8

A Cybersecurity Situational Awareness and Information-sharing Solution for Local Public Administrations Based on Advanced Big Data Analysis: The CS-AWARE Project

Thomas Schaberreiter[1], Juha Röning[2], Gerald Quirchmayr[1], Veronika Kupfersberger[1], Chris Wills[3], Matteo Bregonzio[4], Adamantios Koumpis[5], Juliano Efson Sales[5], Laurentiu Vasiliu[6], Kim Gammelgaard[7], Alexandros Papanikolaou[8], Konstantinos Rantos[9] and Arnolt Spyros[8]

[1]University of Vienna – Faculty of Computer Science, Austria
[2]University of Oulu – Faculty of Information Technology and Electrical Engineering, Finland
[3]CARIS Research Ltd., United Kingdom
[4]3rd Place, Italy
[5]University of Passau, Germany
[6]Peracton, Ireland
[7]RheaSoft, Denmark
[8]InnoSec, Greece
[9]Eastern Macedonia and Thrace Institute of Technology,
Department of Computer and Informatics Engineering, Greece
E-mail: thomas.schaberreiter@univie.ac.at; juha.röning@oulu.fi;
gerald.quirchmayr@univie.ac.at; veronika.kupfersberger@univie.ac.at;
ccwills@carisresearch.co.uk; matteo.bregonzio@3rdplace.com;
adamantios.koumpis@uni-passau.de; juliano-sales@uni-passau.de;
laurentiu.vasiliu@peracton.com; kim@rheasoft.dk;
a.papanikolaou@innosec.gr; krantos@teiemt.gr; a.spyros@innosec.gr

In this chapter, the EU-H2020 project CS-AWARE (running from 2017 to 2020) is presented. CS-AWARE proposes a cybersecurity awareness solution for local public administrations (LPAs) in line with the currently developing European legislatory cybersecurity framework. CS-AWARE aims to increase the automation of cybersecurity awareness approaches, by collecting cybersecurity relevant information from sources both inside and outside of monitored LPA systems, performing advanced big data analysis to set this information in context for detecting and classifying threats and to detect relevant mitigation or prevention strategies. CS-AWARE aims to advance the function of a classical decision support system by enabling supervised system self-healing in cases where clear mitigation or prevention strategies for a specific threat could be detected. One of the key aspects of the European cybersecurity strategy is a cooperative and collaborative approach towards cybersecurity. CS-AWARE is built around this concept and relies on cybersecurity information being shared by relevant authorities in order to enhance awareness capabilities. At the same time, CS-AWARE enables system operators to share incidents with relevant authorities to help protect the larger community from similar incidents. So far, CS-AWARE has shown promising results, and work continues with integrating the various components needed for the CS-AWARE solution. An extensive trial period towards the end of the project will help to assess the validity of the approach in day-to-day LPA operations.

8.1 Introduction

As is the case in other sectors, the problem of securing ICT infrastructures is increasingly causing major worries in local public administration. While local public administrations are, compared to other areas, rarely the target of an attack, using its ICT infrastructure as a springboard for the infiltration of other government systems is of great concern for system administrators. Another significant issue is the danger of becoming a victim of collateral damage ensuing from widespread attacks, as happened to hospitals in the 2017 ransomware attacks [1], causing severe damages to local public administration as well, and going far beyond the loss of reputation. Depending on the criticality of services provided by a local public administration, the damage caused by a successful DDoS, ransomware, malware, or, in the worst case, a destruction-orientated APT attack, can be substantial.

Against this background, the H2020-funded CS-AWARE project[1] aims to equip local public administrations with a toolset allowing them to gain a better picture of vulnerabilities and threats or infiltrations of their ICT systems. This will be achieved via an underlying information flow model including components for information collection, analysis and visualisation which contribute to an integrated awareness picture that gives an overview of the current status in the monitored infrastructure and raises the awareness for both looming and already materialized threats.

Starting from a requirements and situation analysis based on workshops following the soft systems methodology (SSM), Rich Pictures serve as tools for developing a core information flow model that facilitates the information collection, analysis and rendering/visualization processes. In addition to these steps, recommendations are suggested that can either be used as support for human decision makers or are directly executed by (re)configuration scripts to realign defensive capabilities in such a way that existing attacks can be dealt with and developing ones can be prevented from getting through.

In CS-AWARE we develop the building blocks for a cybersecurity awareness solution that builds upon a holistic socio-technological system and dependency analysis. An overview of the proposed approach can be seen in Figure 8.1. After data collection, which is composed of static information collected during system and dependency analysis as well as dynamic information collected at run-time, an analysis and decision support component as well multi-lingual support, will process the information to support the main objectives of the solution:

- Provide situational awareness to system operators or administrators via visualization
- Provide supervised self-healing in cases where the analysis engine could determine an automated solution to prevent or mitigate a detected cybersecurity incident
- Provide the capabilities to share cybersecurity related information with relevant communities to help prevent or mitigate similar incidents for other organizations

To ensure the practical feasibility of the approach, processes and tools developed in this project from the requirements analysis onwards, two city administrations, one medium sized and one large and complex which included outsourced operations, are involved to provide the necessary

[1] https://cs-aware.eu/

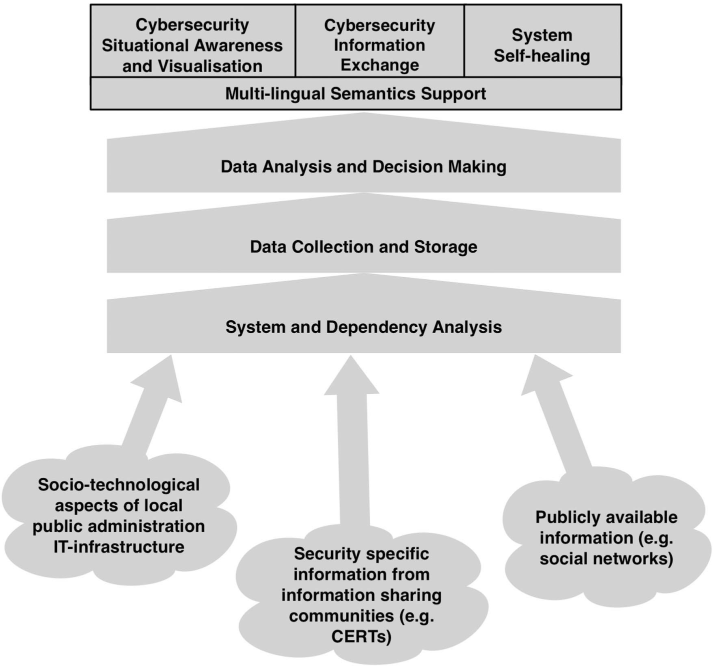

Cybersecurity Situational Awareness and Visualisation
Cybersecurity Information Exchange
System Self-healing
Multi-lingual Semantics Support
Data Analysis and Decision Making
Data Collection and Storage
System and Dependency Analysis
Socio-technological aspects of local public administration IT-infrastructure
Security specific information from information sharing communities (e.g. CERTs)
Publicly available information (e.g. social networks)

Figure 8.1 The CS-AWARE approach.

guidance and support. Assuming that the pilot implementations are satisfactory at the end of the project, the commercialisation group of the project will then advance the toolset and the services around it into a commercial operation. With the Network and Information Security (NIS) Directive [2] and General Data Protection Regulation (GDPR) [3] having become binding legislation in the European Union in May 2018, it is expected that the need for such a toolset will increase, way beyond local public administration.

The remainder of the chapter is organized as follows: Section 2 discusses related work. Section 8.3 details the CS-AWARE concept and framework, while Section 4 specifies implementation aspects of the main framework components. Section 5 discusses the project results and experiences so far, and Section 6 concludes the chapter.

8.2 Related Work

Cybersecurity affects both individuals and organisations, being one of today's most challenging societal security problems. Next to strategic/critical infrastructures, large commercial enterprises, SMEs and also governmental or non-governmental organisations (NGOs) are affected. Expanding beyond the technology-focused boundaries of classical information technology (IT) security, cybersecurity is strongly interlinked with organisational and behavioural aspects of IT operations, and the need to adhere to the existing and upcoming legal and regulatory framework for cybersecurity. This is particularly true in the European Union, where substantial efforts have been made to introduce a comprehensive and coherent legal framework for cybersecurity. Consequent upon the EU cybersecurity strategy [4], the two main legislatory efforts have been the NIS directive [2] and the GDPR [3]. One of the main aspects of the NIS directive, as well as the European cybersecurity strategies is cooperation and collaboration among relevant actors in cybersecurity, as is pictured Figure 8.2 taken from the EU cybersecurity strategy, identifying the main actors relevant for a cooperative and collaborative cybersecurity environment. Enabling technologies for coordination and cooperation efforts are essential for situational awareness and information sharing among relevant communities and authorities. In the long term, it is expected that information sharing can improve cybersecurity sustainably and benefit society and economy in its entirety as an outcome of the enhanced awareness so generated. Current reports such as the 2018 Europol IOCTA (Internet Organised Crime Threat Assessment) [1], support and encourage the growing importance of collaboration and coordination in order to address current and future cybersecurity challenges.

In common with the challenges faced by the NIS, GDPR compliance efforts require greater understanding of an organizations systems in order

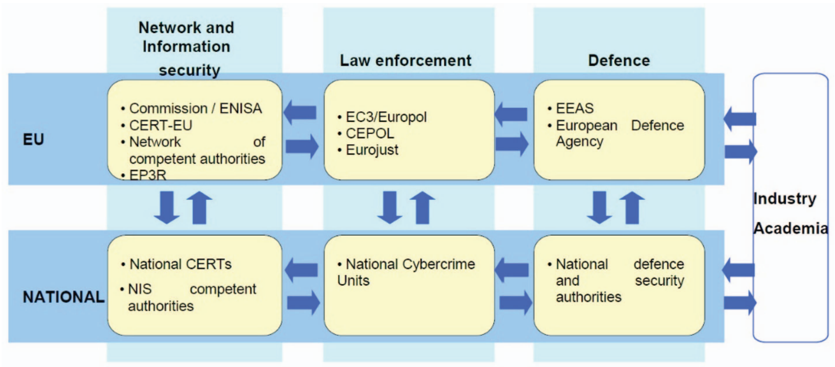

Network and Information security
Law enforcement
Defence
EU
NATIONAL
• Commission / ENISA
• CERT-EU
• Network of competent authorities
• EP3R
• EC3/Europol
• CEPOL
• Eurojust
• EEAS
• European Defence Agency
• National CERTs
• NIS competent authorities
• National Cybercrime Units
• National defence and security authorities
Industry
Academia

Figure 8.2 Roles and responsibilities in European cybersecurity strategy.

to identify and understand GDPR relevant information and information flows. Awareness technologies like the one proposed in CS-AWARE enable organizations to assess and manage GDPR compliance.

Situational awareness in the CS-AWARE context is a runtime mechanism to gather cybersecurity relevant data from an IT infrastructure and visualise the current situation for a user or operator. Understanding the entirety of the cybersecurity relevant aspects of the internal system is one of the cornerstones for ensuring useful as well as successful collaboration and cooperation between institutions. This is a complex task that will greatly improve the cybersecurity of organisations in the context of cybersecurity situational awareness and cooperative/collaborative strategies towards cybersecurity. Therefore, a system and dependency analysis methodology has been introduced to analyse the environment and

1. Identify assets and dependencies within the system and how to monitor them
2. Capture the socio-technical relations within the organisation and the purely technical aspects
3. Identify external information sources, either official or from dedicated communities
4. Provide the results in an output that can be utilised by support tools

Our work is based on established and well proven methods related to systems thinking, the soft systems methodology (SSM) [5, 6] as well as PROTOS-MATINE [7, 8] and GraphingWiki [9] for system analysis and management/visualization of results. Since technology is only one of many factors in cybersecurity, the system and dependency analysis is designed to detect and analyse the socio-technical nature of an IT infrastructure. It does so by considering the human, organisational and technological factors, as well as other legal/regulatory and business related factors that may contribute to the cybersecurity in a specific context. The key concepts are holism (looking at the entirety of the domain and not at isolated components) and systemic (treating things as systems, using systems ideas and adopting a systems perspective). As can be seen in Figure 8.3, systems thinking is a way of looking at some part of the world, by choosing to regard it as a system, using a framework of perspectives to understand its complexity and undertake some process of change.

Hard and soft systems thinking are the two concepts of systems thinking. Hard systems design is based on systems analysis and systems engineering and it builds on the idea that the world is comprised of systems that can be

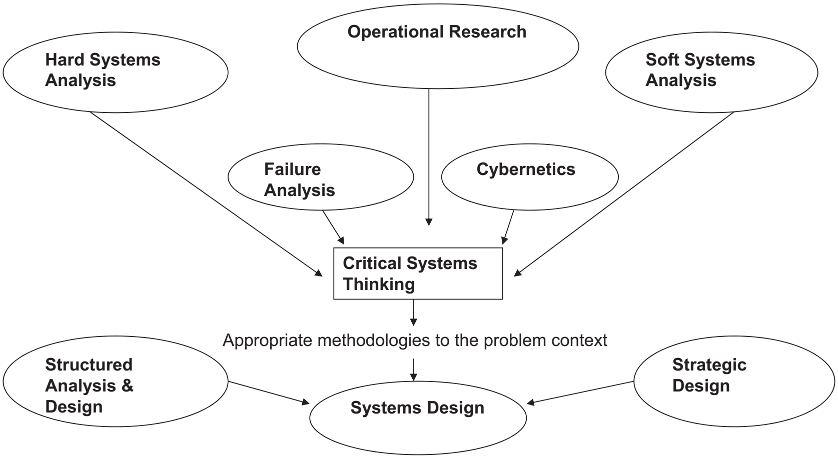


Figure 8.3 Systems thinking.

described and that these systems can be understood through rational analysis. Hard systems design assumes that there is a clear consensus as to the nature of the problem that is to be solved. It is unable to depict, understand, or make provisions for "soft" variables such as people, culture, politics or aesthetics. While hard systems design is highly appropriate for domains involving engineering systems structures that require little input from people, the complex systems and interactions in critical infrastructures or other organisations – especially with cybersecurity in mind – usually do not allow this type of analysis. Soft systems design is therefore much more appropriate and suitable for analysing human activity systems that require constant interaction with, and intervention from people.

Complex systems in software engineering are systems where single components function autonomously but are dependent on the outputs of other components [10, 11] and require abstraction in software engineering can occur in two ways, according to Sokolowski et al. [12] either by limiting the information covered by the model to only the components which are relevant and ignoring the remaining or by reproducing a minimized version of the real-world concept. This procedure of abstraction is critical and sometimes considered one of the most important capabilities of a software engineer [13].

The CS-AWARE modelling approach of the information flow of the complex system is influenced by the Data Flow Diagram (DFD) language defined by Li and Chen [14] but adapted to suit the domains needs. Information flows can cover multiple granularities of interconnections between components, but on a high-level can be classified in three categories: direct information flow, indirect information flow and general information flow [15]. The types of data flows of the original DFD have been adapted, while types of activities were added to ensure that the diversity of the system can be modelled easily. This approach was chosen due to the strong focus and importance of the information flow in the CS-AWARE solution as well as the need for individualised entities.

The role of PROTOS-MATINE and GraphingWiki in this proposed analysis method is to complement the SSM analysis with information from other sources, and provide a solid base for discussion through visualisation in dedicated workshops with the system users and operators. One of the capabilities of GraphingWiki is to instantly link gathered information to other relevant information and thus allow an update of the graphical representation of the analysed system as soon as new information arrives. This feature is used together with SSM analysis to create more dynamic discussions and give even more incentive to the participants to create a system model that is as close to reality as possible.

8.3 The CS-AWARE Concept and Framework

The CS-AWARE framework is the core of the CS-AWARE solution and is based largely on the analysis of cybersecurity requirements for local public administrations and the existing technologies. The aim of the framework is to provide a unified understanding of which components interact with each other and in what way this interaction is made possible. The framework provides a high-level overview of the main components, most of which are represented by one of the consortium partners, as well as a more detailed view of the main subcomponents or processes each of them consist of. Additionally, the relations between these components are defined as well as, in the case of data flows, the data format in which the exchange takes place. The high-level nature of the framework was crucial, since some technical details will only be specifiable during the projects implementation phase.

The CS-AWARE framework consists of an information flow model as well as individual interface definitions for each of the components. The model is a high-level, abstract view on how each of the separate technology components cooperates with the others and in what relation they stand to each other. This might be data flows or also logical control flows between the modules. The focus of the current design of CS-AWARE lies on layers

3, 4 and 7, namely the network, transport as well as application layers of the LPAs systems. To facilitate further analysis, the detailed investigation into the appropriate connections was based on the ETL structured diagram. ETL stands for Extraction, Transformation and Load and is a process most commonly used for database warehouses. Extract stands for the gathering of the data from various sources, Transform for cleaning and manipulating the data to ensure integrity and completeness, Load for transferring the data into its target space [16]. Since the CS-AWARE solution is evidently not a database warehouse, the final layer load was adjusted to better suit the framework's nature and renamed the data-provisioning layer. In our case, the division into layers will mainly be applied to facilitate the structuring of the following, more detailed diagrams of the subcomponents, processes and their interrelationships.

The data extraction layer covers all components responsible for defining relevant data and extracting it, as well as the sources themselves. The system dependency analysis is where the analyst defines relevant sources and data necessary for monitoring the LPAs systems. This information is fed into the data collection module via a control flow, which then extracts the data accordingly.

The data transformation layer summates all components tasked with transforming and analysing the data in some way. The first step is to filter and adapt the data as required before it can be, if necessary, run through the natural language processing information extraction component. The data analysis and pattern recognition and the multi-language support module further process the data. For visualising and sharing the detected incidents and data patterns, the data provisioning layer was defined. This is where all collected information is either visually presented to the end user, shared with selected information sharing communities or used for self-healing rule definition.

The approach chosen to present the CS-AWARE framework interface specifications is based on the classical I/P/O - Input, Process, Output – model, where each component consists of as many input, process and output entities as is required, as visualized in Figure 8.4. For each component, all other building blocks providing data or control flows are summarized as inputs, including which data format they use. Additionally, each component has one or multiple processes or sub components that execute the respective logic of the module and are described in detail as well. Each sub process has inputting and outputting components. Finally, the output components are

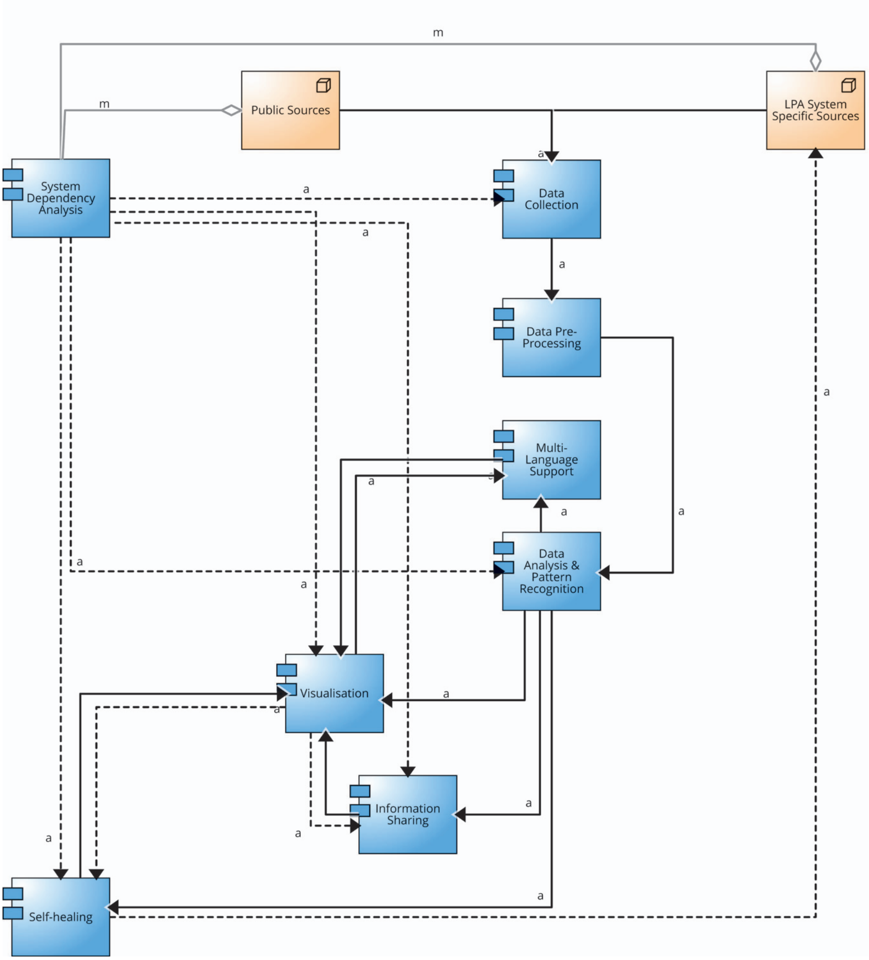
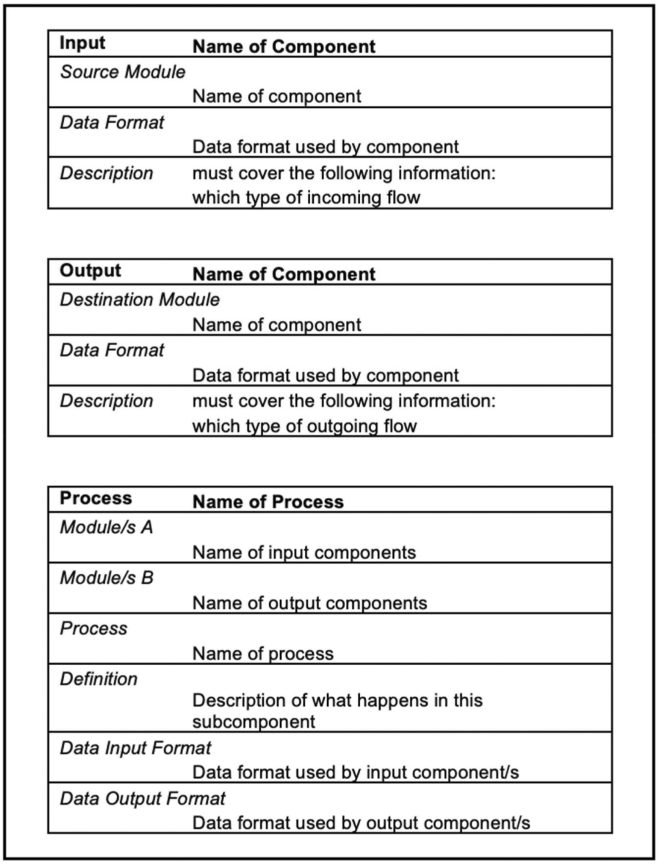


Figure 8.4 CS-AWARE framework.

defined by the same information as the inputs; data format and which type of information flow they use. In preparation of conceptualising the framework, various models and approaches were researched. In the end the CS-AWARE framework was based on the information flows between the components. Nevertheless, it is in line with the NIST cybersecurity framework [17], which identifies five functions as its core: Identify, Protect, Detect, Respond and Recover, making it also compliant with the Italian cybersecurity report, which is based on the NIST framework [18].



Input	**Name of Component**
Source Module	Name of component
Data Format	Data format used by component
Description	must cover the following information: which type of incoming flow

Output	**Name of Component**
Destination Module	Name of component
Data Format	Data format used by component
Description	must cover the following information: which type of outgoing flow

Process	**Name of Process**
Module/s A	Name of input components
Module/s B	Name of output components
Process	Name of process
Definition	Description of what happens in this subcomponent
Data Input Format	Data format used by input component/s
Data Output Format	Data format used by output component/s

Figure 8.5 I/P/O interface definition framework.

It was decided that the communication between components illustrated in Figure 8.5, as well as the communication with relevant authorities via the information sharing component will be in accordance with the STIX2 protocol [19]. STIX2 is a modern and flexible protocol to express and link cybersecurity information and is expected to gain wide adoption over the coming years. An open-source java implementation of the protocol specification was developed by CS-AWARE[2] to facilitate wider adoption of the protocol.

[2] https://github.com/cs-aware/stix2

8.4 Framework Implementation

This Section discusses in more detail the main framework components identified in Section 3. Section 4.1 discusses the system and dependency analysis approach, Section 4.2 details the data collection and pre-processing steps, Section 4.4 and Section 4.3 discuss the multi-language support and data analysis. In Section 4.5 the visualization component is detailed while Sections 4.6 and 4.7 discuss the information sharing and self-healing components respectively.

8.4.1 System and Dependency Analysis

For analysing the networks and systems in the two European CS-AWARE piloting cities in different countries, one with a population in excess of 2.5 million and a one with a population in excess of 150,000, the Systems Methodology (SSM) was used in conjunction with GraphingWiki. The two cities participated in the project as pilot use cases for whom cybersecurity awareness systems were to be built as an output of the project.

The two cities presented very different problem domains: one city's system was extremely large, reflecting as it did the size of the population it served and potentially had 15+ million concurrent citizen users. These users can access the city's systems both from their homes, public buildings and wireless hotspots around the city. This city has outsourced the management many of its key systems. The network topology, the systems and underlying process combine to form what overall is an extremely complex system. The size and complexity of the system precludes any one individual, or indeed small group of employees' form having a complete understanding of all of the systems or the links between systems and their processes and sub-processes. The smaller city operates, manages and maintains all of its own systems.

SSM is a well-proven analytical approach to systems analysis that has been used in an extremely wide range of settings. It is beyond the scope of this chapter to give anything but a brief description of the methodology.

SSM consists of seven stages:

1. Enter the problem situation
2. Express the problem situation
3. Formulate root definitions of systems behaviour
4. Build conceptual models of systems in root definitions
5. Compare models with real-world situations
6. Define possible and feasible changes
7. Take action to improve the problem situation

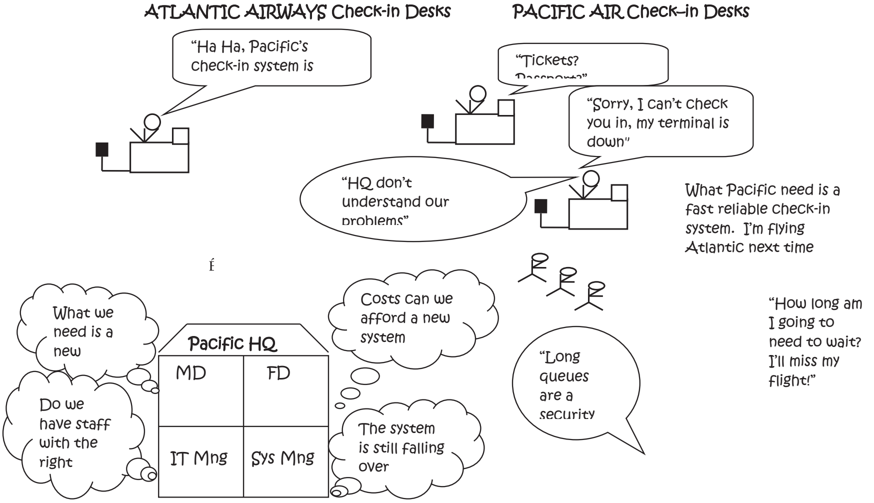

ATLANTIC AIRWAYS Check-in Desks
PACIFIC AIR Check-in Desks
"Ha Ha, Pacific's check-in system is
"Tickets? Passport?"
"Sorry, I can't check you in, my terminal is down"
"HQ don't understand our problems"
What Pacific need is a fast reliable check-in system. I'm flying Atlantic next time
É
What we need is a new
Do we have staff with the right
Pacific HQ
MD
FD
IT Mng
Sys Mng
Costs can we afford a new system
The system is still falling over
"Long queues are a security
"How long am I going to need to wait? I'll miss my flight!"

Figure 8.6 Soft systems analysis rich picture.

The problem situation is explored (expressed), by drawing "Rich Pictures". These pictures are cartoon-like representations that are intended to encompass all of the elements of the situation being examined, be they technical, social, economic, political. A machine drawn example can be seen in Figure 8.6, and depicts a malfunctioning airline passenger check-in system and outlines different viewpoints of those involved when one airline's check-in systems fails.

The analysis of both of the city's operation was conducted in the first two of a proposed series of three workshops. In the first workshops, the participants were asked to draw rich pictures to identify their city's key critical systems (those systems critical to their city's ability to provide services to its citizens and those systems storing or processing sensitive or personal information). Having identified the critical systems, further rich pictures were drawn to explore the interrelationships between the systems so identified in terms of network connectivity and information flows.

These rich pictures informed the development of a series of GraphingWiki graphs, like the one seen in Figure 8.7, which enabled the analysts to represent and model their understanding of the networks and systems in both pilot cities. Each of the nodes is a wiki page that holds the semantic descriptions of the respective elements.

A second round of workshops in the pilot cities was undertaken in which the analysts decided to use the CATWOE approach [20] to gain a better

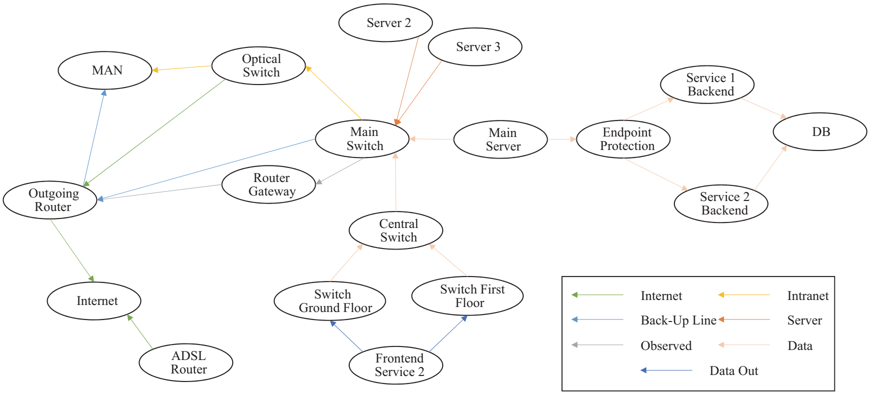


Figure 8.7 System and dependency analysis use case example.

understanding of the processes depicted in the rich pictures created during the first workshops. CATWOE (a mnemonic) was used to identify, express explore and explain the following features in the key rich pictures drawn in the workshops. In doing so, the participants described the processes and sub-processes of the key systems identified in the first and second workshops.

Customers — The organisations customers. The stakeholders of the system

Actors — The employees of the organisation. The people involved in ensuring that a transformation takes place

Transformation — The process by which inputs become outputs e.g. raw materials become finished goods

World view — The wider view of all of the interested parties – employees, suppliers, customers etc... The "big picture"

Owner — The owner of the system or process. The organisation in control

Environmental constraints — Finances, legislation ethics

These CATWOE analyses were then used in a plenary session to further correct and refine the representation of the systems as mapped out in the GraphingWiki, and allowed to identify the information flows through the systems each of those processes produce during day-to-day operations. The identification of information flows is considered a key aspect of understanding where and how to best monitor the systems in the cybersecurity context and are the key to interface the analysed systems with CS-AWARE.

8.4.2 Data Collection and Pre-Processing

For data collection and pre-processing the main challenge in CS-AWARE is to deal with the diversity of data collected from various sources, ranging from cybersecurity information that is heavily structured (e.g. STIX based information sources), to loosely structured information (e.g. log files) or completely free semantic text (e.g. social media). It was decided to convert incoming data from all sources to STIX2 format in the pre-processing stage.

Data collection and pre-processing are applied to multiple sources and the retrieved data is stored in a data-lake. To handle large volumes and a variety of file formats, a big-data pipeline has been implemented following a flexible approach, so that data sources can easily be integrated at a later stage, should additional relevant data sources be identified. Importantly, the collection has been executed in compliance of GDPR regulation where personal data are removed or anonymized at source, since personal data is not required for CS-AWARE operation in the majority of use cases. The implemented framework aggregates and ingests three main classes of information sources:

- Logs from servers, databases, applications and network/security devices from within monitored systems
- Cyber threat intelligence from specialised websites and feeds
- More general cybersecurity related notifications and warnings collected from social networks

In order to collect information from the monitored systems within the local public administrations that usually do not have APIs for data collection, a collector interface was developed to be hosted within the monitored systems. It acts as a local collector of data that is relevant for CS-AWARE and provides an interface to the CS-AWARE solution which may be hosted in the cloud. The conversion to STIX2 format is usually straight forward, because the relevant information is often based on unusual behaviour which can be easily modelled in STIX2.

Threat intelligence sources usually provide a public API that allows collection of data, but there is no agreed or standardized data format in which this data is provided. Common formats are among others STIX1/STIX2, comma separated values (CSV), eXtensible Markup Language (XML) or Java Object Notation (JSON). Since CS-AWARE operates on STIX2, all collected data entries are converted to STIX2 in data pre-processing. Since almost exclusively information with a strong cybersecurity context is shared by threat intelligence sources, the conversion is usually straight-forward.

Threat intelligence notifications collection is performed every 12 hours and stored within the CS-AWARE repository.

As part of this project we want to explore the opportunity of cybersecurity prevention and notifications by listening to social media sources such as Reddit and Twitter. The intuition here is that a cyber-attack may propagate following a certain pattern that could be anticipated by social media warnings, and social media conversations often provide an early indicator to information that may be shared by threat intelligence at a later time. A challenge with utilizing unstructured semantic text like social media is to assess the relevance of each element and assign a structure to it so that it can be processed in an automated way. In CS-AWARE we try to answer this challenge with a natural language processing (NLP) based information extraction approach that will be discussed in more detail in Section 4.3.

As project repository we believe that a winning approach would be a cloud based big-data repository since it offers a ready-to-use framework designed to scale up in a cost effective manner. For this type of challenge, a popular approach involves using a queue system, such as Apache Kafka, and a database where the data is stored; this infrastructure could be well replicated on major cloud providers. Having said that, a full functioning big-data pipeline has a fixed cost even if not fully exploited. For this reason, we preferred a slim and flexible solution where costs are compressed. In more detail, we created the CS-AWARE data-lake on AWS S3[3] storage. AWS S3 provides capabilities to store and retrieve any amount of data from anywhere. It is worth mentioning that thanks to a structured folder hierarchy, it is intuitive and straight forward to retrieve the needed information. Despite the low cost and simplicity, this approach already demonstrated to be fast and stable.

8.4.3 Multi-language Support

In CS-AWARE, the existing technology to support handling of multiple languages is used and has been adopted to fit specific needs of the project context and the use cases. To this aim *Graphene*, a rule-based information extraction system developed in the context of research conducted at the University of Passau, was utilized. There are two use-cases for multi-language support in CS-AWARE: multi-language support at the input when cybersecurity relevant information is collected from multiple sources, and

[3] https://aws.amazon.com/s3/

multi-language support at the output to inform the system operators of the systems security status in their chosen language. In this Section we focus on the first use case, where the challenge is not only to translate new incoming information to a meta-language, but at the same time to extract the most relevant information using natural language processing (NLP) methods.

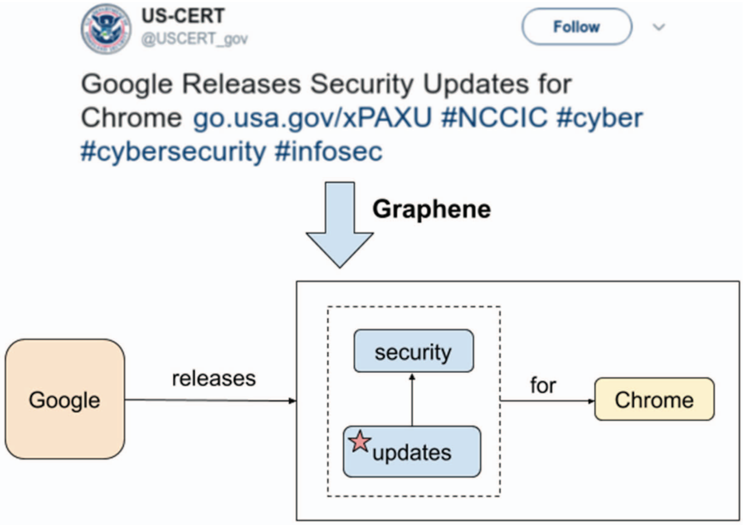
In the project framework, Graphene is responsible for all functions of the NLP information extraction component. The tool uses a two-layered transformation stage consisting of a *clausal disembedding layer* and a *phrasal disembedding layer*, together with *rhetorical relation identification*. To put this in simpler terms, the main approach we take here is to *simplify complex sentences* before applying a set of tailored rules to transform a text into the knowledge graph. During the CS-AWARE project, we had the opportunity to mature the original research prototype as a technology which is both easy to deploy as a service and integrate as a product using de-facto web standards. Additionally, we also had the opportunity to implement and add a new extraction layer responsible for transforming complex categories – what one would call 'coarse-grained information' – into a graph of fine-grained knowledge, as described in the implementation section.

Consequent upon Graphene's ability to extract complex categories, we are able to extract useful information in the correct level of granularity. As an example, we consider the case of a recent tweet written by the United States Computer Emergency Readiness Team (US-CERT), as shown in Figure 8.8.

Once we remove the links and hashtags, the knowledge graph generated from Graphene allows us to identify vendor and products that might be under attack or suffering from new vulnerabilities. With this functionality, both types of information can be forwarded to users and system admins as quickly as they are published in a social network like Twitter. More elaborate information and technical details about the information extraction strategy, including the sentence simplification step and the identification of the rhetorical structures can be found in [21], while for the extraction of complex categories more elaborate information can be found in [22].

8.4.4 Data Analysis

One of the main tasks of the CS-AWARE platform is to look for various threat patterns some of which may have not been detected or recognised as such before and which can signal either a clear threat or a suspicious behaviour that may possibly or potentially be a threat. The way we define a threat pattern at a conceptual level that it is considered as an open set of individual threat parameters with unique settings/values aimed to capture anomalous events.



US-CERT
@USCERT_gov

Google Releases Security Updates for Chrome go.usa.gov/xPAXU #NCCIC #cyber #cybersecurity #infosec

Figure 8.8 From tweets to knowledge graphs.

Such a set of threat parameters can be altered and improved with time as the knowledge about threats expands. Once such patterns catch an occurrence of multiple suspicious events simultaneously, then the identified events are flagged for further analysis. Many times one suspicious event may not be enough to be considered a threat but when multiple suspicious events happen, then the chances to have a threat increases.

In light of the above, we take as data analysis as being the set of processes where all data sources are assembled, combined and searched for unusual or threat patterns. Handling the data sources, their format and the way they should be cleaned from overhead, prepared and then analysed is vital for finding unusual threats or patterns that otherwise may go undetected by the existing tools in the market. Our data analysis efforts are focused on internal data sources belonging to organizations that use the CS-AWARE platform, such as logs, as well as external data sources such as threat intelligence platforms, specialized cyber-security forums, news and solutions. Such data is in a raw form and will have to be filtered and processed in order to extract only the most useful data for analysis.

The data analysis focuses on extracting the most probable elements of information that could form cyber security threats as well as info related to such threats. The data analysis that builds on a Peracton MAARS$^{\mathrm{TM}}$ component combines the above sources to identify threats and possible security incidents. Some combinations, assuming a proper information pre-processing, will be quite straightforward to process while others might need some more advanced analysis.

In this respect, the data analysis engine should be able to perform at least the following with regards to the above sources:

- *Match vulnerability information to assets* – e.g. a vulnerability found on a specific OS version; is it applicable to monitored LPAs.
- *Combine threat information with logs and assets* – the analysis should be done based on specific attributes that characterize an attack e.g. to identify a security incident regarding suspicious activity originating from a specific IP and targeting specific systems there is a need to match these attack characteristics to the information we have, i.e. we have to analyse threat information provided by external sources to give values to these attributes, and once we do so, process LPA's logs and LPA's assets inventory to identify these.
- *Attack pattern matching* – analyse network and system activity to identify potential security incidents based on attack patterns either collected from external sources or specified by CS-AWARE security experts. The engine's efficiency strongly depends on the defined patterns. Although the engine should demonstrate its ability through a pre-defined set of patterns, it should also be able to accommodate additional patterns that security experts would like to define in the future.

We expect that the data analysis should provide information about the criticality of the specific security incident and, based on this classification, suggest the most appropriate risk mitigation option (if available from data). Revisiting the above example where a threat that reports suspicious activity originates from specific IPs and targets specific systems, the (risk) analysis for the following scenarios will give us the corresponding results described below:

- *Scenario 1*: Threat is flagged by external sources as critical and the LPA has systems that are vulnerable to this malicious activity: the risk for the organization is high. A risk mitigation strategy should be applied, i.e. an action is required to mitigate the risk, the details of which are subject to the information provided by external sources or by a CS-AWARE security expert.
- *Scenario 2*: Threat is flagged by external sources as critical, logs indicate incoming traffic from this IP, yet the LPA has *no* systems that are vulnerable to this malicious activity: the risk is low. In this case, *no* action is required.

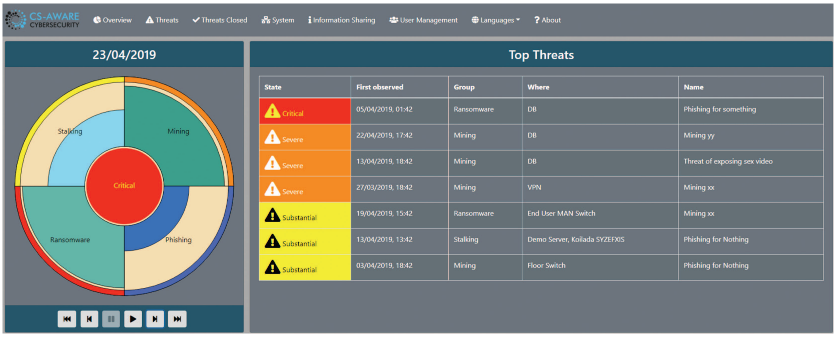
8.4.5 Visualization

The visualization component will show the users (e.g. system administrator, management) the level of cyber threats to their system and will make it possible for system administrators to cooperate with the system to identify self-healing procedures and to share information with external partners regarding new cyber threats that have been identified in the analytics module.

The visualisation module is also the main user interface of the CS-AWARE product for administration. In order to provide cybersecurity awareness, it is necessary to visualise the threats, the threat level, the possible self-healing strategies and the information shared with the cybersecurity community. It is also necessary to have an interface to communicate back to the system information regarding controlling the aforementioned topics as well as lower level administration. The visualisation component will take care of this according to the work done in the dependency analysis and in good cooperation with other parts of the CS-AWARE solution. For the interchange of data, the STIX2 format has been determined as being the basic communication format between the modules, as it is commonly used in the field of cybersecurity and is both fairly stable, extensible when needed and with a reasonable support of frameworks with which to work.

The number of cybersecurity events has also been rising over the past years and as more and more of our society is based on information systems, the issues have multiplied over time. Before this project, a number of independent vendors have different visualisation means to show how their particular system is threatened by cyber security events. In a large-scale facility like most LPAs this results in a large number of reports on what is going on in their field of operation. For the system administrator this only gives a partial overview of the cyber security events, as it on one hand only delivers the view from the single vendor, and on the other hand often is too complex to be useful. The number of different reports to choose from can be high and they are usually only collected per vendor. This makes it difficult to assess the full cyber security overview. The paucity of overview leaves the cybersecurity awareness level lower than it could be. This is a situation that needs to be remedied.

The main gap is that current systems lack a significant cognitive component, in order to propagate the overall level of cybersecurity awareness. Specifically, we have identified the lack of single point of overview, and together with the rising level of entropy in cybersecurity reporting, which is believed to be the consequence of the multitude of sources that may not be connected. This both results with information overload and the already



Top Threats

State	First observed	Group	Where	Name
⚠ Critical	05/04/2019, 01:42	Ransomware	DB	Phishing for something
⚠ Severe	22/04/2019, 17:42	Mining	DB	Mining yy
⚠ Severe	13/04/2019, 18:42	Mining	DB	Threat of exposing sex video
⚠ Severe	27/03/2019, 18:42	Mining	VPN	Mining xx
⚠ Substantial	19/04/2019, 15:42	Ransomware	End User MAN Switch	Mining xx
⚠ Substantial	13/04/2019, 13:42	Stalking	Demo Server, Koilada SYZEFXIS	Phishing for Nothing
⚠ Substantial	03/04/2019, 18:42	Mining	Floor Switch	Phishing for Nothing

Figure 8.9 The CS-AWARE visualization component.

mentioned lack of common cybersecurity awareness. The main solution is to have a single interface to propagate the immediate cybersecurity awareness situation to the system administrator and other users who have a need for this information. For this we have developed a dashboard that – in an early version – is shown in Figure 8.9. It makes it easy to overview all concurrent cybersecurity threats and vulnerabilities as well as a summarised threat level. Through the dashboard, the system administrator has direct access to self-healing strategies, suggestions as well as possible information sharing texts on newly found threats.

We are generating a single view of cybersecurity threats and vulnerabilities that will show all of the major threat types and the summarised threat level. These will be shown over time to help understand the urgency and how the change in threat level is evolving, in order to mitigate a threat in the best way at that time. A reduction in time spent looking for a cybersecurity issue is worth many hours of post-mortem issue fixing and cleaning. Notice that the dashboard will have a graphic that continuously shows development over time in both size and colour, in order to let the system administrator act swiftly and becoming aware of cybersecurity events much faster than going through heaps of internet pages to find a possible change.

The visualisation component has interfaces to the system and dependency analysis, the data analysis and pattern recognition as well as the multi-language (NLP) support components, and also to the self-healing and information sharing components, where information sharing to the cybersecurity communities will be for the common good. This way the visualisation component enhances the cybersecurity awareness and helps the system administrators maintain their systems unaffected through a faster and better decision making and self-healing process.

8.4.6 Cybersecurity Information Exchange

The CS-AWARE cybersecurity information exchange (CIE) provides a dissemination point for cyber threat information (CTI) that CS-AWARE components have collected, analysed, identified and classified as “shareable”. It is the interface to external entities, such as Computer Emergency Response Teams (CERTs), Computer Security Incident Response Teams (CSIRTs) and other threat intelligence platforms to inform them about threats, sightings (i.e. an observation related to a threat) and mitigation actions. This information will be mainly produced by the CS-AWARE data analysis component or by external sources and enhanced by the former.

CTI is information that is constantly generated and shared among devices and departments within (especially large) organisations which have well-established procedures for appropriately handling sensitive, classified and personal information found within CTI. When CTI is about to be shared with external entities, several interoperability and security issues have to be confronted [23], which can be categorised according to the three layers depicted in Figure 8.10.

Although the *legal* framework might encourage or require the sharing of cyber-threat information, as the NIS directive does, several other legal requirements might prohibit or restrict the uncontrolled sharing of CTI. One of the main legal restrictions arises from the GDPR and relates to the personal information shared with external entities without the user’s consent. In the case of CTI, personal data might be part of the shared sightings, such as IP addresses or usernames of entities that have been identified as sources of malicious activity. CTI sharing with external entities should not impact privacy and personally identifiable information (PII), and therefore, data

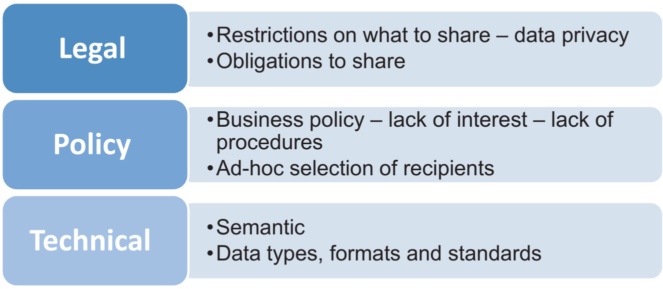

Legal
•Restrictions on what to share – data privacy
•Obligations to share
Policy
•Business policy – lack of interest – lack of procedures
•Ad-hoc selection of recipients
Technical
•Semantic
•Data types, formats and standards

Figure 8.10 CTI exchange interoperability layers.

anonymization should take place if necessary, prior to sharing CTI with external entities or being made public. However, certain data that under certain circumstances might be considered as personal (e.g. IPs), are very important for the receiving parties to have. Otherwise the information provided becomes useless, and therefore should be excluded from any anonymization processing. Moreover, based on Article 49 of the GDPR, the processing of personal data by certain entities, such as CERTs and CSIRTs, strictly for the purposes of ensuring network security is permitted as it constitutes a legitimate interest of the data controller.

An organization's *policy* should address issues related to information sharing, while well-established procedures and appropriately deployed measures will help avoid the leakage of classified or sensitive information. Data sanitisation [24] is one of the solutions that the organisations should consider utilising to ensure that no sensitive or classified information is disclosed to unauthorised entities while sharing CTI with external entities. Policy restrictions with regards to sharing should also be supported by appropriate technical measures.

On the *technical* layer, adoption of standardised schemes used for sharing cyber-threat information is deemed necessary to achieve the necessary semantic and technical interoperability. The STIX2 protocol is the information model and serialisation solution adopted by CS-AWARE for the communication and sharing of CTI.

STIX2 also supports data markings which can facilitate enforcement of policies regarding the sharing of information. More specifically, STIX2 supports statements (copyright, terms of use, ...) applied to the shared content as well as the Traffic Light Protocol (TLP)[4,5] (a set of designations used to ensure that sensitive information is shared with the appropriate audience by providing four options as shown in Figure 8.11). Although optimized for human readability and person-to-person sharing and not for automated sharing exchanges, the adoption of TLP in CS-AWARE will help restrict information sharing only with specific entities or platforms and avoid any further unnecessary or unauthorized dissemination thereof.

Considering the limitations of TLP which cannot support fine-grained policies, the CS-AWARE information exchange component also adopted the Information Exchange Policy (IEP), a JSON based framework developed by

[4] https://www.first.org/global/sigs/tlp/

[5] https://www.enisa.europa.eu/topics/csirts-in-europe/glossary/considerations-on-the-traffic-light-protocol

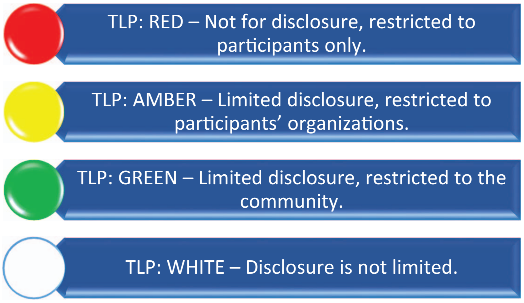

TLP: RED – Not for disclosure, restricted to participants only.
TLP: AMBER – Limited disclosure, restricted to participants' organizations.
TLP: GREEN – Limited disclosure, restricted to the community.
TLP: WHITE – Disclosure is not limited.

Figure 8.11 The traffic light protocol.

FIRST IEP SIG (2016) [25]. IEP is not supported in the current version of STIX, yet STIX compatibility was considered in its design.

8.4.7 System Self-Healing

Self-healing is described as the ability of systems to autonomously diagnose and recover from faults with transparency and within certain criteria. Although these criteria vary according to the system's infrastructure, they often include requirements such as availability, reliability and stability [26]. Self-healing constitutes an important building block of the CS-AWARE architecture, which aims to assist LPA administrators in responding to identified vulnerabilities and high-risk threats by providing customised healing solutions or recommendations. The self-healing component is an innovative fully-supervised solution that uses the results of the analysis performed by the analysis component. The latter processes cyber threat information collected from external sources, internal logs and LPA architecture specifics and produces knowledge about potential high-risk situations for a specific LPA. Based on the aforementioned outcome, self-healing looks for the most appropriate mitigation solution among those provided by the external sources or found in the self-healing enhanced database of appropriate solutions. Supervision is defined as the degree of required human interaction concerning the feedback mechanism and the expansion of self-healing mechanisms [26]. Self-healing systems are categorised as fully supervised, semi-supervised or unsupervised. Figure 8.12 provides an overview of the research work accomplished on self-healing properties as published in [27].

The composed mitigation rules aim to enhance the availability and overall security of the system while simultaneously reducing the required

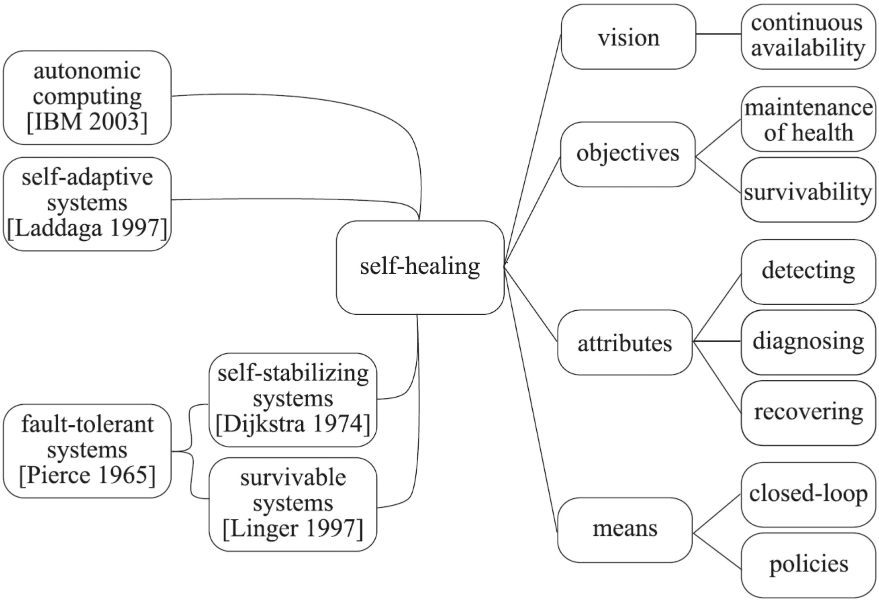

autonomic computing [IBM 2003]
self-adaptive systems [Laddaga 1997]
self-healing
fault-tolerant systems [Pierce 1965]
self-stabilizing systems [Dijkstra 1974]
survivable systems [Linger 1997]
vision
continuous availability
objectives
maintenance of health
survivability
attributes
detecting
diagnosing
recovering
means
closed-loop
policies

Figure 8.12 Properties of self-healing research.

workload of system maintainability. Furthermore, the CS-AWARE self-healing component has the ability autonomously to diagnose and mitigate threats, while ensuring that the system's administrator, who is always aware of the system behaviour, can prevent configuration changes that may raise incompatibility issues.

The self-healing component also interacts with the visualisation component for the following purposes:

- inform administrators about mitigation solutions applied to LPA systems
- request LPA administrator permission to apply a solution
- provide recommendations about how to confront an identified high-risk situation or vulnerability

The self-healing component is fully supervised, always allowing the LPA administrator to decide whether or not they want to apply the suggested mitigation rule. It utilises the results of the data analysis component provided in STIX2. Once the self-healing receives input data from the analysis component, it identifies the threat type and composes the proper mitigation rule autonomously. Rules composed by the self-healing module incorporate three alternatives:

- Inform LPAs about which acts to perform in order to avoid the threat or reduce the impact (recommendation)
- Ask for the LPA administrator's permission in order to apply the rule automatically
- Automatically apply the rule, provided that the administrator has set this preference through the visualisation component

The self-healing component consists of three main and three auxiliary subcomponents, whose interaction is shown in Figure 8.13. The main subcomponents were defined in the CS-AWARE framework while auxiliary subcomponents were defined during the design phase to facilitate the composition and application of mitigation rules. The self-healing policies are a database which contains records of potential threats that might be detected in an LPA system and the corresponding mitigation rules. The mitigation rules are stored in a human-readable generic format as well as in machine-readable format. Moreover, the self-healing policies subcomponent includes entries which contain the CLI syntax of LPAs central nodes.

The decision engine initiates the composition of a rule. It performs searches of the self-healing policies database for a matching rule. If it finds a match, it initiates a rule which is in a human-readable format. The security rules composer accepts input from the decision engine subcomponent in a human-readable format and converts it to a machine-readable format based on the CLI syntax of the affected node. The parser parses the STIX package

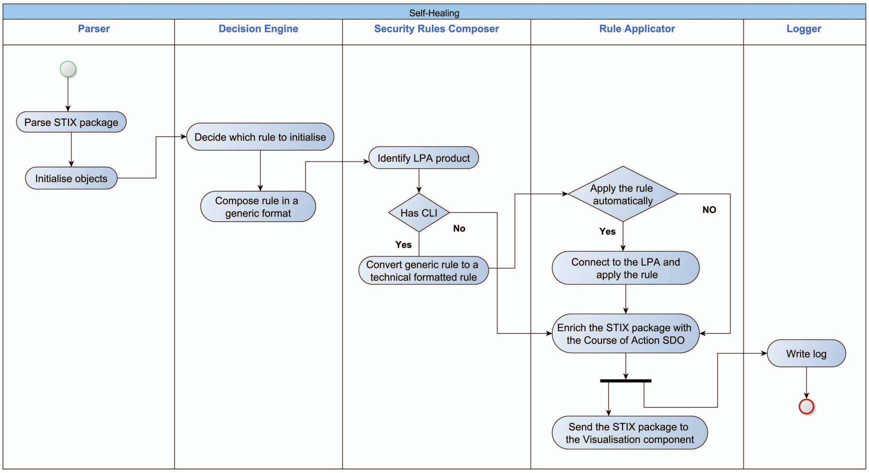

Self-Healing
Parser
Decision Engine
Security Rules Composer
Rule Applicator
Logger
Parse STIX package
Initialise objects
Decide which rule to initialise
Compose rule in a generic format
Identify LPA product
Has CLI
No
Yes
Convert generic rule to a technical formatted rule
Apply the rule automatically
NO
Yes
Connect to the LPA and apply the rule
Enrich the STIX package with the Course of Action SDO
Send the STIX package to the Visualisation component
Write log

Figure 8.13 Self-healing subcomponents activity diagram.

and extracts useful data for the process of mitigation rules composition, and the rule applicator is responsible for enriching the STIX package with the mitigation rule, sending data to the visualisation component and for applying the rule on the remote machine. In case the mitigation rule must be applied remotely then the self-healing connects to the remote node and applies the rule automatically provided that the LPA administrator has given permission. Finally, the logger writes a log entry in the log file which contains information about how the mitigation rule was applied.

8.5 Discussion

The demand for cybersecurity tools is strong. An alarming rate of purposeful cyber-attacks forces authorities on different levels to do more than just to be reactive operations. At the same time new regulatory and legal requirements are implemented by the highest-level authorities and are effecting how systems can be operated and data can be handled on the regional level. In Europe, especially the NIS directive is concerned with how the most critical services for our society are handling cybersecurity, while the GDPR is protecting an individual person's information and privacy. This has caused actions and worries with private companies but is affecting also many functions of local public administrations. Although the local public administrations have not been the direct targets of malware attacks they are crucial providers of services governing our everyday lives and are heavily influencing society on a regional level. The CS-AWARE project has proven to be even more current and relevant than we could have anticipated during the time of writing the proposal.

The first year of the project has been successful. Two rounds of dependency analysis workshops at our piloting municipalities have been completed and have provided extensive insight into the operations of local public administrations. We have gained valuable information that has influenced and guided the CS-AWARE framework development and implementation. We have seen that there are substantial differences in LPA operations between countries even on the European level. Besides the obvious differences in language in national and regional levels in Europe, we have seen that the rules and regulations guiding LPA operations are substantially different between countries, and may affect how cybersecurity tools like the CS-AWARE toolset can be deployed and operated. We have also seen however that the CS-AWARE concept and framework is well suited to handle these differences due to the flexible and socio-technological analysis at its base. We believe that we have proven the framework to be valid. It is now modified and adjusted based on knowledge and circumstances derived from the LPA use cases. The project will continue with the framework implementation and integration, and an extensive piloting phase towards the end of the project will allow us to draw broader conclusions about the usefulness of cybersecurity awareness technologies in day-to-day operations of local public administrations.

An important lesson we have already learned at this stage is how important collaboration and information sharing are. Cooperation and collaboration is essential and becoming more relevant in future, since there are many actors on the local public administration level. Small cities and communes with individually centralized organizations, but each distributing responsibilities among external experts. The larger the commune, the greater appears to be the silo effect. Then even a single service forms, an isolated unit which does not have direct collaboration with other city services. Information sharing is therefore a key factor to generate and understand the full picture of the internal infrastructures. While our information sharing efforts were focused on sharing cybersecurity information with external authorities, such as the NIS competent authorities listed in Figure 8.2, we have seen that in practice already sharing with different actors on the local level (other departments or suppliers) may have a significant positive effect on cybersecurity. This is one aspect that will be more closely looked into during the piloting phase of CS-AWARE. We are investigating this even further in another H2020 project, CinCan (Continuous Integration for the Collaborative Analysis of Incidents)[6], where we also try to promote sharing and reporting vulnerability information between different countries' CERT organizations.

We feel that CS-AWARE is not just an individual project, but a continuous path we need and have now started to follow. Technology touches every aspect of our lives and we need tools that allow us to safely utilise them by covering all legal security requirements.

8.6 Conclusion

In this Chapter we have presented the EU-H2020 project CS-AWARE (running from 2017 to 2020), aiming to provide cybersecurity awareness technology to local public administrations. CS-AWARE has several unique features, like the socio-technological system and dependency analysis at the

[6] https://cincan.io/index.html

core of the technology that allows a fine grained understanding of LPA cybersecurity requirements on a per case basis. Furthermore, the strong focus on automated incident detection and classification, as well as our efforts towards system self-healing and cooperation/collaboration with relevant authorities are pushing the current state-of-the-art, and are in line with cybersecurity efforts on a European and global level.

In light of a substantially changing legal cybersecurity framework in Europe, we have shown that CS-AWARE is an enabling technology for many cybersecurity requirements imposed by these regulations. For example, information sharing of cybersecurity incidents is a requirement of the NIS directive for organizations classified as critical infrastructures, and may in future be extended to other sectors as well. Similarly, the identification of personal information and information flows within organizations systems, as done in the system and dependency analysis of CS-AWARE, is a key requirement for GDPR compliance.

We have detailed the CS-AWARE framework and have shown how the different building blocks are implemented in CS-AWARE. We have discussed the first results of the project, especially the outcomes of two rounds of system and dependency analysis workshops in the piloting municipalities of CS-AWARE, and we have discussed how those results are influencing the framework implementation and integration in preparation for the piloting phase of the project. Our initial results show the necessity of awareness technologies in LPAs. Administrators and system operators are looking for solutions that improve awareness of cybersecurity incidents on a system level and assist with prevention or mitigation of such incidents. We have seen a specific need for awareness as well as improved collaboration and cooperation between different departments or suppliers, an area that is often neglected but has significant potential for introducing cybersecurity risks.

CS-AWARE will continue with further developing of the technological base and integration of the components that form the CS-AWARE framework. An extensive piloting phase towards the end of the project will give insights into the practical feasibility and relevance of the awareness generating technologies, and allow us to evaluate how both system administrators and system users can benefit from CS-AWARE. The piloting phase will be accompanied by social sciences based study to evaluate how the CS-AWARE technologies are accepted by its users in day-to-day operations. At the same time, we will continue to promote CS-AWARE among potential users, implementers and authorities to bridge the gap between legal and regulatory requirements and actual technology that can fulfil those requirements. In an era where it is thought that cybersecurity can only be effective through cooperation and collaboration, constant interaction between the main actors is important to achieve a comprehensive and holistic solution.

Acknowledgements

The authors would like to thank the EU H2020 project CS-AWARE (grant number 740723) for supporting the research presented in this chapter.

References

[1] Europol, "Internet Organized Crime Threat Assessment (IOCTA) 2018," Online, 2018.

[2] THE EUROPEAN PARLIAMENT AND THE COUNCIL OF THE EUROPEAN UNION, DIRECTIVE (EU) 2016/1148 OF THE EUROPEAN PARLIAMENT AND OF THE COUNCIL *of 6 July 2016 concerning measures for a high common level of security of network and information systems across the Union,* 2016.

[3] THE COUNCIL OF THE EUROPEAN UNION, REGULATION (EU) 2016/679 OF THE EUROPEAN PARLIAMENT AND OF THE COUNCIL of *27 April 2016 on the protection of natural persons with regard to the processing of personal data and on the free movement of such data, and repealing Directive 95/46/EC,* 2016.

[4] European Commission and High Representative of the European Union for Foreign Affairs and Security Policy, *Cybersecurity Strategy of the European Union: An Open, Safe and Secure Cyberspace,* JOIN(2013) 1 final, 2013.

[5] P. Checkland, "Systems Thinking, Systems Practice," Wiley [rev 1999 ed], 1981.

[6] P. Checkland, "Soft Systems in Action," Wiley [rev 1999 ed], 1990.

[7] P. Pietikainen, K. Karjalainen, J. Röning and J. Eronen, "Socio-technical Security Assessment of a VoIP System," in 2010 *Fourth International Conference on Emerging Security Information, Systems and Technologies,* 2010.

[8] T. Schaberreiter, K. Kittilä, K. Halunen, J. Röning and D. Khadraoui, "Risk Assessment in Critical Infrastructure Security Modelling Based on Dependency Analysis," *in Critical Information Infrastructure Security:*

6th International Workshop, CRITIS 2011, Lucerne, Switzerland, September 8–9, 2011, Revised Selected Papers, 2011.

[9] J. Eronen and J. Röning, “Graphingwiki – a semantic wiki extension for visualising and inferring protocol dependency,” in First Workshop on *Semantic Wikis – From Wiki To Semantics, 2006.*

[10] J. Jiang, J. Yu and J. Lei, “Finding influential agent groups in complex multiagent software systems based on citation network analyses,” *Advances in Engineering Software,* pp. 57–69, 2015.

[11] L. Saitta and J.-D. Zucker, Abstraction in artificial intelligence and complex systems, Springer, 2013.

[12] J. Sokolowski, C. Turnitsa and S. Diallo, “A conceptual modeling method for critical infrastructure modeling,” *in Simulation Symposium,* 2008. *ANSS 2008. 41st Annual,* 2008.

[13] J. Kramer, “Is abstraction the key to computing?,” *Communications of the ACM,* pp. 36–42, 2007.

[14] Y.-L. Chen and Q. Li, Modeling and Analysis of Enterprise and Information Systems: from requirements to realization, Springer, 2009.

[15] P. Clemente, J. Rouzaud-Cornabas and C. Toinard, From a generic framework for expressing integrity properties to a dynamic mac enforcement for operating systems, Springer, 2010, pp. 131–161.

[16] S. Bansal and S. Kagemann, “Integrating big data: A semantic extract-transform-load framework,” *Computer,* pp. 42–50, 2015.

[17] NIST National Institute of Standards and Technology, “Framework for Improving Critical Infrastructure Cybersecurity,” 2015.

[18] CINI Cyber Security National Laboratory, “Italian Cyber Security Report 2015 – A National Cyber Security Framework,” 2016.

[19] OASIS Committee Specification 01, STIX Version 2.0. Part 1: STIX Core Concepts, R. Piazza, J. Wunder and B. Jordan, Eds., 2017.

[20] D. S. Smyth and P. B. Checkland, “Using a Systems Approach: The Structure of Root Definitions,” *Journal of Applied Systems Analysis,* vol. 6, no. 1, 1976.

[21] M. Cetto, C. Niklaus, A. Freitas and S. Handschuh, “Graphene: Semantically-Linked Propositions in Open Information Extraction,” in *In Proceedings of the 27th International Conference on Computational Linguistics (COLING),* New-Mexico, USA, 2018.

[22] J. E. Sales, A. Freitas, B. Davis and S. Handschuh, “A Compositional-Distributional Semantic Model for Searching Complex Entity Categories,” in *5th Joint Conference on Lexical and Computational Semantics (*SEM),* Berlin, 2016.

[23] C. S. Johnson, M. L. Badger, D. A. Waltermire, J. Snyder and C. Skorupka, "Guide to Cyber Threat Information Sharing," *National Institute of Standards and Technology,* pp. NIST SP 800-150, 2016.

[24] M. Bishop, B. Bhumiratana, R. Crawford and K. Lwvitt, "How to sanitize data?," in *13th IEEE International Workshops on Enabling Technologies: Infrastructure for Collaborative Enterprises,* Modena, 2004.

[25] Forum of Incident Response and Security Teams (FIRST), "Information Exchange Policy Framework, Version 1.0".

[26] C. Schneider, A. Barker and S. Dobson, "A survey of self-healing systems frameworks," *Software: Practice and experience,* vol. 45, no. 10, pp. 1378–1394, 2014.

[27] H. Psaier and S. Dustdar, "A survey on self-healing systems: approaches and systems," *Computing,* vol. 91, no. 1, p. 47, 2010.