTIR 8286 Integrating Cybersecurity and Enterprise Risk Management (ERM)*. Retrieved in April 2021. 2020. URL: https://doi.org/10.6028/NIST.IR.8286.

[45] *OASIS Cyber Threat Intelligence (CTI): Introduction to STIX*. Retrieved in July 2020. URL: https://oasis-open.github.io/cti-documentation/stix/intro.

[46] *OASIS Cyber Threat Intelligence (CTI): Introduction to TAXII*. Retrieved in July 2020. URL: https://oasis-open.github.io/cti-documentation/taxii/intro.

[47] *OTX Alienvault: Documentation*. Retrieved in August 2021. URL: https://cybersecurity.att.com/documentation/usm-appliance/otx/about-otx.htm.

[48] T. Paul Cichonskim Tom Millar and K. Scarfone. *NIST Special Publication 800-61 Revision 2: Computer Security Incident Handling Guide*. Retrieved in July 2020. 2012. URL: http://dx.doi.org/10.6028/NIST.SP.800-61r2.

[49] *Portugal COVID-19 Situational Awareness Dashboard*. Retrieved in July 2020. URL: https://covid19.min-saude.pt/ponto-de-situacao-atual-em-portugal/.

[50] V. Prenosil and I. Ghafir. "Advanced Persistent Threat Attack Detection". In: *International Journal of Advancements in Computer Networks and Its Security-IJCNS* 4.4 (2014). URL: https://www.researchgate.net/publication/305956804.

[51] *Rapid7 InsightVM*. Retrieved in August 2021. URL: https://www.rapid7.com/products/insightvm/.

[52] R. Ruefle. *Defining Computer Security Incident Response Teams*. Retrieved in July 2020. 2007. URL: https://resources.sei.cmu.edu/asset_files/WhitePaper/2007_019_001_294579.pdf.

[53] *Security Scorecard: Security Ratings*. Retrieved in August 2021. URL: https://securityscorecard.com/product/security-ratings.

[54] F. Shi. *Threat spotlight: Coronavirus-related phishing*. Retrieved in April 2021. 2020. URL: https://blog.barracuda.com/2020/03/26/threat-spotlight-coronavirus-related-phishing/.

[55] S. Sławińska. *Cyber Kill Chain – what is it and how to use it to stop advanced methods of attack?* Retrieved in May 2021. 2020. URL: https://seqred.pl/en/cyber-kill-chain-what-is-it-and-how-to-use-it-to-stop-advanced-methods-of-attack/.

[56] *SOC Analysts Quitting Over Burnout, Lack of Visibility*. Retrieved in July 2020. URL: https://www.channelfutures.com/mssp-insider/soc-analysts-quitting-over-burnout-lack-of-visibility.

[57] *Systems Hardening*. Retrieved in August 2021. URL: https://www.beyondtrust.com/resources/glossary/systems-hardening.

[58] *The Cyber Kill Chain*. Retrieved in April 2021. URL: https://www.lockheedmartin.com/en-us/capabilities/cyber/cyber-kill-chain.html.

[59] *The Economics of Security Operations Centers*. Retrieved in July 2020. 2020. URL: https://respond-software.com/resources/reports-ebooks/ponemon-institute-respond-software-2/?utm_source=Survey&utm_medium=press%5C%20release&utm_campaign=Ponemon%5C%20Report.

[60] *The ELK Stack: Logstash*. Retrieved in August 2021. URL: https://aws.amazon.com/pt/elasticsearch-service/the-elk-stack/logstash.

[61] *The Importance of Building a Security Operations Center*. Retrieved in July 2020. URL: https://www.mcafee.com/enterprise/en-us/security-awareness/operations/building-a-soc.html.

[62] *The Importance of Cyber Risk Management*. Retrieved in April 2021. URL: https://www.csriskmanagement.co.uk/the-importance-of-cyber-risk-management/.

[63] *The Modern Security Operations Center, SecOps and SIEM: How They Work Together*. Retrieved in July 2020. URL: https://www.exabeam.com/siem-guide/the-soc-secops-and-siem/.

[64] *The Pyramid of Pain*. Retrieved in April 2021. URL: https://www.netsurion.com/articles/the-pyramid-of-pain.

[65] *The security operations center (SOC) team: operations responsibilities*. Retrieved in July 2020. URL: https://cybersecurity.att.com/solutions/security-operations-center/building-a-soc/soc-team.

[66] *TheHive Project*. Retrieved in August 2021. URL: http://thehive-project.org.

[67] A. T. Tunggal. *What Is Cybersecurity Risk? A Thorough Definition*. Retrieved in April 2021. 2021. URL: https://www.upguard.com/blog/cybersecurity-risk.

[68] M. Vielberth et al. "Security Operations Center: A Systematic Study and Open Challenges". In: *IEEE Access* PP (Dec. 2020). DOI: 10.1109/ACCESS.2020.3045514.

[69] *What is a REST API?* Retrieved in August 2021. URL: https://www.redhat.com/en/topics/api/what-is-a-rest-api.

[70] *What is an ODBC Driver?* Retrieved in August 2021. URL: https://www.progress.com/faqs/datadirect-odbc-faqs/what-is-an-odbc-driver.

[71] *What is Cybercrime?* Retrieved in February 2021. URL: https://www.kaspersky.com/resource-center/threats/what-is-cybercrime.

[72] *What is DSN (Data Source Name)?* Retrieved in August 2021. URL: https://support.microsoft.com/en-us/topic/what-is-a-dsn-data-source-name-ae9a0c76-22fc-8a30-606e-2436fe26e89f.

[73] *What Is Endpoint Detection and Response (EDR)?* Retrieved in August 2021. URL: https://www.mcafee.com/enterprise/en-us/security-awareness/endpoint/what-is-endpoint-detection-and-response.html.

[74] *What is NoSQL?* Retrieved in August 2021. URL: https://www.mongodb.com/nosql-explained.

[75] B. Wieder and M.-L. Ossimitz. "The Impact of Business Intelligence on the Quality of Decision Making – A Mediation Model". In: *Procedia Computer Science* 64 (Dec. 2015), pp. 1163–1171. DOI: 10.1016/j.procs.2015.08.599.

[76] *Worm:W32/Slammer*. Retrieved in July 2020. URL: https://www.f-secure.com/v-descs/mssqlm.shtml.

[77] *X-Force Threat Intelligence Index 2020*. Retrieved in July 2020.

[78] C. Zimmerman. *Ten Strategies of a World-Class Cybersecurity Operations Center*. Retrieved in July 2020. 2014. URL: https://www.mitre.org/sites/default/files/publications/pr-13-1028-mitre-10-strategies-cyber-ops-center.pdf.

I

System A: Dashboard Visualizations

Figure I.1: System A Dashboard: Overview

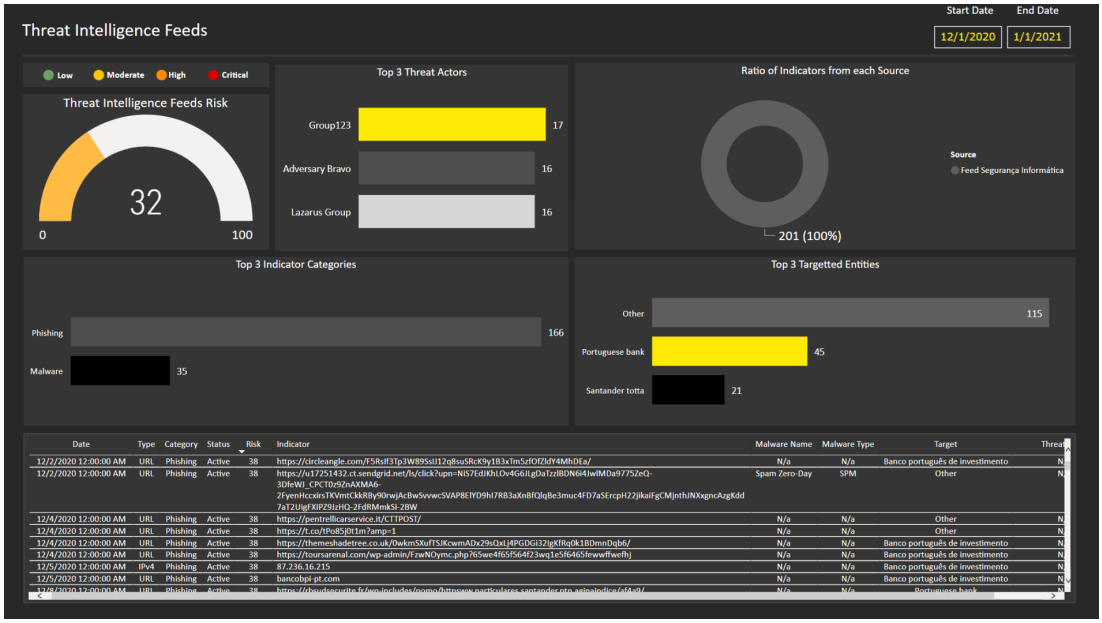


Figure I.2: System A Dashboard: Threat Intelligence

Figure I.3: System A Dashboard: Internal

Figure I.4: System A Dashboard: Third-Party Rating

II

System B: Dashboard Visualizations

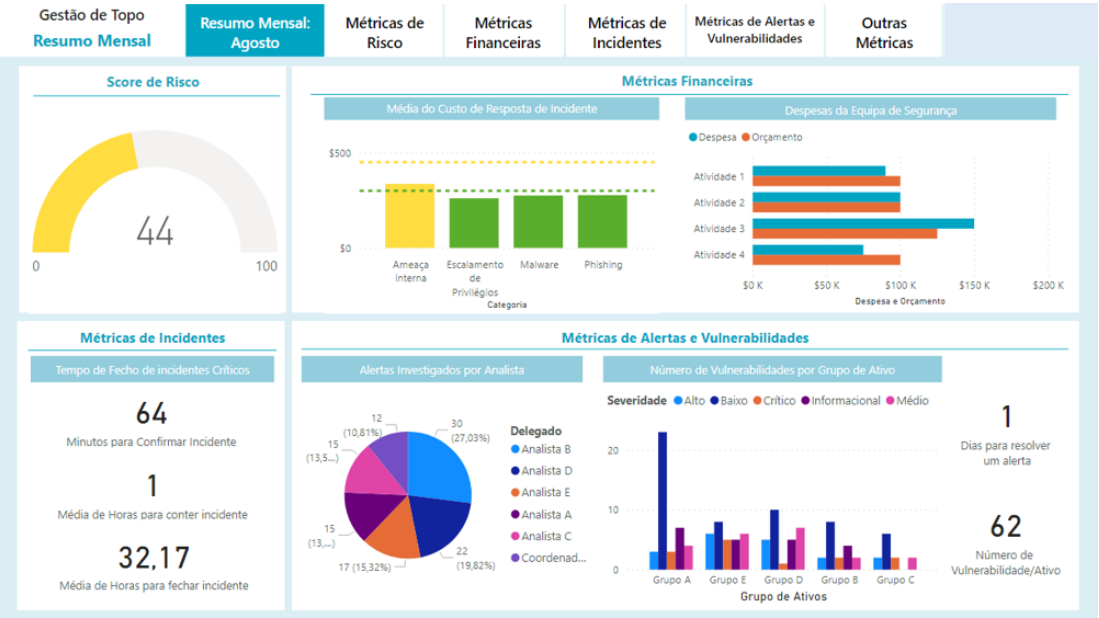


Figure II.1: Top Management Dashboard: Monthly Overview

Figure II.2: Top Management Dashboard: Global Risk

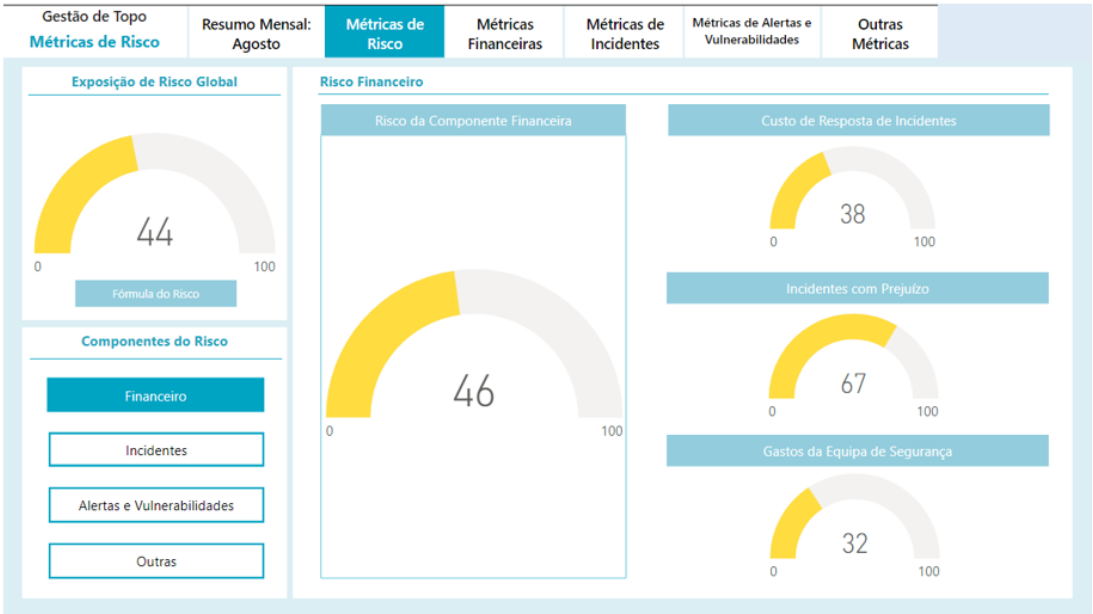


Figure II.3: Top Management Dashboard: Financial Risk

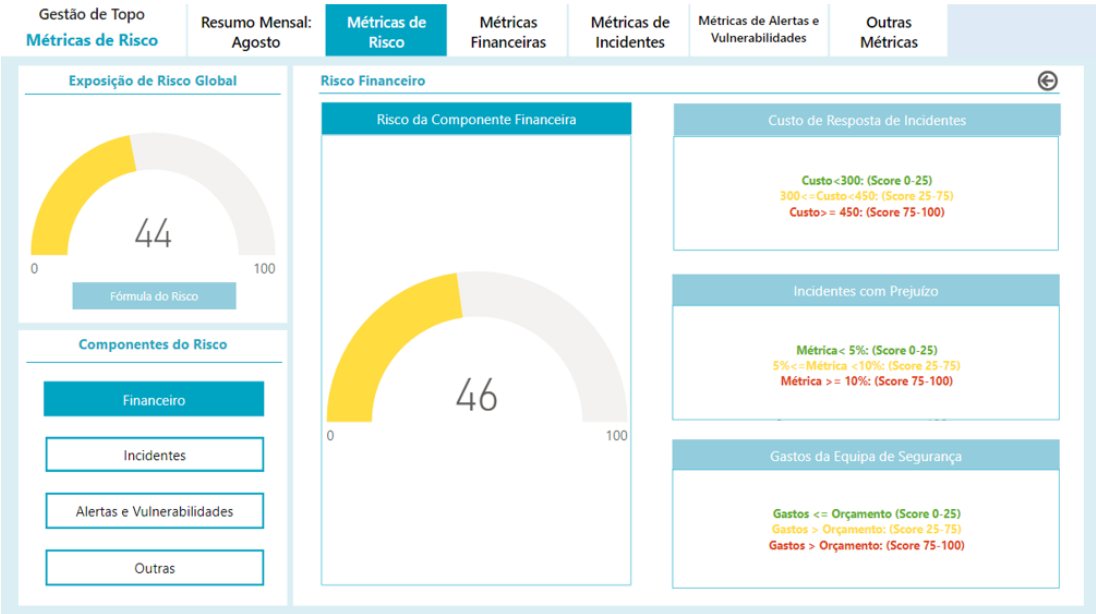


Figure II.4: Top Management Dashboard: Financial Risk (Thresholds)

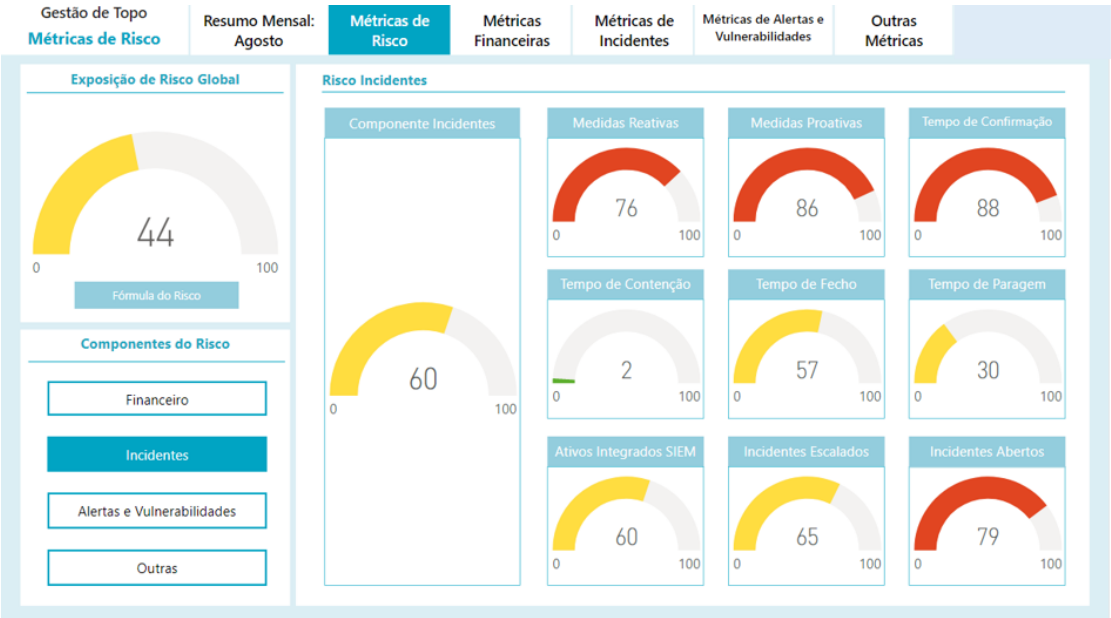


Figure II.5: Top Management Dashboard: Incidents Risk

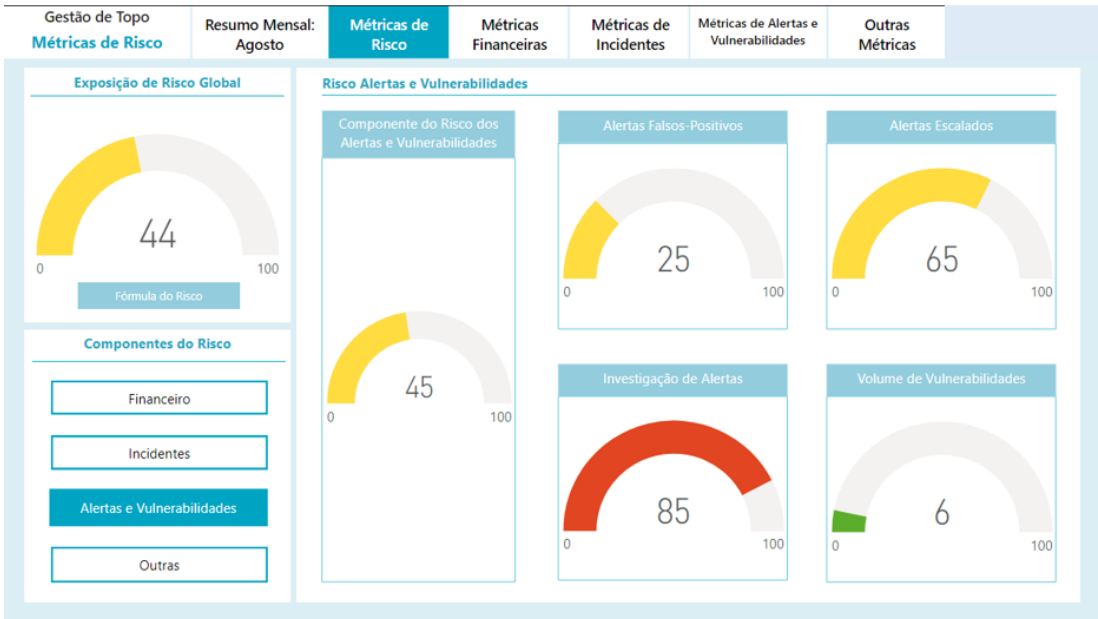


Figure II.6: Top Management Dashboard: Alerts and Vulnerabilities Risk

Figure II.7: Top Management Dashboard: Other Risk

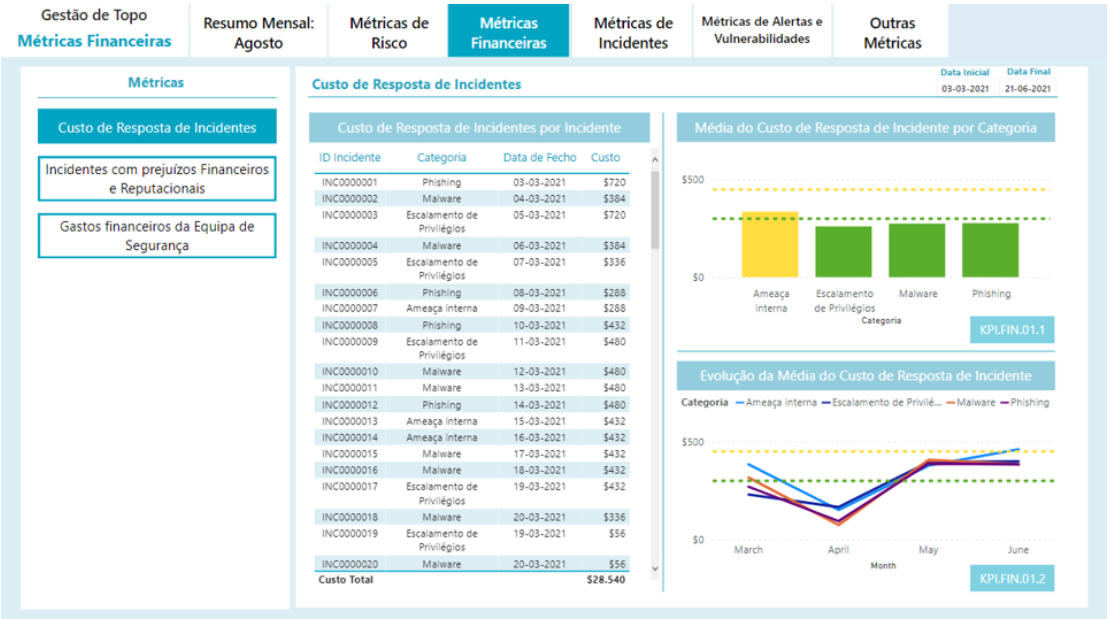


Figure II.8: Top Management Dashboard: Incident Financial Estimation

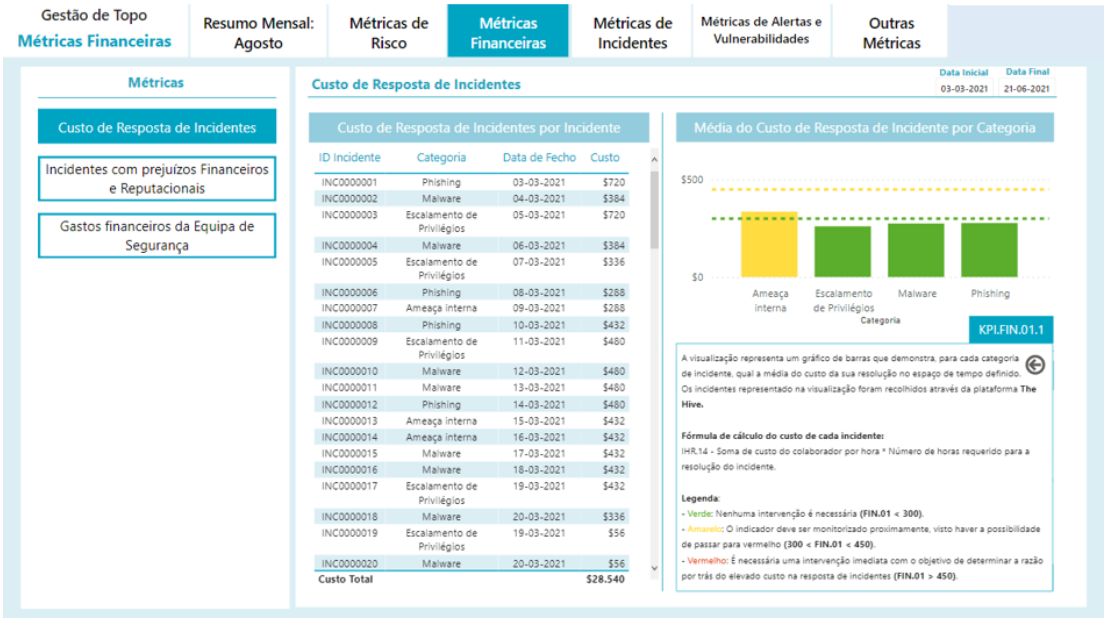


Figure II.9: Top Management Dashboard: Incident Financial Estimation (KPI.FIN.01.1 Description)

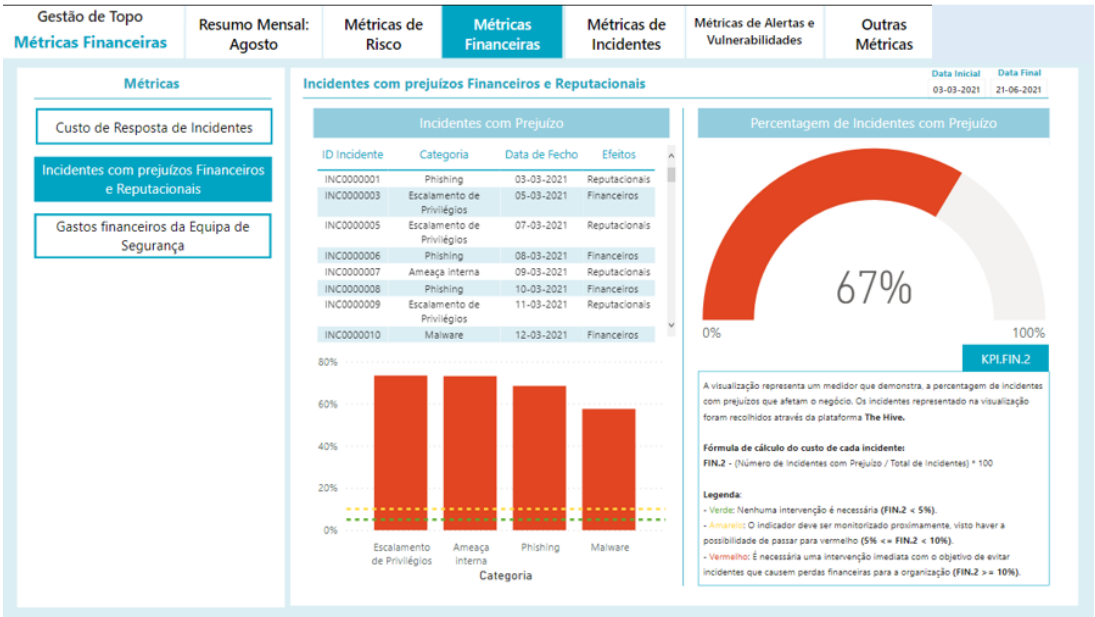


Figure II.10: Top Management Dashboard: Financial and Reputational Costs of Security Incidents

Figure II.11: Top Management Dashboard: Expenses of the Security Team

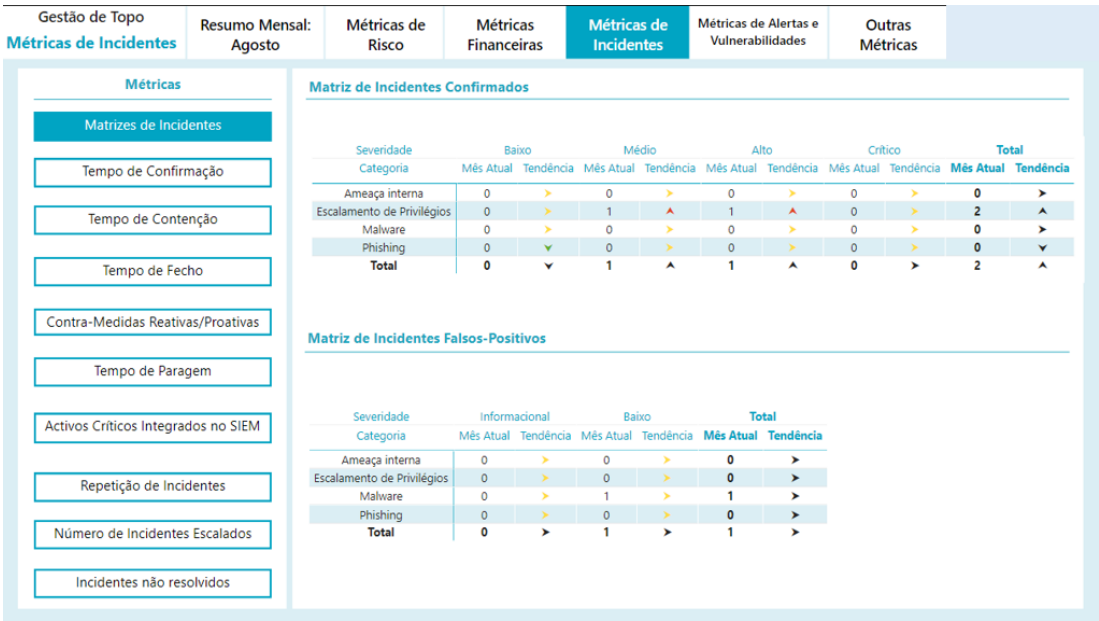


Figure II.12: Top Management Dashboard: Confirmed Incidents Matrix and False Positives Matrix

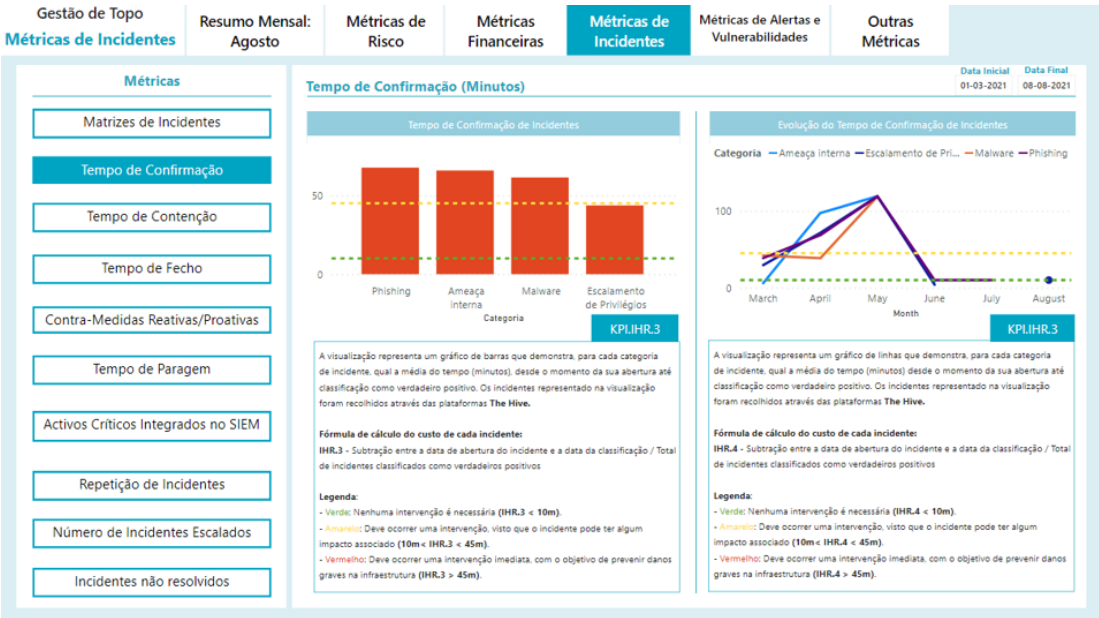


Figure II.13: Top Management Dashboard: Time to Confirmation

Figure II.14: Top Management Dashboard: Time to Containment

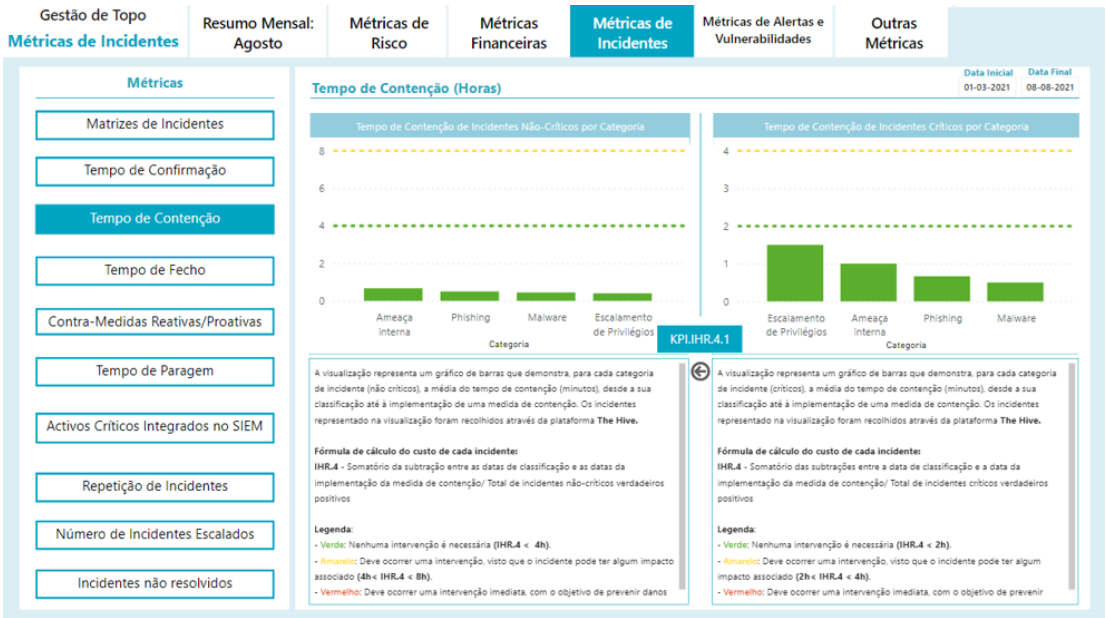


Figure II.15: Top Management Dashboard: Time to Containment (Description Boxes)

Figure II.16: Top Management Dashboard: Time to Closure

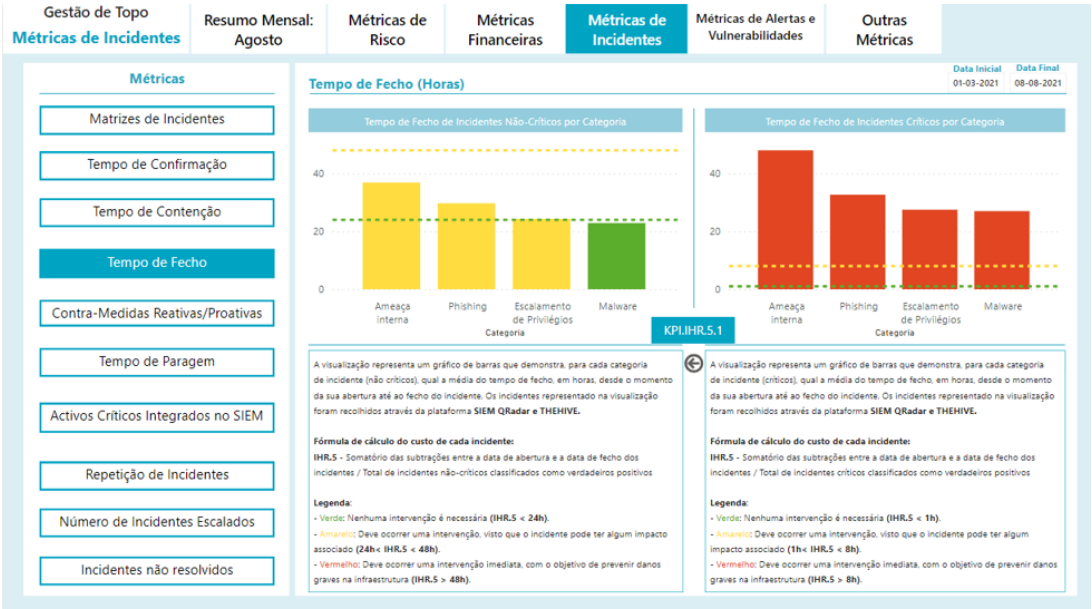


Figure II.17: Top Management Dashboard: Time to Closure (Description Boxes)

Figure II.18: Top Management Dashboard: Reactive and Proactive Counter Measures

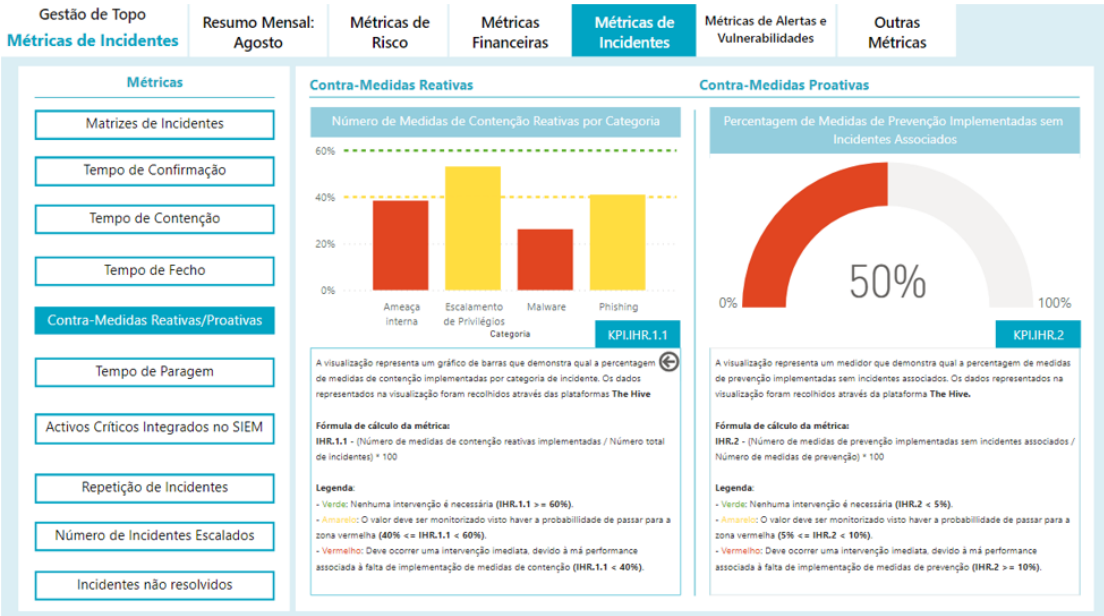


Figure II.19: Top Management Dashboard: Reactive and Proactive Counter Measures (Description)

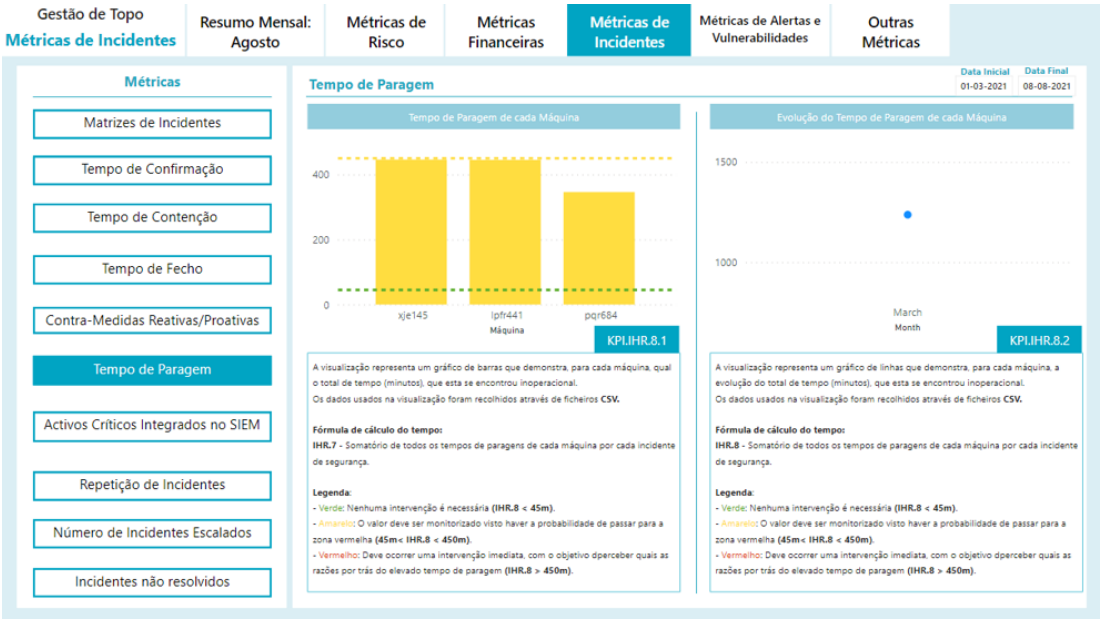


Figure II.20: Top Management Dashboard: Unscheduled Downtime

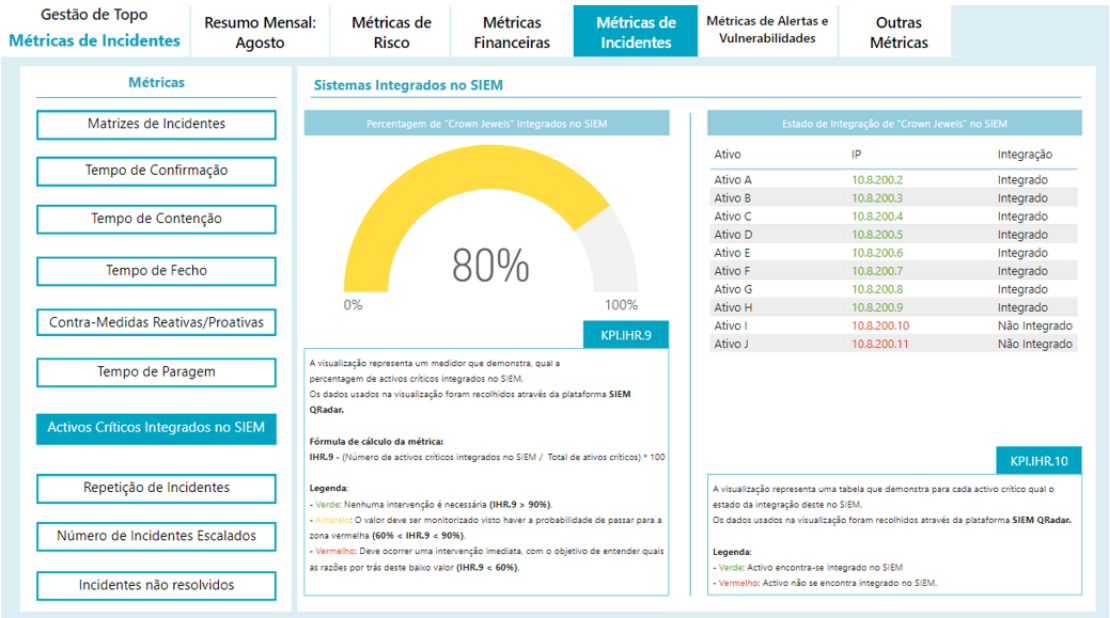


Figure II.21: Top Management Dashboard: Critical Assets Integrated in the SIEM

Gestão de Topo
Métricas de Incidentes

Resumo Mensal: Agosto | Métricas de Risco | Métricas Financeiras | Métricas de Incidentes | Métricas de Alertas e Vulnerabilidades | Outras Métricas

Métricas

- Matrizes de Incidentes
- Tempo de Confirmação
- Tempo de Contenção
- Tempo de Fecho
- Contra-Medidas Reativas/Proativas
- Tempo de Paragem
- Activos Críticos Integrados no SIEM
- Repetição de Incidentes
- Número de Incidentes Escalados
- Incidentes não resolvidos

Incidentes repetidos com a mesma Categoria nos mesmos Sistemas

OS Máquina	Linux		Windows		Total	
Categoria	Mês Atual	Tendência	Mês Atual	Tendência	Mês Atual	Tendência
Ameaça interna	0	➢	0	⌄	0	⌄
Escalamento de Privilégios	0	➢	2	⌃	2	⌃
Malware	0	⌄	1	⌄	1	⌄
Phishing	0	⌄	0	⌄	0	⌄
Total	**0**	⌄	**3**	⌄	**3**	⌄

KPI.IHR.11

A visualização representa uma matriz que cruza a categoria dos incidentes com o sistema operativo da máquina em que estes ocorreram. Desta forma, dois dados podem ser extraídos, o número de incidentes do mês atual para cada categoria de incidente em cada sistema operativo diferente e a "trend" desse valor, ou seja, comparativamente ao mês passado se o valor aumentou, diminuiu ou se se manteve igual.
Os dados usados na visualização foram recolhidos através da plataforma de incidentes **The Hive.**

Fórmula de cálculo da trend:
IHR.11 -Número de incidentes do mês atual - Número de incidentes do mês anterior

Legenda:
- Verde: Ocorreu uma diminuição no número de incidentes face ao mês anterior **(IHR.11 < 0)**.
- Amarelo: O número de incidentes do mês atual face ao do mês anterior manteve-se **(IHR.11 = 0)**.
- Vermelho: Deve ocorrer uma intervenção imediata, com o objetivo de entender quais as razões por trás do aumento do número de incidentes face ao mês anterior **(IHR.11 > 0)**.

Figure II.22: Top Management Dashboard: Repeated Incidents

Gestão de Topo
Métricas de Incidentes

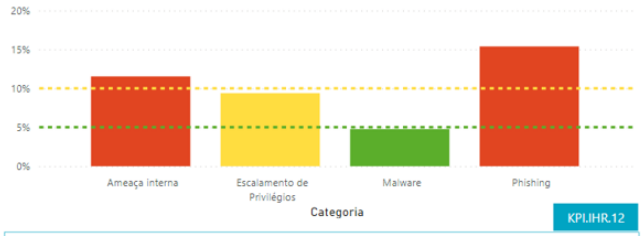
Resumo Mensal: Agosto | Métricas de Risco | Métricas Financeiras | Métricas de Incidentes | Métricas de Alertas e Vulnerabilidades | Outras Métricas

Métricas

- Matrizes de Incidentes
- Tempo de Confirmação
- Tempo de Contenção
- Tempo de Fecho
- Contra-Medidas Reativas/Proativas
- Tempo de Paragem
- Activos Críticos Integrados no SIEM
- Repetição de Incidentes
- Número de Incidentes Escalados
- Incidentes não resolvidos

Incidentes Escalados para o Coordenador do SOC

Data Inicial 01-03-2021 | Data Final 08-08-2021



KPI.IHR.12

A visualização representa um gráfico de barras que identifica para cada categoria de incidente qual a percentagem que teve de ser escalada para o Coordenador do SOC.
Os dados usados na visualização foram recolhidos através da plataforma **The Hive.**

Fórmula de cálculo da métrica:
IHR.12 - (Número de incidentes escalados da categoria / Número de incidentes total da categoria) * 100

Legenda:
- Verde: Nenhuma intervenção é requerida **(IHR.12 < 5%)**.
- Amarelo: O valor deve ser monitorizado visto haver a probabilidade de passar para a zona vermelha(**5% <= IHR.12 < 10%**).
- Vermelho: Deve ocorrer uma intervenção imediata, com o objetivo de entender quais as razões por trás do aumento do escalamento de incidentes para o Coordenador do SOC **(IHR.12 >= 10%)**.

Figure II.23: Top Management Dashboard: Escalated Incidents

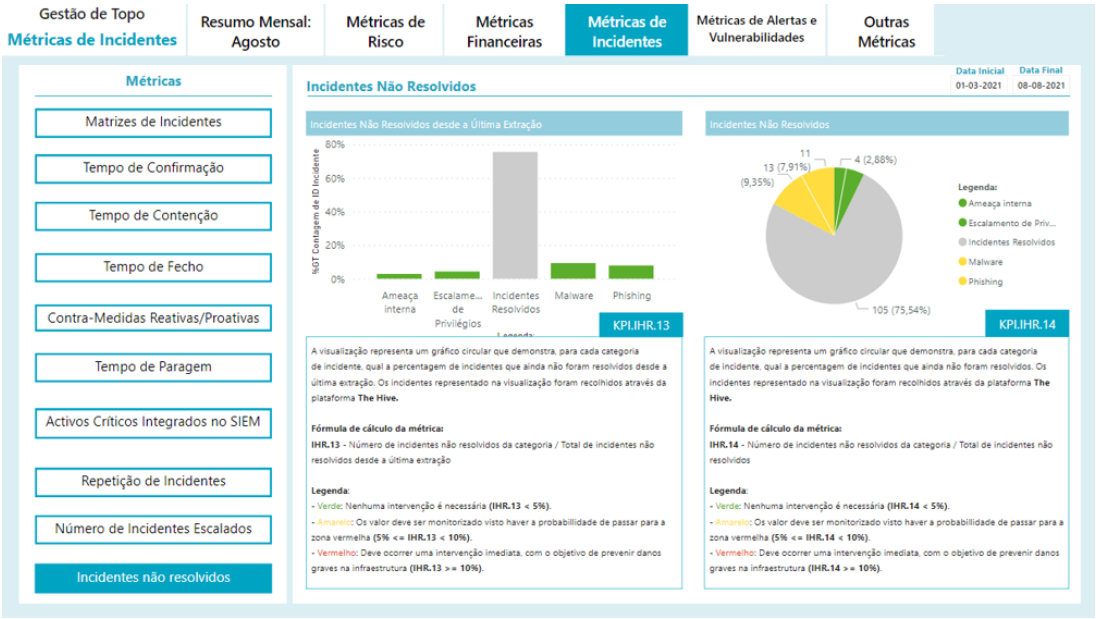


Figure II.24: Top Management Dashboard: Unresolved Incidents

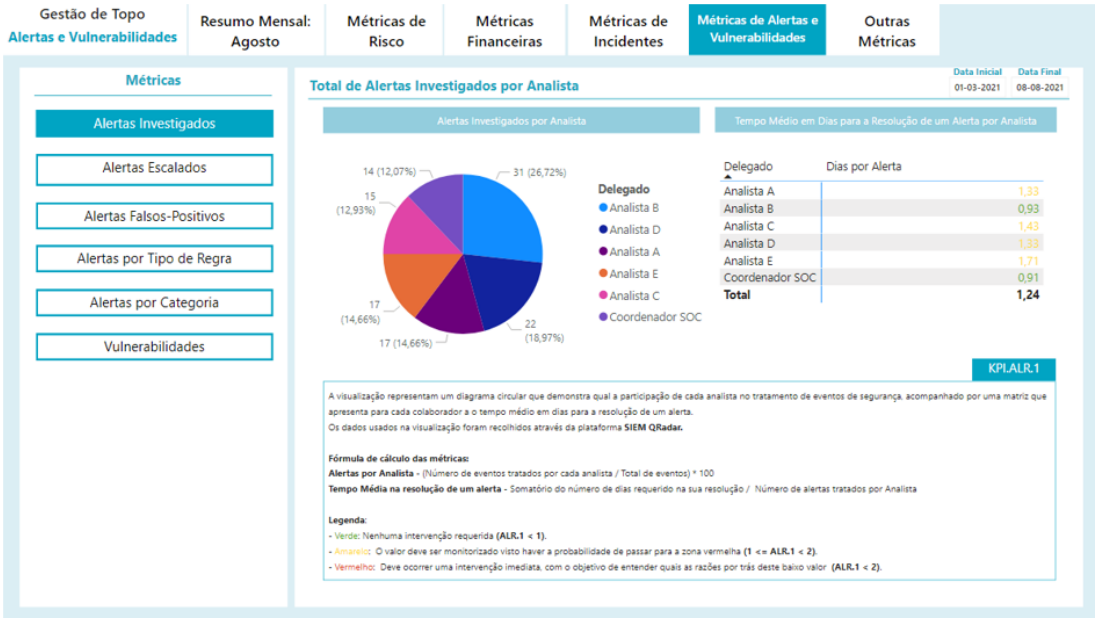


Figure II.25: Top Management Dashboard: Alerts Investigated per Analyst

Figure II.26: Top Management Dashboard: Escalated Alerts

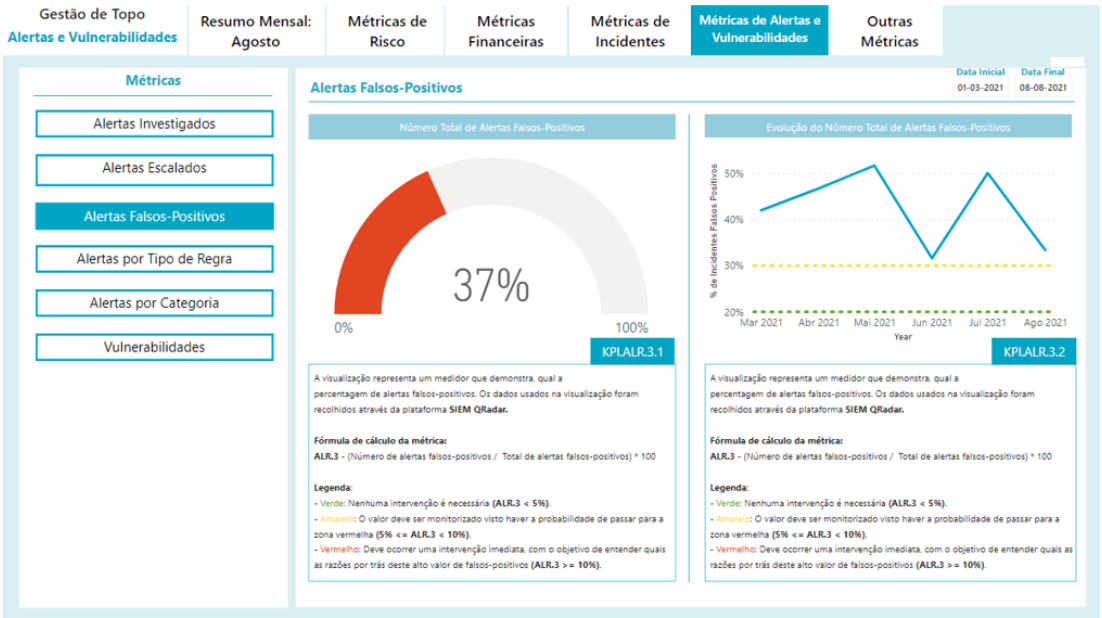


Figure II.27: Top Management Dashboard: False-Positive Alerts

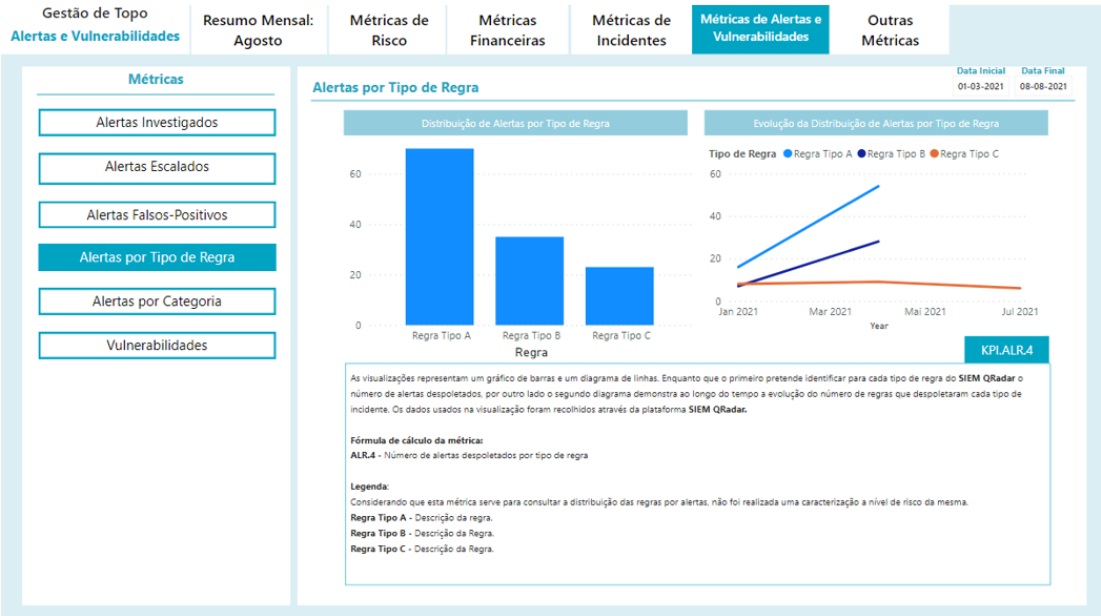


Figure II.28: Top Management Dashboard: Alerts Distribution by Rule

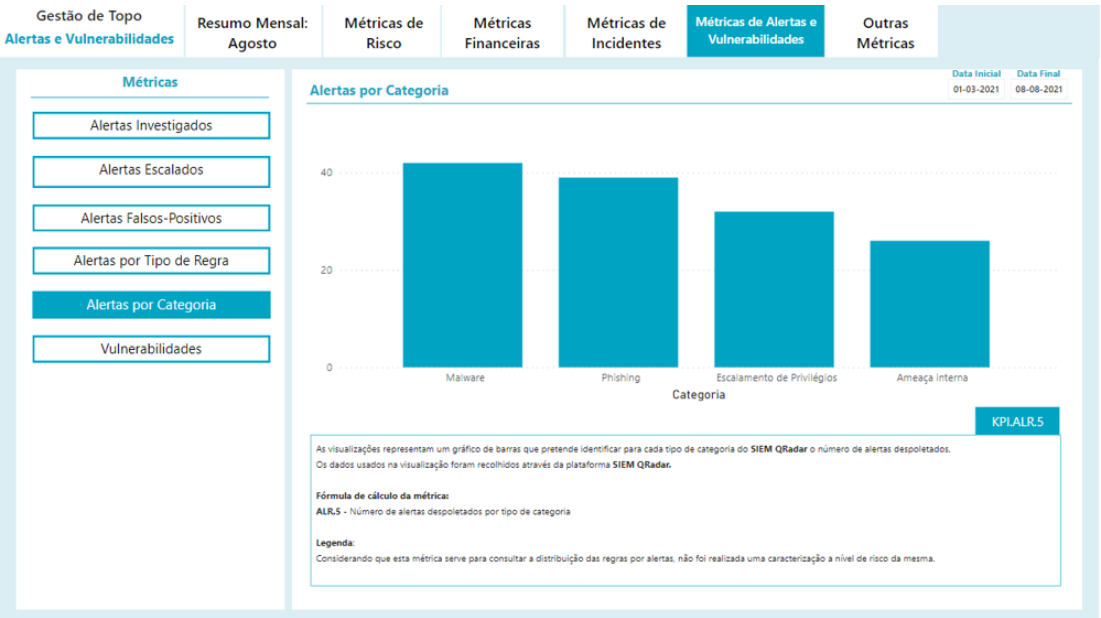


Figure II.29: Top Management Dashboard: Alerts Distribution by Category

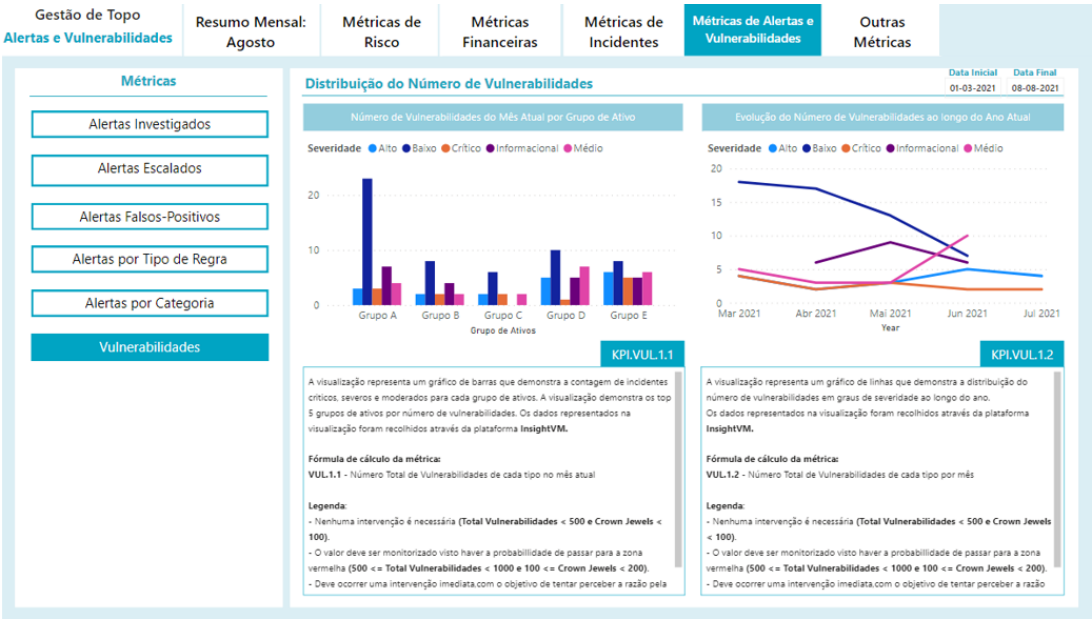


Figure II.30: Top Management Dashboard: Vulnerabilities

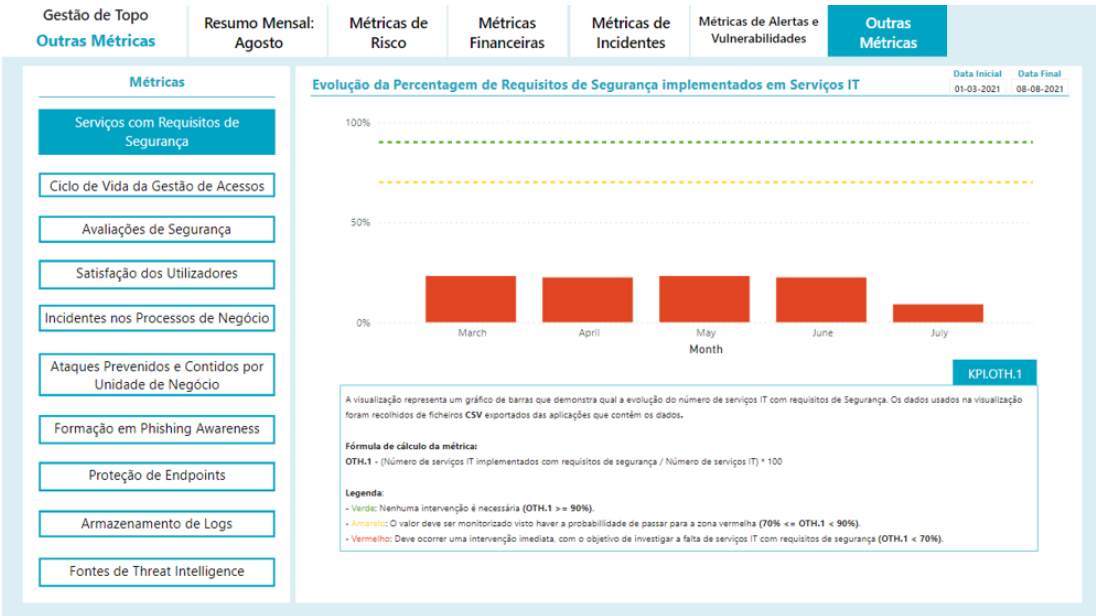


Figure II.31: Top Management Dashboard: IT Services with Security Requirements

Figure II.32: Top Management Dashboard: Access Management

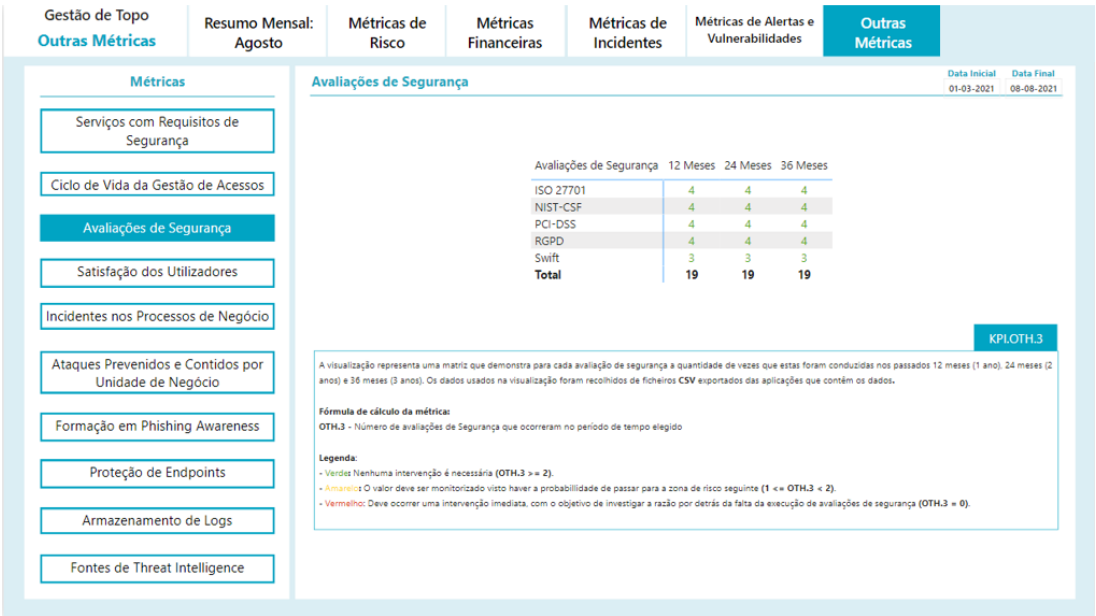


Figure II.33: Top Management Dashboard: Security Assessments

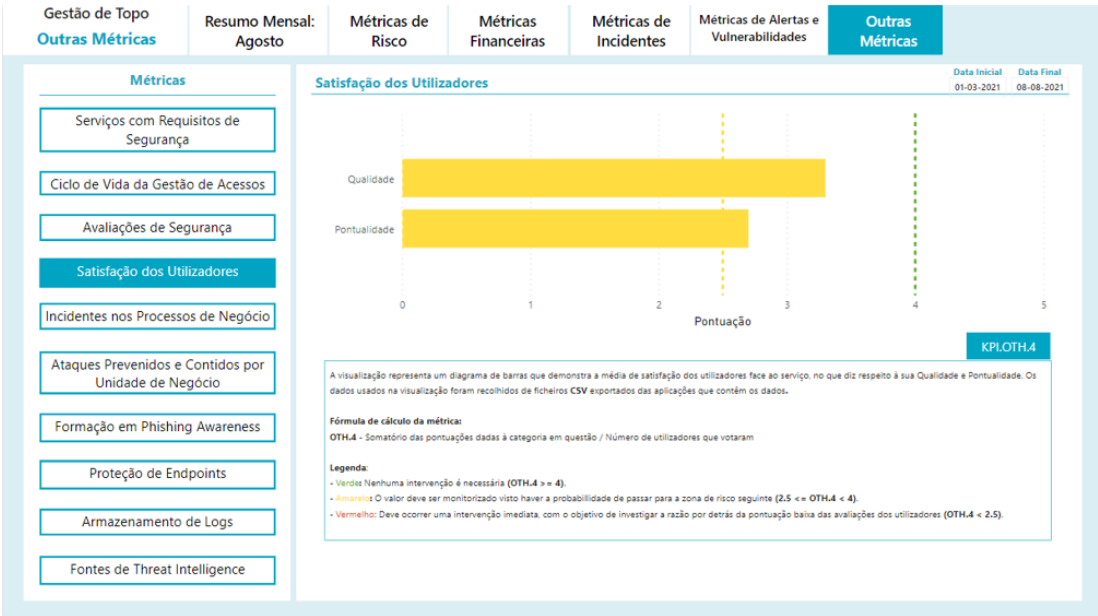


Figure II.34: Top Management Dashboard: User Satisfaction

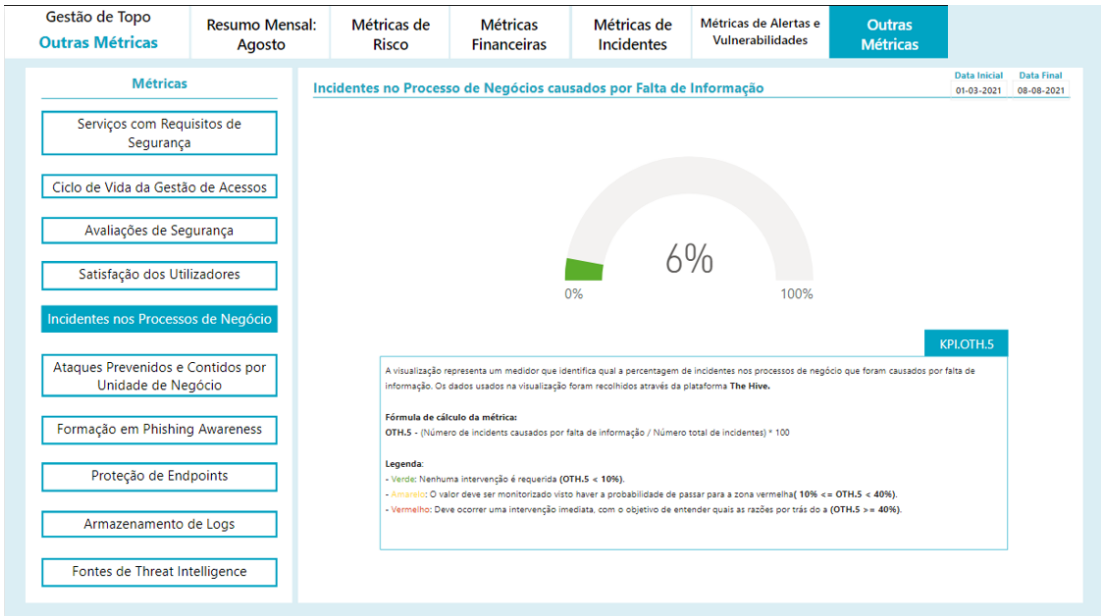


Figure II.35: Top Management Dashboard: Business Process Incidents

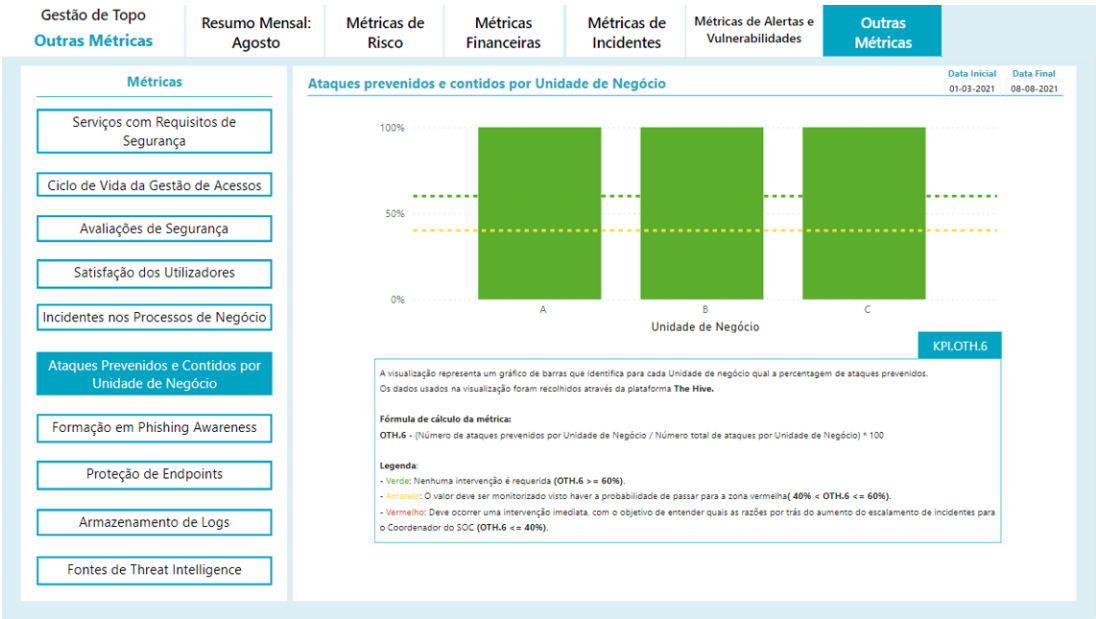


Figure II.36: Top Management Dashboard: Prevented Attacks by Business Unit

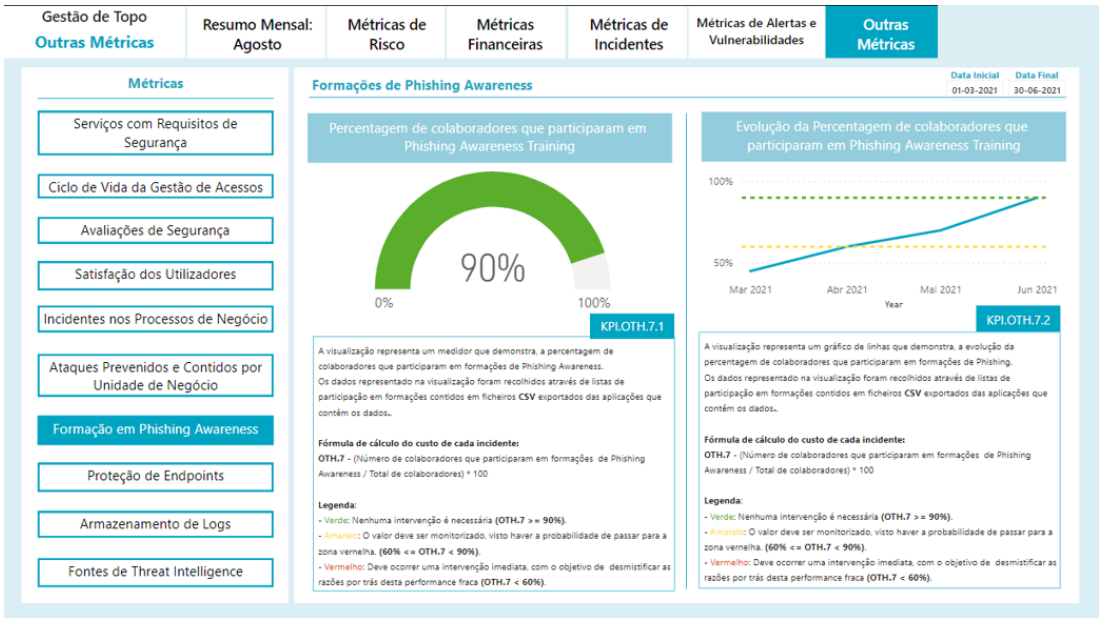


Figure II.37: Top Management Dashboard: Phishing Awareness Training

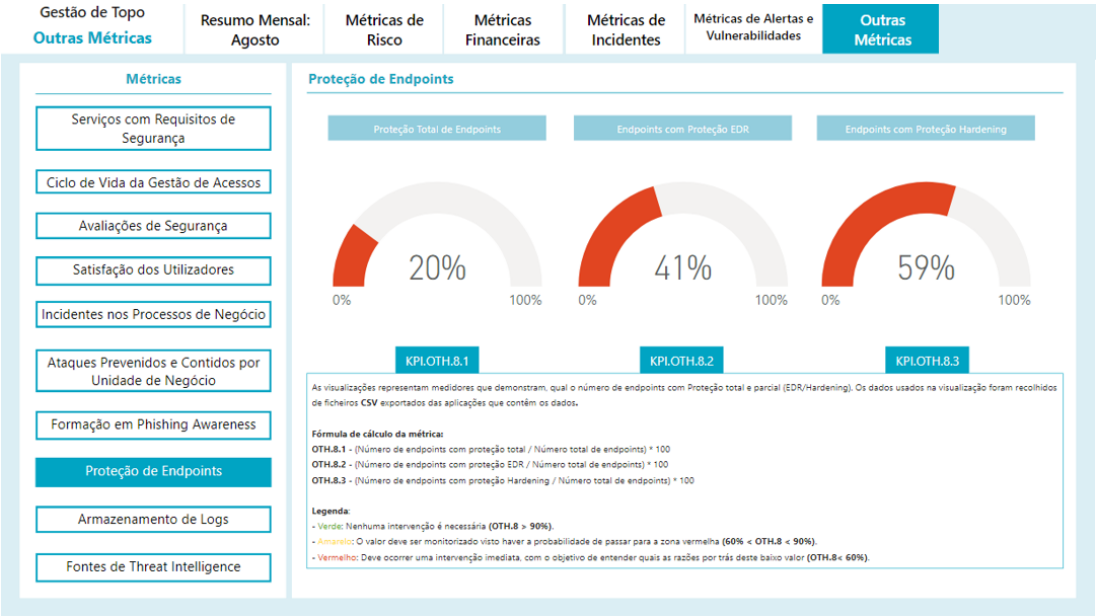


Figure II.38: Top Management Dashboard: Endpoint Security

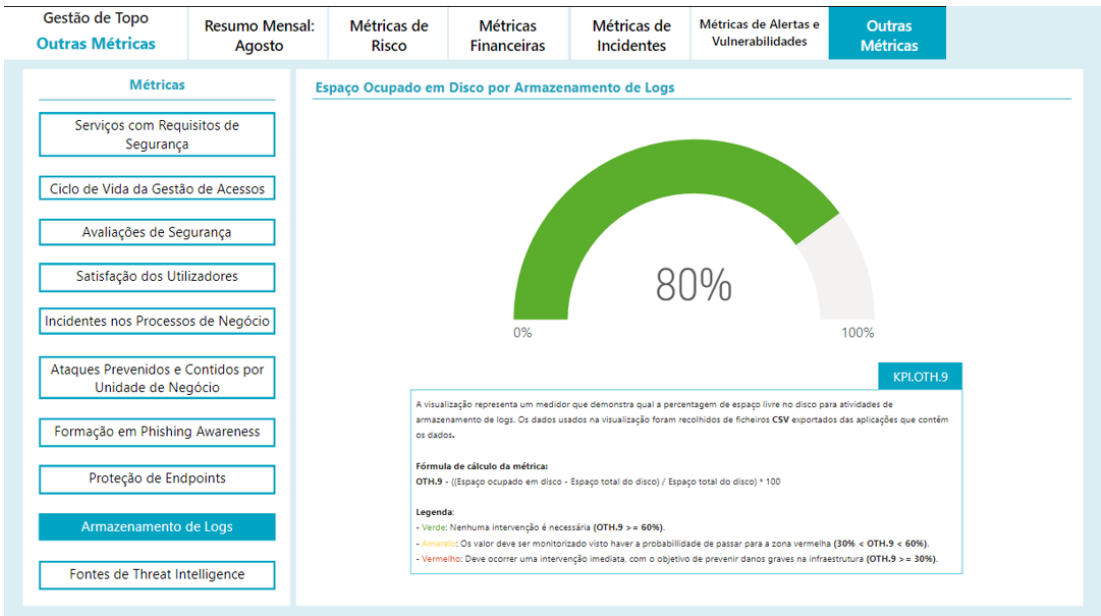


Figure II.39: Top Management Dashboard: Available Storage Log Management Tool

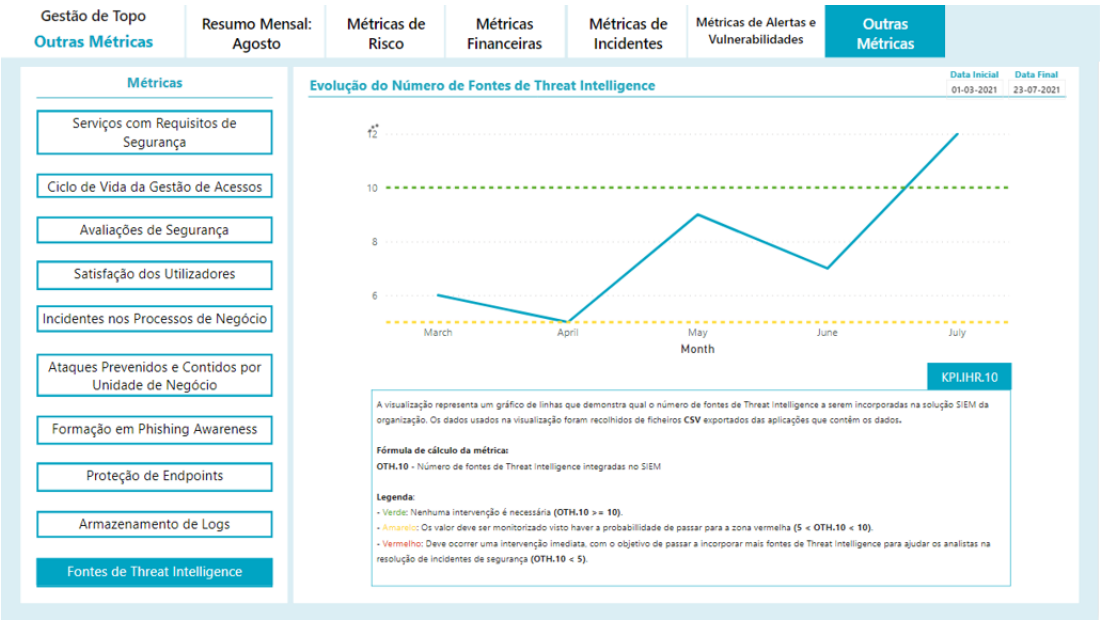


Figure II.40: Top Management Dashboard: Threat Intelligence Sources