D7.1

Security & Privacy Algorithm Innovation Report

Project number:	833683
Project acronym:	CyberSANE
Project title:	Cyber Security Incident Handling, Warning and Response System for the European Critical Infrastructures
Start date of the project:	1st September, 2019
Duration:	36 months
Programme:	H2020-SU-ICT-2018

Deliverable type:	Report
Deliverable reference number:	DS-01-833683 / D<7.1>/ DRAFT | N.1
Work package contributing to the deliverable:	WP 7
Due date:	02 2021 – M18
Actual submission date:	19/03/2021

Responsible organisation:	CNR
Editor:	Oleksii Osliak (CNR)
Dissemination level:	PU
Revision:	DRAFT | N.1

Abstract:	Report on novel algorithms for data anonymization, document sanitization as well as the cryptographic primitives and protocols. This deliverable will be a report reflecting the outcomes of T7.1, T7.2 and T7.3.
Keywords:	Sensitive data protection, Access Control, Usage Control, Encryption, Secure Information Sharing
	The project CyberSANE has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 833683.

Editor

Oleksii Osliak (CNR)

Contributors (ordered according to beneficiary numbers)

Luis Landeiro, Daniel Ascensão (PDMFC)

Konstantinos Kontakis, Georgios Nikitakis, Tziortzia Koutsouri (STS)

Fabio Martinelli, Oleksii Osliak (CNR)

Karantzias Thanos, Serra Marcello, Tamburini Nicola, Laras Paris (Maggioli)

Eva Papadogiannaki, Christos Tzagkarakis, Manos Athanatos (FORTH)

Daniele Dellagiacoma, Umar Ismail, Haris Mouratidis (UoB)

Nathalie Mitton (Inria)

Disclaimer

Executive Summary

Due to the potential negative impact caused by security incidents, Critical infrastructures (CIs) and Critical Information Infrastructure (CIIs) require advanced protection from cyber attacks. Success in information assets security depends on the capabilities of an organization to predict and prevent incidents. Therefore, organizations share information regarding ongoing cyberattacks and emerging threats, also providing strategies for their countermeasures.

Since the information that describes CIs and CIIs is often critical and may be used by third parties in organizing hacking campaigns, we focus here on the latest advanced security techniques and strategies for sensitive data protection. We consider information protection as one of the crucial aspects for the successful deployment of the CyberSANE platform. Therefore, this document provides the results obtained after an extensive study on the latest strategies regarding sensitive data protection, storing, and sharing in order to identify and use the most advanced approaches. After series of discussions, the research areas were selected in order to incorporate approaches we have in the technical WP7. The findings of this survey aim to act as the basis for the rest of Tasks. To achieve the goal for sensitive data protection, the presented strategies could be either implemented from scratch, or the capabilities of tools owned by the consortium may be enhanced through the development of additional features.

Contents

List of Figures

List of Tables

Chapter 1 Introduction

This deliverable presents the findings and outcomes of the work performed in Tasks 7.1, 7.2 and 7.3. We provide a detailed description of the state-of-the-art security methods and techniques for sensitive data protection. These approaches will be used to enhance the security capabilities of the CyberSANE. Different areas were covered, and the most advanced have been chosen after discussions at technical levels. Selected approaches aim at satisfying the needs and requirements of today's Critical Infrastructure (CI) and Critical Information Infrastructure (CII). Furthermore, we present our first technical activities and achievements related to the advanced data sharing and anonymization. The rest of the document is structured as follows:

- Chapter 2 overviews the state-of-the-art in privacy-preserving techniques describing anonymizations strategies and innovations in secure data storage;
- Chapter 3 overviews the latest encryption methods also providing a comparison between encryption algorithms and existing functions;
- Chapter 4 describes the state-of-the-art and the latest initiatives in blockchain technologies across different application scenarios;
- Chapter 5 presents a novel modelling language together with a process for cyber incident handling for incidents identification and handling;
- Chapter 6 describes the platform used for the advanced sharing and anonymization of information shared by organizations that operate within the CI and CII domains;
- Chapter 7 provides concluding remarks;
- Chapter 8 includes a glossary of used abbreviations;
- Chapter 9 concludes the deliverable with the bibliography.

Chapter 2 Privacy-Enabled Techniques

This chapter describes the latest initiatives for sensitive data protection that could find applicability in the context of the CyberSANE platform. The presented works focus both on widely adapted techniques and on frameworks for specific privacy and security needs. Since information that describes CIs and CIIs may include sensitive data, different access control techniques and anonymization methods are used to protect this information from a potential misuse.

2.1 Anonymity - the business of keeping data safe

Since the inception of the information age, companies and organizations in general have struggled to keep safe the entrusted personal data from their users, suppliers and employees. Data is key to improve decision making, but as we are still at the brink of the new age, the potential for insight gathering is still at its infancy. Decision makers more often than ever before rely in technical analysis for more accurate forecasting, making aware the risks that arise by keeping the raw data with inadequate protection mechanisms. Multiple cases of data leaks and exploitations have been on the news lately, of noteworthy I would like to provide to the reader the Equifax Breach from 2017[1], with a quote from the Electronic Privacy Information Center:

> *"Equifax, one of the three largest consumer credit reporting agencies in the United States, announced in September 2017 that its systems had been breached and the sensitive personal data of 148 million Americans had been compromised. The data breached included names, home addresses, phone numbers, dates of birth, social security numbers, and driver's license numbers. The credit card numbers of approximately 209,000 consumers were also breached. The Equifax breach is unprecedented in scope and severity. There have been larger security breaches by other companies in the past, but the sensitivity of the personal information held by Equifax and the scale of the problem makes this breach unprecedented." – epic.org*

Ignoring the multi-million dollars lost on government contracts and the millions paid out in the settlement fees, the leak was so severe that the impact reached the CEO who was replaced as the ultimate responsible for not applying industry best practices to keep their data safe.

The techniques that allow data owners to manipulate a dataset with sensitive information and transform it into one with privacy-preserving properties which can be safely shared with other parties have been available for a while now. Moreover, several publications on the subject can be found on the literature. These techniques are often applied in data subsets or only on record fractions, leading to cases where organizations release what they believe are anonymized datasets. However, after conducting a thorough and careful review, they can be re-identified with reasonable human or computer effort. In this report we will analyse the benefits of these techniques and provide valuable insights into their strengths and limitations.

[1] https://epic.org/privacy/data-breach/equifax/

2.1.1 k-anonymity

K-anonymity is a technique introduced by Latanya Sweeney in 1998 on her now famous paper 'Protecting privacy when disclosing information: k-anonymity'[2]. It was introduced as an approach to deal with data sharing on datasets that require structural integrity, partially due to relations with other datasets through links, or inside the proper dataset to maintain referential integrity. The key concept behind this technique is what has been known as hiding inside a crowd, this means that for k-anonymity to be valid there needs to be at least k entities with the same attributes in the target dataset. These attributes are usually the ones that could be used to target a specific entity, thus making sure that the generalization rules or anonymization rules applied, ensure at keeping secret the unique entity in a minimal k-sized group.

However, this technique alone is not enough to ensure that the dataset will always be simultaneously relevant and anonymized. One example is when the dataset universe is split among a set of characteristics and there are one or two entities which are very significant outliers. At this point, removing them makes the dataset irrelevant, while keeping them in makes it impossible to safely anonymize. It is easier to understand it with a personal example, Portugal as part of OCDE publishes regularly information about the innovation and R&D investment of the companies that operate there. But when these reports are split by industry type, a single entity dominates >90% of some of those indices. Because there is only one big Oil & Gas company in the country, since others are at best niche players with little to no investment or revenue. So, in such cases a decision must be made by a human, should we just remove the dataset altogether, or publish the anonymized version that will for an informed investigator be obvious that the data is reflective of that organization in particular?

Let us consider the following dataset that represent the Diseases at a local hospital for a given day.

Birthdate	Name	Sex	ZIP	Marital Status	Disease
09/11/1984	Joseph Silva	M	10249	Married	HIV
09/01/1978	John the Rock	M	10242	Single	HIV
01/06/1959	Nathalie Jones	F	10242	Married	Obesity
01/23/1954	Jeremy Angar	M	10249	Single	Hypertension
03/15/1953	Kat White	F	10212	Divorced	Hypertension
03/30/1938	Mika Jay	M	10249	Single	Obesity
09/18/1935	Maria Vaugh	F	10212	Divorced	Obesity
03/15/1933	Sandra Stones	F	10252	Divorced	HIV
01/02/2000	Kim Chan	F	10254	Single	COVID-19

Table 1: Sample dataset

[2] https://epic.org/privacy/reidentification/Samarati_Sweeney_paper.pdf

There are two common methods that are usually applied when achieving k-anonymity on a given dataset:

- Suppression – Suppression of method for replacing a given attribute with a static field such as "-" or "N/A", in the above dataset this would be applied to the Name field. It can also be applied by just removing the Field from the dataset, though this sometimes isn't possible due to historical reasons where the original pipeline for processing data requires a given structure. The application of this method to the above dataset produces **Error! Reference source not found.** below.

Birthdate	Name	Sex	ZIP	Marital Status	Disease
09/11/1984	*	M	10249	Married	HIV
09/01/1978	*	M	10242	Single	HIV
01/06/1959	*	F	10242	Married	Obesity
01/23/1954	*	M	10249	Single	Hypertension
03/15/1953	*	F	10212	Divorced	Hypertension
03/30/1938	*	M	10249	Single	Obesity
09/18/1935	*	F	10212	Divorced	Obesity
03/15/1933	*	F	10252	Divorced	HIV
01/02/2000	*	F	10254	Single	COVID-19

Table 2: Supressed Dataset

- Generalization – Generalization is the process of taking specific data, exploiting its structure or way it is created, and replacing it with a more generic concept that still holds some of the original intent. A typical example is geolocation, where replacing a street name with a city, country, or continent, might be a way to reduce the specificity of the data but still keep some relevance about location. In the above example, ZIP code can leak too much information, so its last character can be redacted by replacing it with a dummy character such as "-". Birthdays can also reveal a bit too much about the underlying subject, where a common generalization is to remove the day and month, and leave only the year. In some cases, it might be even needed to aggregate the years in a range such as 1980-85.

Birthdate	Name	Sex	ZIP	Marital Status	Disease
1984	*	M	1024*	Married	HIV
1978	*	M	1024*	Single	HIV
1959	*	F	1024*	Married	Obesity
1954	*	M	1024*	Single	Hypertension
1953	*	F	1021*	Divorced	Hypertension
1938	*	M	1024*	Single	Obesity
1935	*	F	1021*	Divorced	Obesity
1933	*	F	1025*	Divorced	HIV
2000	*	F	1025*	Single	COVID-19

Table 3: Generalization Example

There are attacks against k-anonymity which are more effective if there is a previous knowledge on the entities that are expected to be in the dataset.

The following examples are shortcomings for k-anonymity:

1. High-Dimension Datasets: It's known since at least the early 2000's that k-anonymity doesn't work well in high-dimension datasets. More recently De Montjoye, Yves-Alexandre, Cesar A. Hidalgo, Michel Verleysen, and Vincent D. Blondel [3] showed that with just 4 locations of carrier antennas, the spatio-temporal points are enough to uniquely identify 95% of the individuals in the paper "Unique in the Crowd: The privacy bounds of human mobility"
2. Dataset distortion: Olivia Angiuli, Joe Blitzstein, and Jim Waldo[4] showed that applying suppression and generalization techniques to ensure k-anonymity can skew the results when the distribution of the dataset is not uniform, with particular issues on datapoints that represent unique features.
3. Homogeneity attack & Background Knowledge: When suppression is applied to mask certain parts of a feature, to make all entries look the same, it can provide a false sense of security, since the remaining parts might be enough for an attacker with previous knowledge to know the right records by matching only the public available data.

2.1.1.1 l-diversity

l-diversity is an extension to k-anonymity which aims to make data more generic by forming groups through reduction of the granularity of the record features. While grouping gives extra protection and helps preserve the privacy guarantee of the anonymization processes used, it also reduces the usefulness of the dataset since it might remove its unique or useful characteristics. The core concept behind l-diversity is to make sure that each given record matches at least with k-1 other records. This trick helps to protect identities to the k-level individually, in short, it adds protection by providing intra-group diversity for critical values.

2.1.1.2 t-closeness

In 2007 Ninghui Li, Tiancheng Li, Suresh Venkatasubramanian introduced t-closeness[5] with the purpose of improving k-anonymity and l-diversity techniques. This can be seen as a refinement of l-diversity requiring that the distribution of a given feature in any group class is close to the distribution of the feature on the overall dataset.

2.1.2 Conclusion

The integrated CyberSANE framework will incorporate the PrivacyNet functionalities that will implement different anonymization techniques allowing the dissemination of privacy aware

[3] http://dspace.mit.edu/bitstream/1721.1/92263/1/Hidalgo_Unique%20in%20the%20crowd.pdf
[4] https://queue.acm.org/detail.cfm?id=2838930
[5] https://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.158.6171

datasets through the agreements managed by the ShareNet. The ShareNet agreements will enforce the executing of the anonymization techniques, which will be defined in a later stage of the project, by the PrivacyNet.

2.2 Innovations in privacy aware data storage, processing and sharing

Cyber Threat Intelligence (CTI) sharing between trusted entities became an invaluable technique used by security specialists to address emerging and ongoing threats. It allows organizations to inform each other about cyber incidents that they face or may encounter in future, and describe countermeasure strategies to prevent these incidents or mitigate their negative impact on a system. However, information exchange itself is a prone process that comes together with multiple challenges, including automation, standardization, and more importantly, protection of private and confidential data. Nevertheless, initiatives towards the standardization of CTI have been already proposed and are currently used by numerous tools and platforms to produce, represent, and share CTI in an automated manner, making the sensitive information protection to fell on data owner shoulders. However, enforcement of fine-grained security policies can protect sensitive data from unauthorized access and potential abuse. Furthermore, security policies should define a list of anonymization operations that should be executed before providing access to CTI. In addition to this, it is crucial to enforce security policies during the whole usage of CTI, and revoke and terminate data usage if security policies are not satisfied anymore. In this section, we now detail some control mechanisms for data access and usage.

2.2.1 Data Access and Usage Control

Controlling access to physical and information assets is a crucial task for many organizations starting from Small and Medium Enterprise (SMEs) up to international corporations. Access control itself is a security mechanism used to assure that only trusted principals are granted to access a resource (Abadi 2003). Another definition given in (Shirey 2007), defines access control as "a process by which the use of resources is regulated according to a security policy and is permitted only by authorized users, programs, processes, or other systems according to that policy". In practice, access control models rely on and accompany with other security mechanisms in a computer environment (Samarati 1994) including authorization database, auditing systems, etc. Access control is enforced by a component known as a reference monitor that mediates every subject's access attempt to objects within an ecosystem. This component communicates with the authorization database that includes security policies to determine if the user attempting to do the operation is authorized to perform it or not.

Starting from the Lampson's matrix (Lampson 1974) introduced in late 1960's, many access control models have been proposed. However, in practice only Discretionary Access Control (DAC) (Samarati 1994), Mandatory Access Control (MAC) (R. Sandhu, Lattice-based access control models 1993) and Role-Based Access Control (RBAC) (David Ferraiolo 2001), (Ravi Sandhu 1996) achieved success. Meanwhile, those traditional access control models (Pierangela Samarati 2000) check whether subjects hold the proper rights before granting them the access to the requested objects. In fact, other access control approaches provided by Context Aware Access Control (CAAC) (Guangsen Zhang 2004.), Task-Based Access Control (TBAC) (R. K. Sandhu 1998) and Risk-Adaptable Access Control (RAdAC) (Farroha 2012) models are also used in security administration. Among all of them, the RBAC and ABAC are the most widely used approaches found on real-world applications.

2.2.1.1 Role-Based Access Control

Security administration in organizations with a large number of employees is a complex process that requires a security specialist to define specific access rights for different users. Security administrators tend to make use of an approach provided by the RBAC model in order to simplify this process, since various users may be assigned to the same role and thus have different privileges. Hence, the most important concept in RBAC model is the role, which is a grouping mechanism used to categorize subjects based on various properties (R. K. Xin Jin 2012). The role may arise from the hierarchy of the organization, while each employee, i.e., subject, may be assigned to one or multiple roles, thus having different access privileges (Khambhammettu 2008). Moreover, RBAC considers the usage of groups, privileges groupings (Baldwin 1990), (Thomsen 1990), and separation of duty concept (Wilson 1987), (R. Sandhu, Transaction control expressions for separation of duties 1988), (Nash 1989).

Several extensions to RBAC by combining attributes and roles have been proposed. Some works defined parameterized privileges for restricting access to a subset of objects (Iglio 1997), (Khayat 2004), (Evered 2003), while other works proposed to consider object sensitive role (Jeffrey Fischer 2009) and attributed role access control (Christian Schläger 2006).

Despite benefits and advantages comparing to other traditional access control approaches (Loomis 2010), the RBAC model has limitations regarding contextual information starting from time and location up to environmental-specific conditions like temperature, pressure, available amount of money of the user's credit card, etc. To overcome limitations existing in the RBAC model, a new approach, known as ABAC was introduced.

2.2.1.2 Attribute-Based Access Control

The Attribute-Based Access Control (ABAC) model (Sylvia Osborn 2000), (R. K. Xin Jin 2012), (Vincent C. Hu n.d.) became a promising approach for security administrators in defining access restrictions to resources in various infrastructures (M. G. Sandhu 2016), (R. K. Xin Jin 2012), (D. Richard Kuhn 2010), (Maanak Gupta 2018). This model is a result of the approach that encompasses the benefits of traditional access control models including aforementioned DAC, MAC, and RBAC, whilst surpassing their limitations. Literature provides several definitions of the ABAC model (Lingyu Wang 2004), (Isabel F. Cruz 2008), (Tong 2005). However, one of the most consummate definitions that cover all aspects of ABAC model was given by the National Institute of Standards and Technology (NIST) defining it as "*an access control method where subject requests to perform operations on objects are granted or denied based on assigned attributes of the subject, assigned attributes of the object, environment conditions, and a set of policies that are specified in terms of those attributes and conditions*" (Vincent C. Hu n.d.).

The ABAC model instead of relying on predefined roles, uses *attributes* to express identities, clearances, sensitivity and other properties of entities (e.g., users, subjects, objects) as well as operational environment (e.g., time, location, etc.). Hence, the ABAC allows the modelling of policies which take into consideration the contextual information that may affect the decision-making. The core components of the ABAC are the following:

- **Attributes** are characteristics used to define specific aspects of the subjects, objects, and environment conditions. Every attribute is a property expressed as a “name:value” pair associated with any entity in the system.
- A **subject** is an entity that requests to perform an operation upon the object. Attributes of subjects may describe their names, ID numbers, affiliation to an organization, location, IP addresses, etc.
- An **object** is an information system-related entity that contains or receives information. It can be the resource entity (e.g., files, records, tables) as well as anything upon which a

subject may request to execute an operation (e.g., applications, services, devices). Attributes of the object may describe its type, capacity, sensitivity, location, etc.

- An **operation** also sometimes referred as *action*, represents an execution of a function that the subject requests upon an object. Operations may vary from simple functions like read, write, delete up to the execution of specific processes.
- A **policy** is a collection of *rules* that determine the set of permissible operations, which a subject may execute on an object in predefined environment conditions.

Since the ABAC model relies on attributes for describing entities, it avoids the need of assigning directly and explicit authorizations to individual subjects, before any request to perform an operation on the object (Vincent C. Hu n.d.). Furthermore, the ABAC model provides a flexible approach for large enterprises, since access control management is often a time-consuming and sophisticated process due to the large Access Control List (ACL) as well as a variety of roles and groups considered in security policies.

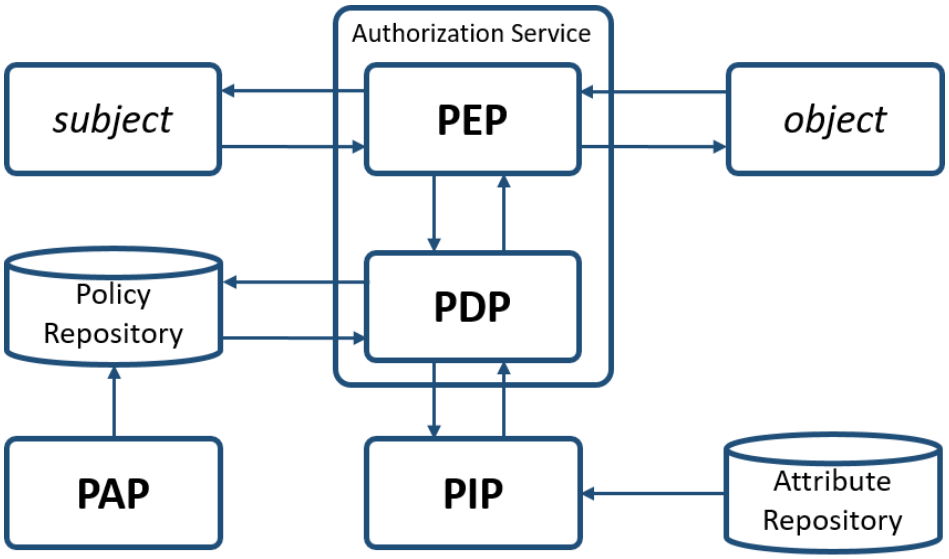

Figure 1: Typical Access Control Mechanism

Typical Access Control Mechanism includes the multiple functional elements shown on Figure 1 and are better known as *points* (Vincent C. Hu n.d.). These points are designed to handle specific operations, including retrieval and management of security policies, access requests evaluation, and attributes retrieval and assessment. Each functional point of the ACM is defined as follows:

- **Policy Decision Point** - a component of the ACM that computes the access decision;
- **Policy Enforcement Point** - a component which either gives or denies access to the resource;
- **Policy Information Point** - a component that enables ACM to retrieve attributes or another data required for the policy evaluation;
- **Policy Administration Point** - a component that serves as a user interface that allows creating and managing security policies.

Depending on security needs, size of an organization, and application of the ACM, its implementation may have multiple elements with the same functionalities. However, the main objective will remain unaltered.

2.2.2 Usage Control

This section describes the usage control model proposed by R. Sandhu and J. Park referred as UCON. UCON enhances traditional ABAC (Sylvia Osborn 2000), (R. K. Xin Jin 2012), (Vincent C. Hu n.d.) model providing continuity of control also considering mutability of attributes (J. P. Sandhu, The UCONABC usage control model 2004). Hence, values of attributes used for the decision-making process are mutable and can change over time. Furthermore, attribute value changes might affect the entire security policy enforcement, allowing thus the re-evaluation of the request and possibly the revocation of previously granted access. The continuity of control means that access decisions are evaluated before granting access and during access rights execution on a resource. Thus, if attribute values change while the access is under process and new values do not satisfy the security policy anymore, then the system with the implemented UCON paradigm revokes the granted access rights and terminates the usage of the resource.

Both in ABAC and UCON models attributes of the entity that requests the access, resource, and environment are used to evaluate a request to access resources. Therefore, in the UCON model there are multiple components which represent the resource (*object*) to be protected, entities that issue requests (*subjects*) to access and execute some *access rights* on resources.

- **Subjects**. A subject is an entity that requests an access to a resource and executes granted access rights on requested resources (J. P. Sandhu, The UCONABC usage control model 2004). In the UCON model, a subject is represented by a set of corresponding attributes, ATT(S), which may define subject's characteristics, properties, and capabilities (e.g. ID, affiliation, role, location).
- **Objects**. Objects represent resources that subjects can access or use. Depending on the application of the UCON model, objects can be of various types starting from files, network sockets up to high-level services, low-level computational resources, etc. Same as subject, objects in UCON are characterized by a set of corresponding attributes denoted as ATT(O) and may vary from, the type, computational capability, security label assigned, etc.
- **Attributes**. Additionally, to attributes of subjects and objects, the UCON model defines environmental attributes, denoted as ATT(E), which are system-central characteristics about the computational environment, in which a subject and an object operate. The most common environmental attribute is a system time.

The main novelty of the UCON model is the mutability of attribute values. This aspect is also a backbone of the model since changes of attribute values may affect previously taken access decisions in a sense that the system with the enabled UCON paradigm will re-evaluate the request against security policies. Although depending on the application domain of the UCON model, the number and the type of attributes may be different, there are only three main reasons that cause changes in attribute values. Thus, attribute values change may be caused by the nature, by activities of subjects and objects and attribute values can be modified as the result of access. Moreover, the UCON model specifies two main categories as *mutable* and *immutable* e.g., time and subject's ID respectively. As stated in (J. P. Sandhu, The UCONABC usage control model 2004), mutable attributes are categorized as follow:

- **Exclusive/Inclusive attributes** which are used to resolve conflicts of interests, e.g. dynamic separation of duty;
- **Consumable attributes** which are destroyed as the result of a security policy enforcement;
- **Immediate revocation attributes** which terminate access if an attribute value changed to a certain number, e.g., time, amount of money available on the back account;
- **Obligation attributes** are attributes whose values change as the result of *obligation* actions fulfilment.

Considering the time validity, attributes may be temporary or local, and thus valid only for one access, as well as persistent or global meaning that those attributes may be valid for many accesses. The work in (J. P. Sandhu, The UCONABC usage control model 2004) provides a more detailed classification.

- **Rights**. As any other access control model, UCON access rights denote permissions which subjects may exploit on objects (J. P. Sandhu, The UCONABC usage control model 2004). However, the main difference between UCON and traditional access control models is a long-lived access right in the UCON model.

Additionally, to the components described above, UCON defines three decision factors, namely *authorizations*, *conditions*, and *obligations*, which affect the result of evaluation of the request. While authorizations define predicates which put restrictions on attributes of the subject and/or object, conditions are environment constraints that must be valid before or during the usage of a resource.

Differently to traditional access control models, UCON authorizations are evaluated before granting access as well as during access rights execution, and thus called pre-authorizations and on-authorizations respectively. Moreover, since changes in attribute values of the operational environment may affect the decision-making, conditions must be considered as well.

Same as the authorization predicates, condition predicates are evaluated before granting access to an object and/or during the usage of the object by the subject, and thus called pre-conditions on-conditions respectively. Furthermore, values change in environmental attributes can happen as a result of environmental modifications, e.g., temperature, pressure, the number of subjects simultaneously accessing the object, etc. Another novelty introduced by the UCON model is the presence of obligations, which term was firstly introduced by Lockman and Minsky in their work "Ensuring integrity by adding obligations to privileges" (Lockman 1985). Obligations verify whether the mandatory task or actions relevant to the usage of resource were fulfilled, and according to (Basel Katt 2008), UCON obligations can be fulfilled before, during, and after access rights execution depending on the application domain. Furthermore, in UCON, obligations are defined as a tuple OBL = (OBS;OBO;OBA;WHEN;DURATION), where OBS and OBO refer to the obligation subject and obligation object respectively, OBA is the action that has to be performed (e.g., delete, send notification, etc.), WHEN addresses when obligations should be fulfilled, i.e., before the access, during the access or after the access, and DURATION defines an interval in which obligations must be fulfilled.

The ShareNet component of the CyberSANE platform will realize the UCON paradigm to enable ongoing control on data usage and to ensure that only authorized entities can access data. Furthermore, ShareNet will interact with the PrivacyNet component to execute anonymization operations on specified information according to security constraints defined in the corresponding policy. Chapter 6 describes this in more detail.

2.3 Privacy by design and by default

Building on the legal analysis outlined in “D2.2 - Legal and Ethical Requirements”, this section intends to take a closer look at the principle of privacy by design and by default through the lens of the European Data Protection Board (EDPB)’s Guidelines on Data Protection by Design and

by Default[6] (EDPB, Guidelines 4/2019, adopted on 20 October 2020). The Guidelines illustrate how controllers should implement the Data Protection by Design and by Default (DPbDD) principles set out in article 25 GDPR, which are considered as complementary and mutually reinforcing concepts. The Guidelines also offer important clarifications on how to combine DPbDD with the principles listed under art. 5 GDPR.

As explained in D 2.2, taking into account the DPbDD principles from the design phase will support and enable compliance by the end users of the CyberSANE system, which will act as controllers under GDPR. DPbDD (and related Guidelines) are most relevant to the development of the CyberSANE system: as recommended by the EDPB, producers and processors should be proactive in ensuring "state of the art" standards (for instance for privacy-enhancing technologies) and prove to controllers "how their hardware, software, services or systems enable the controller to comply with the requirement to accountability in accordance with DPbDD", including through key performance indicators.[7]

2.3.1 Article 25(1): Data protection by design

The elements that the controller has to consider when defining the data protection by design measures applicable to a specific operation are the following:

a) *state of the art*: it is a dynamic concept, which evolves through time and requires periodic re-evaluation *vis à vis* technological innovations. It applies both to technological and organizational measures;
b) *cost of implementation*: as underlined by the EDPB, the cost element is a factor to be taken into consideration when implementing the data protection by design principle, but not a justification for not adopting data protection by design measures. In other words, the controller can legitimately opt for a less
c) *nature, scope, context and purpose of processing*: according to the Guidelines, the nature of the processing concerns intrinsic features of the processing (including for instance special categories of data and imbalance of power between data subject and controllers); the scope relates to the size and width of the processing operations; the context and purpose refers respectively to the circumstances surrounding the processing and the goals of said processing. In the assessment and interpretation of these four elements, consistency with other GDRP provisions, such as art. 24, 32 and 35 GDPR must be ensured.
d) *risks of varying likelihood and severity for rights and freedoms of natural persons posed by the processing*: a risk-based approach underlies compliance with art. 25 GDPR, as well as with art. 24, 32 and 35 GDPR. These provisions require a coherent approach in carrying out case by case assessments of data protections risks, primarily through Data Protection Impact Assessments (DPIAs).

Implementation of data protection by design must take place during the time of determining the means for processing, which include "the architecture, procedures, protocols, layout and

[6] EDPB, Guidelines 4/2019 on Data Protection by Design and by Default, adopted on 20 October 2020, available at https://edpb.europa.eu/our-work-tools/public-consultations-art-704/2019/guidelines-42019-article-25-data-protection-design_en
[7] Guidelines 4/2019, p. 30.

appearance" of processing. However, the controller has an obligation to regularly review and re-assess the effectiveness of the adopted measures *vis à vis* levels of risks which might change throughout the processing.

2.3.2 Article 25(2): Data Protection by Default

The principle of data protection by default, set out by art. 25(2) GDPR, mandates that - by default - processing is limited to what is strictly necessary to achieve a pre-determined lawful purpose. Consequently, the amount of data collected, the processing operations performed on such data and the period of storage must not exceed what is strictly necessary for the specific processing purposes. Hence, as art. 25 GDPR requires the end-users of the CyberSANE system to assess the data protection risks connected to the adoption of the system, CyberSANE must allow the de-activation of the features which are incompatible with the data minimization by default principle. Specifically, the data minimization obligation pursuant to art. 25 GDPR concerns:

- the amount of personal data collected;

- the extent of their processing;

- the period of their storage;

- their accessibility.

2.3.3 DPbDD and the principles under art. 5 GDPR

Part 3 of the Guidelines, which illustrates how to achieve DPbDD through the principles set out under art. 5 GDPR - lawfulness, transparency, purpose limitation, data minimization, accuracy, storage limitation, integrity, confidentiality and accountability - is particularly important for a successful development and implementation of the CyberSANE system. According to the EDPB, DPbDD must be operationalized and integrated in all the principles outlined in art. 5 GDPR to achieve the effective implementation of such principles. The following sub-paragraphs provide an illustration of the measures suggested by the EDPB to ensure that compliance with the principles under art. 5 GDPR is aligned with DPbDD values.

2.3.3.1 Lawfulness

The end-users of the CyberSANE system will consider the following key DPbDD elements to comply with the lawfulness principle[8]:

- determining the correct legal basis for the processing and make sure that each processing operation has its own legal basis;
- clear link between a legal basis and a specific processing purpose (purpose specification);
- Processing must be necessary (necessity);
- the data subject must be as autonomous as possible in exerting control over his or her data (autonomy);

[8] Guidelines 4/2019, p. 16.

- the provision of consent must align with the requirements set out under art. 7 GDPR, as further specified in the Guidelines 5/2020 on consent;
- re-adapting the processing following a change of legal basis changes;
- power imbalances and vulnerabilities of the data subject must be duly considered when legitimate interest is the legal basis for processing (balancing of interest);
- in case of joint controllership, the respective responsibilities of the controllers must be defined transparently.

2.3.3.2 Transparency

Key DPbDD elements for transparency - to be considered when, for instance, setting up a privacy policy section on the end users' websites - include the following[9]: clarity; accessibility; universal design; intelligibility; plurality of channels and media; plurality of layers.

2.3.3.3 Fairness

The following elements are among the key DPbDD features when implementing fairness[10]:

- the data subject must be as autonomous as possible in exerting control over his or her data;
- the processing should align to and meet the reasonable expectations of the data subject, avoid any discrimination and exploitation of data subjects, particularly when in a state of vulnerability;
- qualified human intervention for the purposes of art. 22 GDPR must be envisaged by the controller;
- data subjects must be duly informed about the use of algorithms in assessments or predictions concerning their personal situation and behaviour (e.g., health, work performance, location, preferences, etc) and controllers must review such algorithms on a rolling basis against unfairness and biases.

2.3.3.4 Purpose limitation

Key DPbDD elements for purpose limitation which must be considered in the design of the CyberSANE functionalities include:

- definition of the specific purposes must precede the processing design and orient such process (predetermination and purpose orientation);
- new processing purposes must be compatible with the original purpose of processing data; controllers should adopt encryption and hashing methods to prevent the repurposing of data (purpose compatibility and limitation of re-use)

2.3.3.5 Data minimization

[9] Guidelines 4/2019, p. 15.
[10] Guidelines 4/2019, p. 17.

Data minimization translates the principle of necessity, according to which processing must be limited to the personal data that is adequate, relevant and necessary for the purpose of processing. The following measures are among the key DPbDD elements as regards compliance with data minimization[11]:

- limitation, relevance and necessity to the purpose;
- pseudonymization of personal data, with separate storage of identification keys;
- anonymization and deletion of personal data as soon as not (anymore) necessary for the processing purpose;
- state of the art technologies for data avoidance and minimization.

2.3.3.6 Accuracy

Key DPbDD when implementing accuracy are the following[12]:

- verification of the correctness of the data at different times into the processing, including by granting data subjects of effective access to personal data;
- erasure and rectification of inaccurate data without delay;
- mitigation of error propagation;
- adopting design features to limit inaccuracy.

2.3.3.7 Storage limitation

Key DPbDD in the implementation of the storage limitation principle are the following[13]:

- effective deletion and anonymization;
- capability to justify and disclose rationale behind envisaged storage period, including for back-ups and logs;
- enforcement of retention policies and internal testing about compliance of the organization with such policies;
- limit the flow and temporary storage of copies of personal data.

2.3.3.8 Integrity and confidentiality

As sensitive incident-related information will be managed, stored and exchanged across CIIs, including for forensic purposes, the integrity and confidentiality principle is particularly relevant to CyberSANE. Key DPbDD measures listed by the EDPB include[14]:

- implementing security requirements from early stages of design;
- defining information security management plans;
- risk assessments on the security of personal data, including through “threat modelling” and analysis of the vulnerabilities of a software;
- regular checks on the resilience of the security system;

[11] Guidelines 4/2019, p. 21.
[12] Guidelines 4/2019, p. 23.
[13] Guidelines 4/2019, p. 25.
[14] Guidelines 4/2019, p. 26-27.

- defining access limitation policies;
- secure storage, with assessment of risks associated with centralized and decentralized storage for different types of personal data;
- limit back-ups and logs to what is needed for information security;
- disaster recovery and business continuity plans;
- incident management procedures, including notification to the data protection authority and to the data subjects.

The Guidelines 1/2021 on Examples regarding Data Breach Notification[15], recently issued by the EDPB, provide practical and case-based guidance for compliance with data breach notification and communication requirements. The Guidelines are of outmost relevance for the CyberSANE end-users as they look at the organizational and technical measures for preventing and mitigating the impact of several types of ransomware and data exfiltration attacks, as well as breaches due to an internal human source, to the loss or theft of devices, or to mispostals.

[15] EDPB, Guidelines on Examples regarding Data Breach Notification, adopted on 14 January 2021, versio 1.0, available at https://edpb.europa.eu/sites/edpb/files/consultation/edpb_guidelines_202101_databreachnotificationexamples_v1_en.pdf

Chapter 3 Encryption Methodologies

The enormous critical information sharing created a vital need for its protection and use of different security mechanisms. A variety of encryption methodologies are widely used to protect data from alteration and abuse. Existing encryption techniques have their pros and cons and require different computational capabilities. This chapter overviews well-known and widely adapted encryption methodologies. It describes hash and encryption algorithms, provides an overview of homomorphic and attribute-based encryption techniques, as well as format-preserving encryption algorithms. The presented methodologies have the potential to be used to achieve the security goals of the CyberSANE project.

3.1 Hash Algorithms

Hash algorithms are both widely used in computer science, but there is a difference between a standard hash and a cryptographic hash. The defining difference is that a cryptographic hash function should at least have the property of being one-way. Also, there is a difference between algorithms and functions. Hash algorithms define how the hash function will be used in terms of how the message will be broken and how the message blocks will communicate with each other. On the other hand, hash functions just generate the hash code. There are two groups of hash functions, those with an input parameter and those without one. The hash functions that use a parameter (also called keyed hash functions) have two inputs, a message and a secret key, while the other group just receives a message as input. The first group is usually used to ensure and verify the authenticity of a message, while the second group is usually used for the verification of the data integrity.

The cryptographic hash function consists of an efficient algorithm that takes as input any sequence of a finite bit-length stream and outputs a hash (or digest) which is a binary sequence of fixed bit length (usually around 160 and 512 bits). That output can be seen as the unique fingerprint (representation) of a message. Hash functions are widely used as an essential part of digital signature, password hash storing, and message authentication codes. The usage of hash algorithms can be found in many protocols since they are an important cryptographic primitive. A cryptographic hashing algorithm/function must meet some important requirements:

- First, when given a hash value it should be infeasible to reverse engineer and find the original message. This is known as the Preimage Resistant property.
- Second, the produced hash value should be unique for any different input (hash collision).
- Finally, the Collision Resistant property that states that it should be infeasible to find two messages with the same hash.

The most widely deployed hash functions (like the MD-5 SHA-1, SHA-2 and RIPEMD-160) are those that are based on the Merkle–Damgård construction. The output sizes of those algorithms differ. MD-5 algorithm provides a constant 128 bits output while RIPEMD-160 and the SHA-1 produce a 160 bits output. Finally, the SHA-2 which is a group of three algorithms (SHA-256, SHA-384 and SHA-512) outputs either an output of 256, 384 or 512 bits depending on the algorithm. The ancestor of those hash functions is an earlier simpler algorithm called MD-4.

3.1.1 Message Digest (MD) hash functions

The MD family contains many hash functions like the MD2, MD4, MD5 and MD6. Out of those, the well-known MD5 hash function stands out, which is a corrected version of the MD4 (R. L. Rivest 1991) algorithm, in which some weaknesses were shown, was introduced by Rivest (Dusse 1991) and there was no proof that it is a good one-way cryptographic hash function. MD4 is not recommended for usage anymore because it's prone to collisions, which means that the Collision Resistant property can easily be broken. MD5 receives as an input a bit sequence (message) of an arbitrary bit length and output a 128-bit hash. That procedure is done in four steps, in each step the data are processed in 512-bit blocks divided into sixteen 32-bit words. The advantages of MD5 are that is fast to compute, it has some collision resistance and that it provides a one-way hash. The disadvantages are it has known security flaws and vulnerabilities and that SHA-1 is safer.

3.1.2 Secure Hash Function (SHA)

The SHA family consists of four algorithms, the SHA-0, SHA-1, SHA-2 and SHA-3. They may belong to the same group of functions, but they have structural differences between them.

The unpopular SHA-0 algorithm was designed by the National Institute of Standards and Technology (NIST) in cooperation with NSA and published as a federal standard in 1993. Later, the improvement of SHA-0 arrived, the SHA-1 which is used in many protocols including the Secure Sockets Layer (SSL). The algorithm works in a similar way to the MD5; thus, a message (of any length less than 264 bits) is used as an input, and the algorithm outputs a 160-bit hash value.

The collision resistance of SHA-1 was limited for higher security levels, NIST introduced in 2001 some variants of the SHA-1, the SHA-2 family. That family had at first three variants of the SHA-1 which are the SHA-256, SHA-384 and SHA-512. Later the SHA-224 arrived because it was designed to work better with the 3DES. The SHA-2 family is not preferred for checking the data integrity because it lacks the operational speed of SHA-1.

SHA-3 was proposed in 2012 coming as an output of a competition that was organised by NIST. The Keccak algorithm that won the competition became the new SHA-3 standard. It has a different internal structure and it supports all the hash length variants of SHA-2. Security-wise it is not vulnerable to length extension attacks like the other SHA members and MD5 and it has good performance and resistance to attacks. The different structure of the SHA-3 provides the flexibility to operate on much smaller states, which makes it ideal for embedded systems or smart devices with limited resources like memory and energy.

Table 4 shows the comparison of MD5 and SHA algorithms in terms of output size, rounds and collision status that shows if the algorithm is prone to collision attacks. The number of rounds indicates how many times the algorithm will use the hash function. Every round has an input of a fixed size which is a combination of the previous round and the most recent message block.

Name Of The Algorithm		Size Of Output	Rounds	Collision Status
MD5		128	64	Yes
SHA-1		160	80	Yes
SHA-2	SHA-224	224	64	Theoretical

	SHA-256	256	64	No
	SHA-384	384	80	No
	SHA-512	512	80	No
SHA-3		256/512	24	No

Table 4: Comparison between MD5 and SHA hash algorithms

The minimum output length that a hash functions should have in order to withstand collision attacks is 160-bit output length. More bits like 256 bit or higher will provide long-term security. MD5, that have been widely used it was proven insecure while serious security weaknesses have been found also in SHA-1.

3.2 Encryption Algorithms

The high-level overview of cryptographic encryption algorithms can be observed in the figure below (Figure 2). There are two main categories of key-based cryptography, the asymmetric and symmetric key encryption which are described in the subsections below.

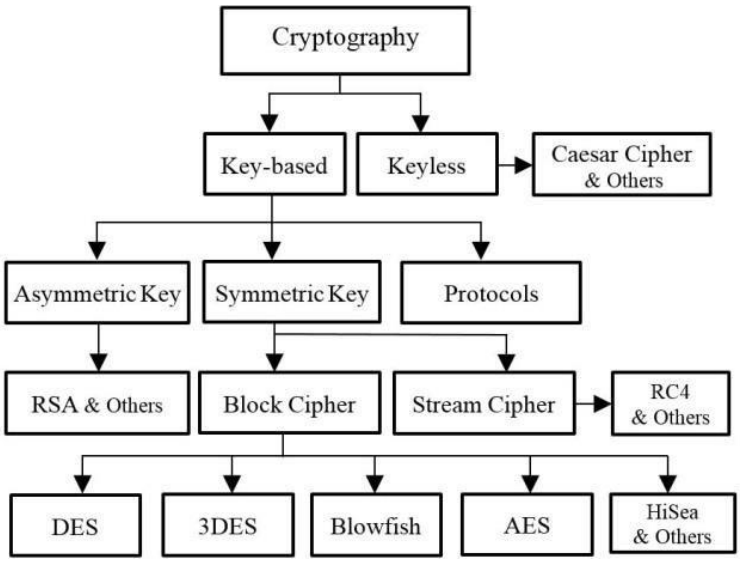

Figure 2. Cryptographic encryption algorithms overview. (Mushtaq 2017)

3.2.1 Symmetric Key encryption algorithms

Symmetric-key encryption provides secrecy between the communications of two parties. Anyone who tries to intercept a message should not be able to see the original message or get any significant information about its content. The preparation for this secure communication channel lies in the mutual pre communication agreement on a common encryption key. This key should only be known by the involved parties of the communication. The reason that this type of encryption is called symmetric lies in the fact that the involved actors use the same key for encryption and decryption. The algorithms used for encryption and decryption are publicly known. That means that anyone who knows the key can encrypt and decrypt a message. Therefore, the

key used in communication must be kept secret. That was the basic problem in the symmetric encryption scheme, the way of sharing the key in a secure and efficient way. The solution came with the discovery of public-key cryptography.

3.2.1.1 Data encryption standard (DES)

The Data Encryption Standard (DES) (Technology 1999) was introduced by NIST and it was the most widely used symmetric-key encryption algorithm. It is a block cipher symmetric-key algorithm that is based on the Feister Cipher. DES declares a key length of 64-bits but in reality, only 56 of them are effective due to the 8-bit usage for error detection. Governments, banks and applications in commerce took the DES as the basis for secure and authentic communication.

3.2.1.2 Triple DES

The computational power increased since DES release thus brute force attacks where feasible. This increased the need for improvements. Triple DES (Mouha 2017) was designed in order to replace the original DES algorithm, which attackers could easily break. Triple DES secured itself by increasing the key length instead of design a complete block cipher and with that way, it's protected against brute force attacks. It uses 168 bits key (3x56) or 112 bits (2x56), has 48 rounds and a block size of 64 bits.

3.2.1.3 International Data Encryption Algorithm (IDEA)

International Data Encryption Algorithm (IDEA) is another symmetric key block cipher that was developed by Xuejia Kai and James Massey in 1991. It uses a block size of 64 bits, has 8 rounds and a key size of 128 bits. Security-wise some attacks like meet-in-the-middle attacks (Eli Biham 2015) are proven to break the algorithm. In general, there are better and faster algorithms.

3.2.1.4 Advanced Encryption Standard (AES)

The more popular and widely used symmetric-key encryption algorithm is the Advanced Encryption Standard (AES) (Morris J. Dworkin 2001). In comparison with the triple DES, it was found to be at least six times faster and stronger. The principles followed by this algorithm when designed were to have compact code, speed on the many infrastructures, simple design and protection against all attacks that were known. AES uses a symmetric key block cipher, takes as an input data of 128bits and utilises the 128, 192, or 256 bits key. The number of rounds that this algorithm uses depend on the key size. AES will use 10 rounds for 128-bit keys, 12 rounds for 192-bit keys and 14 for 256 keys. Each round calculates and uses a different 128-bit key that comes from the computation of the original key.

3.2.1.5 Blowfish

Blowfish (Schneier, Description of a new variable-length key, 64-bit block cipher (Blowfish) 1993) was introduced by Bruce Schneier in 1993 and is a symmetric block cipher based on the Feistel function. Blowfish in contrast to almost all the other algorithms is license and patent-free, which means that it is freely available for everyone. Its structure is that it uses a key with length from 32 - 448 bits and has a 64 bits block. Blowfish algorithm consists of 16 rounds for the encryption process, it uses round keys and the generation process of each key increases the complexity and

the safety of this algorithm. Thus, it protects itself from a brute force attack and supposedly provides better security than existing encryption techniques.

3.2.1.6 Twofish

Twofish (Schneier, The Twofish encryption algorithm 1998) is another symmetric block ciphering algorithm that haves a similar structure as Blowfish. It made its first appearance in 1998 and it was originally created by Schneier. Twofish structure utilizes block figuring like Blowfish as well. This algorithm gives the ability to adjust encryption speed, key setup time, and code size before execution. Twofish is also unlicensed and patent-free which makes it openly accessible. It utilizes key sizes of 128, 192 and 256 bits with a block size of 128 bits and 16 rounds.

3.2.1.7 Threefish

Threefish (Niels Ferguson 2010) is the third symmetric key block ciphering algorithm that was released in the year 2008 by Schneier, et al. Threefish belongs to the same family of algorithms like Blowfish and Twofish. The difference is that Threefish uses three different types of key the 256, 512 or 1024 bits. It has a block size the same as the size of the key with 72 rounds for the first two (256 and 512 bits) and 80 for the last (1024 bits). Another difference for the other algorithms is the ability to tweak cipher block meaning that it takes three parameters as an input. Those parameters are the key, a tweak value and a block of message. The encryption of the block message is achieved from the tweak value.

3.2.1.8 Hybrid Cube Encryption Algorithm (HiSea)

Hybrid Cube Encryption Algorithm (HiSea) (Sapiee Jamel 2011) was developed by Sapiee Jamel in 2011 and is a symmetric block cipher algorithm. This is an enhanced cipher method because it combines the advantages of the public along with symmetric algorithm elements. However, a disadvantage of a hybrid encryption algorithm lies in the secure allocation of keys to the involved parties in the communication. On the other hand, this algorithm is resistant to attacks and compared to other algorithms it doesn't make the ciphertext longer. So, it is a safe and secure option as long as the distribution of the keys is done in a safe manner.

3.2.1.9 The Rivest Cipher (RC) algorithms Family

RC1 was the first draft of what Rivest had in mind for a symmetric key algorithm. Later, different variants of that draft were designed and implemented along with research by the science community. The main pillar of RC was the design of a Symmetric Key encryption algorithm that could be used by the users to protect their data as they travel throughout the network.

RC2 (Lars R. Knudsen 1998) is a block encryption algorithm, developed in 1987 that was designed to replace the DES algorithm. It is a secret key block encryption algorithm that uses flexible key values from 1 byte to 128 bytes. The input and output blocks have a size of 64-bit each. Also, they designed this algorithm for easier implementation on 16-bit microprocessors.

RC3 was never released because it was broken before ever being used on the development process at RSA security.

RC4 (Stallings 2005) which is the only stream cipher of the family, is a symmetric key encryption algorithm. It uses key sizes of 40–2048 bits. The data stream is fused with the generated keys with the XOR operation. Protocols like the Wireless Equivalent Privacy (WEP) uses the RC4

algorithm for confidentiality but it can also be used by many other email encryption services. The cipher can be expected to run very quickly in software. It was considered secure until some test attacks exposed some vulnerabilities.

RC5 (R. L. Rivest 1994) is a 32/64/128-bit block cipher developed in 1994 by Ronald Rivest for RSA Data Security. The characteristics of RC5 are that it is simple, fast and consumes less memory. It is a symmetric block cipher that have parameters like the number of rounds (0-255) the key size (0-255) and the size of the block. The selection of the key is important because if it's long enough it can be considered safe rather than using a short key size which will make the algorithm weak to attacks. Thus, the security depends on the parameters that are chosen.

RC6 (M. J. Ronald L. Rivest 1998) was an AES finalist that was developed in 1997. It is a block cipher that uses 128-bit block size and supports key sizes of 128, 192 and 256 bits with 20 rounds. It was based on the RC5 and the idea was to improve the RC5 and also meet the requirements of the AES. So, the concepts of data-dependent rotations, modular addition and XOR operations came from the RC5. Security-wise is stronger to attacks that may break the RC5. It makes use of 4 registers (Each one of 32 bit) and is more secure than the RC5. It is also protected from various other possible security attacks. It uses fewer rounds and offers higher throughput.

3.2.2 Asymmetric Key encryption algorithms

Asymmetric key encryption algorithms provide a bid advance compared with the symmetric key encryption algorithms which is a way of sharing the secret key in a secure and efficient way. Asymmetric key encryption algorithms use two distinct, yet related keys. The first key, known as the Public Key, is used for encryption while the other, known as the Private Key, is used for decryption. As implied in the name, the Private Key is intended to be private so that only the authenticated recipient can decrypt the message (papers 2003-2004). Figure 3 presents the main concept of the asymmetric key encryption algorithms. The most common algorithms used in the asymmetric key encryption are presented below.

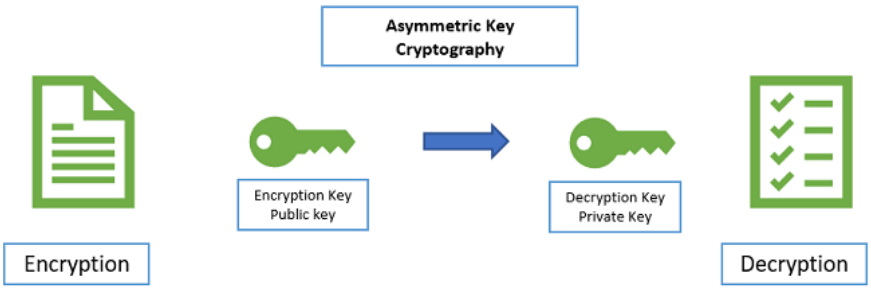

Figure 3 Asymmetric key encryption algorithms concept

3.2.2.1 RSA

The Ron Rivest, Adi Shamir, and Len Adleman system, named RSA (A. S. Ronald L. Rivest 1978) by the initial letter of each creator's name, is an asymmetric key encryption algorithm that uses a pair of keys. That pair contains a public key that is used for encryption and a private key for decryption. The setup of this algorithm is the multiplication of two very large prime numbers and the publication of their product public which will be part of the public key. The origin of that product remains hidden and is used as the secret key. So, the basic idea is that the factors of the product cannot be recovered from the product itself. Thus, the concept of security in the RSA algorithm depends on the tremendous difficulty of factoring.

3.2.2.2 ElGamal

Elliptic Curve Variant (also called ElGamal) (ElGamal 1985) is a cryptosystem based on the Discrete Logarithm Problem. The philosophy of the algorithm comes from the assumption that given a number the discrete logarithms are really hard to find a specific time frame, whilst the opposite operation can be computed efficiently. The size of the secure key size in most cases is greater than 1024 bits, but also 2048 bits can be used. The processing speed of ElGamal is quite slow, so it is used mainly for key authentication protocols. Elliptic Curve cryptography variants of ElGamal are becoming increasingly popular and are drawing the attention of researchers due to their efficiency.

3.2.2.3 Elliptic Curve Cryptography (ECC)

Elliptic Curve Cryptography (ECC) (Víctor Gayoso Martínez n.d.) describes the cryptographic tools and protocols whose security is based on special versions of the discrete logarithm problem. What that means is that they don't use numbers modulo, but they are based on sets of numbers that are associated with mathematical objects called elliptic curves. When it comes to key-size and level of security against known attacks, ECC 160-bit key is the same as an RSA or Digital Signature Algorithm 1024 bit. ECC requires fully exponential time to solve problems as a result it demands less processing power, storage space and bandwidth. Thus, ECC is an attractive solution to devices with constrained computing resources like cellular phones or smart cards. Last but not least, it is worth mentioning that ECC includes several cryptographic schemes that were initially designed for modular numbers such as ElGamal encryption and Digital Signature Algorithm.

The integrated PrivacyNet component will provide anonymization functionalities, which will use encryption methodologies to protect sensitive data from its disclosure. Security policies defined for datasets shared and processed by the ShareNet component will specify the type of encryption mechanism used to anonymize sensitive information described via the specific attributes.

3.3 Attribute-Based Encryption

Attribute-Based Encryption (ABE) is a public-key encryption scheme which encrypts sensitive information based on a set of predefined policies for access control. The decryption of such data can take place only from specific and distinct authorized entities who possess the appropriate user attributes. Any ABE scheme found in the literature nowadays falls into one of the following two primary types:

- Ciphertext Policy ABE integrates the access policy within the ciphertext itself, allowing thus the data owner to define who can decrypt the data (Bethencourt, Sahai and Waters 2007).
- Key Policy ABE integrates the access policy within a user's private key, enabling thus the one who generated the key with the capability to define who can decrypt the data (Goyal, Pandey, et al. 2006).

The next sections of this chapter describe the most influential works towards ABE through the passage of years, as well as the latest advantages and contributions in each primary type. However, it is worth noticing that despite the considerable achievements in this area, the vast majority of ABE schemes still lacks on practicability (Naehrig, Lauter and Vaikuntanathan 2011, Pang, Yang and Jiang 2014), making their adoption apparent on specific only industries like the cloud storage solutions and smart grid services. The issues that should be solved before adopting

an ABE scheme have been analysed in (Liu and Wong, Practical Attribute-Based Encryption: Traitor Tracing, Revocation and Large Universe 2016, Liu, Jiang, et al. 2018), where the authors of these papers list all identified issues, propose potential solutions for them, and ultimately present their own practical schemes.

3.3.1 Ciphertext Policy ABE (CP-ABE)

Most ABE schemes aim at providing a fine-grained access control solution where a user is identified by a finite number of attributes, which in turn denotes his/her decryption capabilities over the encrypted ciphertexts of the underlying cryptosystem. (Brucker, Petritsch and Weber 2010) presented a novel approach for the end-to-end exchange of encrypted information by integrating a break-glass feature into an ABE technique. Doing so, they were able to control the overriding of access restrictions in several types of dynamic environments. Another approach tailored for application in cloud computing environments was presented in (Wang, Liu and Wu 2010), where a CP-ABE scheme was combined with a hierarchically identity-based encryption system. In their work, they also proceeded to both proxy and lazy re-encryptions in order to boost the performance factor required in several enterprises, critical infrastructures (CIs), and providers. A few years later, (Xu and Martin 2012, Li, Shi and Zhang, Searchable ciphertext-policy attribute-based encryption with revocation in cloud storage 2017) implemented dynamic user revocation and key refreshing models that matches with the properties met in a typical ABE scheme, taking also into account the schemes' constructions and limitations provided by (Waters, Ciphertext-Policy Attribute-Based Encryption: An Expressive, Efficient, and Provably Secure Realization 2011). The latter presented an expressive and efficient realization of the CP-ABE in the standard model that was provably secure under the decisional bilinear Diffie-Hellman problem[16]. The same problem has been also used as the proof-of-security assumption in other novel schemes such as (Ostrovsky, Sahai and Waters 2007), where the authors presented a set of techniques which could be applied on CP-ABE schemes and realize non-monotonic access abilities. Last but not least, (Goyal, Jain, et al. 2008) introduced a scheme supporting advanced structures in the form of a bounded size access tree, where each node of the tree bears a varying threshold in order to enhance the expressibility capabilities of their system.

Other works focused into implementing multi-authority ABE schemes as a mean to skip identity-based solutions for the monitoring of users' attributes and secret keys' distribution. (Chase, Multi-authority Attribute Based Encryption 2007) presented a multi-authority scheme that was secure under any number of corrupted authorities, a work which was later improved in terms of privacy and security by replacing the initially presented Central Authority (CA) with an anonymous key issuing protocol (Chase and Chow, Improving Privacy and Security in Multi-Authority Attribute-Based Encryption 2009). Following the same path, (Muller, Katzenbeisser and Eckert 2009) constructed a CP-ABE compatible scheme capable of supporting numerous independent parties for the maintenance of attributes and keys, while (Božović, et al. 2012) identified the potential requirement of guarding the sensitive information from the CA, and implemented a "honest-but-curious" entity which was on purpose unable to arbitrary decrypt system's ciphertexts. Nowadays, many CP-ABE schemes can be found on smart grid infrastructures, aiming at contributing into the optimal and reliable operation of these CIs. (Hur 2013, Hu, et al. 2017) addressed the challenge of hiding from system operators not only the information being exchanged over the network, but the included access policies as well. Doing so, the desirable data privacy and the

[16] https://en.wikipedia.org/wiki/Decisional_Diffie%E2%80%93Hellman_assumption

policy privacy of the underlying organization are being achieved. Finally, a more recent work (Sethi, Pradhan and Bera 2020) has adopted the same concepts, extended the application area based on the current state of smart grid architectures, and developed a practical CP-ABE scheme with obfuscated policies and outsourcing decryption capabilities.

3.3.2 Key Policy ABE (KP-ABE)

In contrast to the CP-ABE cryptosystems, (Goyal, Pandey, et al. 2006) managed to implement an attribute-richer cryptosystem for the fine-grained sharing of encrypted information. Their study coined the term KP-ABE for the first time in the literature, while at the same time, they were also able to provide a delegation mechanism which subsumed the features met in a classic hierarchical ID-based cryptosystem (Gentry and Silverberg, Hierarchical ID-Based Cryptography 2002). Another prominent work in this area is the KP-ABE scheme constructed by (Ostrovsky, Sahai and Waters 2007), which was able to handle even non-monotone Boolean access structures for the representation of private keys involving the AND, OR, NOT and threshold operations. (Lewko, Sahai and Waters 2010) also adopted such non-monotonic access formulas, but they also contributed with various other enhancements including the reduction of ciphertext's size overhead and the generation of constant keys' size. The result of their work was the development of an efficient and secure public key broadcasting encryption system. About a year later, (Attrapadung, Libert and de Panafieu, Expressive Key-Policy Attribute-Based Encryption with Constant-Size Ciphertexts 2011) presented the first KP-ABE which was able to support negated attributes with a constant ciphertext size. Doing so, they were able to narrow down the number of pairing evaluations to a constant number, by embedding a quadratic sized number of attributes within the scheme's private keys.

Even though the expressivity is a required characteristic in many industries, there are also cases where an arbitrary high number of attributes with a smallest possible ciphertext size overcomes the before mentioned criterion. (Wang and Luo 2013) concerned about the linearly growth of ciphertexts' size in relation to the number of attributes associated with them and proposed a new KP-ABE construction with constant ciphertext length, secured under the selective-set model of the general Diffie-Hellman exponent problem. However, the main drawback of the KP-ABE regardless of the proposed construction, lies to the inability of the encryptors to explicitly state the users who are able to decrypt their ciphertexts. Even though that KP-ABE's encryptors are able to choose an arbitrary number of descriptive attributes during the encryption stage, according to (Kumar J. and Aluvalu 2015) the aforementioned limitation has discouraged the extensive usage of this type of schemes in various application areas and enterprises. Relying on the restrictions met in both CP-ABE and KP-ABE schemes, (Attrapadung and Imai, Dual-Policy Attribute Based Encryption 2009) presented a novel Dual-Policy ABE scheme based on the combination of (Goyal, Pandey, et al. 2006) and (Waters, Ciphertext-Policy Attribute-Based Encryption: An Expressive, Efficient, and Provably Secure Realization 2011) constructions, supporting thus two distinct but interoperating access control mechanisms upon the encrypted dataset. A few years later, their work was revisited in order to provide a fully secure ABE construction for the dual predicates (Attrapadung and Yamada, Duality in ABE: Converting Attribute Based Encryption for Dual Predicate and Dual Policy via Computational Encodings 2015), denoting that the Duality in ABE bears some open issues and requires further investigation by the research community.

3.3.3 Identity-Based Encryption

Identity-Based Encryption (IBE) is a special encryption scheme which is closely related with the application of ABE. According to (Herranz 2017), an ABE scheme can also act as an IBE scheme

since both approaches make use of similar ciphertext and computational methodologies. Their main difference lies to the fact that IBE takes advantage of exclusively one attribute, namely the ID, while the ABE is able to take into consideration multiple attributes. Back in 2001, a computationally and bandwidth cheap IBE scheme based on quadratic residues was presented in (Cocks 2001), but as the author himself noted, the cryptosystem was vulnerable to an adaptive chosen ciphertext attack and special care should be taken to block this kind of attacks. The same year another identity-based encryption scheme (Boneh and Franklin, Identity-Based Encryption from the Weil Pairing 2001) was able to address this issue by implementing a Weil pairing[17] methodology and relying cryptosystem's security on a natural variant of the computational Diffie-Hellman problem (Joux 2000). It is also worth noticing that at this study several potential applications of IBE were identified, like the revocation of public keys, delegation of decryption keys and delegation of duties for specific participants. Those application areas were later extended with the introduction of Hierarchical Identity-Based Encryption (HIBE) schemes (Horwitz and Lynn 2002, Boneh, Boyen and Goh, Hierarchical Identity Based Encryption with Constant Size Ciphertext 2005), which were resistant against domain collusion and supported a constant size ciphertext, respectively.

Nevertheless, the first IBE scheme with error-tolerant and collusion-secure capabilities was presented in (Sahai and Waters 2005), acting as the commencement for other research works to propose new IBE techniques for the efficient encryption of the underlying information. Once more, (Waters, Efficient Identity-Based Encryption Without Random Oracles 2005) presented a practical and secure IBE scheme without random oracles, where the security of the cryptosystem was based on the decisional Bilinear Diffie-Hellman problem. On the other hand, (Zhang, Wu and Hu 2012) focused into providing a performance-wise solution regarding the trade-off between private-key and ciphertext sizes in HIBE schemes, while (Z. Wang 2017) implemented an identity-based aggregation protocol to prevent both unauthorized access to smart grid resources and unintentional or malicious human-made errors. Last but not least, IBE and signatures schemes are also met at the fog layer of Supervision Control And Data Acquisition (SCADA) IoT CIs (Baker, et al. 2020), as a medium to guarantee the privacy of sensitive information and allow the multilevel user access upon the underlying system.

3.4 Homomorphic Encryption

Nowadays, the rapid growth of security risks across all kinds of CIs has denoted the need of adopting advanced data protection and privacy-enabled solutions. Among the existing and proposed methodologies, Homomorphic Encryption (HE) keeps gaining ground thanks to its controversial nature compared with the traditional encryption algorithms like the AES, RSA, DES, RC5, etc. The main feature of HE lies to its ability of conducting calculations on the encrypted data without having to decrypt them in the first place. Any produced output of such operations is also derived in encrypted form and is accessible only to the owner of a right cryptographic key, satisfying thus the privacy-preserving condition which is met in several organizations and industries. The very first reference of such a technique could be possibly attributed to (Rivest, Adleman and Dertouzos 1978), but the rational application of it was introduced 30 years later by

[17] https://en.wikipedia.org/wiki/Weil_pairing

(Gentry, A Fully Homomorphic Encryption Scheme 2009). Several different HE techniques were presented throughout the years but were attached to the needs of specific domains of interest. Today, the most commonly used techniques can be categorized into the following three HE schemes based on the number of operations allowed upon the encrypted input (Acar, et al. 2018):

i. Partially Homomorphic Encryption (PHE) allows only one type of operation on the encrypted data, but this type of operation can be applied unlimited number of times.
ii. Somewhat Homomorphic Encryption (SWHE) allows only specific types of operations on the encrypted data, where each one can be also applied a finite only number of times.
iii. Full Homomorphic Encryption (FHE) not only allows unlimited types of operations on the encrypted data, but those operations can be also applied unlimited number of times.

The upcoming sections of this chapter present the progress of HE methodologies, setting as starting point the initially presented PHE schemes. Afterwards, the most crucial SWHE works which were able to dramatically change the HE landscape and give shape to the first FHE schemes are described. Finally, the most prominent studies of the latter scheme that could possibly find application in the context of CyberSANE are introduced.

3.4.1 Partially Homomorphic Encryption

The rapid adoption of cloud-based solutions across many CIIs made several individuals, teams, and institutes, to focus into the encryption of data on edge computing environments using techniques which are classified among one of the aforementioned HE schemes. (Shoukry, et al. 2016) proposed a multi-party privacy-preserving PHE scheme where data were encrypted on the side of the client before their transmission and storage on a cloud environment, while the cloud provider applied a gradient descent algorithm which was unable to completely reveal the encrypted information. Following a similar pattern, (Alexandru, et al. 2020) took advantage of a PHE scheme and secure multi-party computation techniques to develop a cloud-based protocol which could efficiently deal with the quadratic optimization problem of distributed private data. Last but not least, (Murthy and Kavitha 2019) presented one more promising PHE methodology capable of operating on encrypted data prior their upload to a cloud infrastructure. The authors of this paper were able to improve both the performance and the time needed to process the underlying information, while at the same time, information security and privacy were also enhanced as the encryption key was not known to the cloud provider.

On the other hand, (He, Pun and Kuo 2012) deployed a PHE algorithm backed by authentication and digital signature mechanisms, in order to provide a secure and efficient cryptosystem for the exchange of information in smart grids. According to (McDaniel and McLaughlin 2009), smart grids are a special type of CIs which bear several security and privacy challenges that should be addressed. (Gao, et al. 2018) focused into the privacy-preservation issue met in the current cyber-physical systems due to the vast amount of information exchanged in such CIIs. This issue has been turned into a Big Data problem and the authors proposed a generic Privacy-Preserving Auction Scheme based on a PHE implementation to enhance information privacy and secure the network protocol design, a revamped HE technique which was successfully applied in the past to combinatorial auctions (Yokoo and Suzuki 2002, Pan, Zhu and Fang 2012). Other widely accepted PHE schemes with various application areas, involve the deterministic encryption scheme presented by (Paillier 1999), as well as the order-preserving encryption scheme for numeric data proposed by (Agrawal, et al. 2004). The former gave birth to two homomorphic probabilistic encryption schemes in order to deal with the Composite Residuosity Class Problem,

while the latter allows the construction of database indexes over encrypted tables making fruitless the malicious intrusion of a third-party upon a system's database.

3.4.2 Somewhat Homomorphic Encryption

Opposed to the aforementioned PHE schemes and prior to 2009 where the first achievable FHE scheme was introduced by (Gentry, A Fully Homomorphic Encryption Scheme 2009), the majority of HE schemes were able to conduct either addition or multiplication operations over the ciphertexts. Such HE schemes were therefore attributed with the term of SWHE. Polly Cracker scheme (Fellows and Koblitz 1994) is one of the first implementations of this kind which was widely adopted by the research community and gave birth to additional variants (Levy-dit-Vehel and Perret 2004, Van Ly 2006, Albrecht, et al. 2011). However, all of the presented solutions were deemed either too expensive or insecure to be adopted (Steinwandt 2010). Even before that, it was evident that the evaluation of operations upon the encrypted data should be feasible across different sets. (Sander, Young and Yung 1999) proposed the usage of poly-many AND ciphertexts with a single only OR/NOT gate, a SWHE scheme which was later extended by (Ishai and Paskin 2007) to support an arbitrary number of evaluations on the encrypted data of branching program circuits. In the meanwhile, (Boneh, Goh and Nissim, Evaluating 2-DNF Formulas on Ciphertexts 2005) succeeded in providing a revolutionary scheme which was able to carry an unlimited number of additions with one multiplication. The novelty of their work lies to the fact that ciphertext's size remained constant and did not grow exponentially, compared with the traditionally SWHE schemes used at that time. A few years later, (Gentry, Halevi and Vaikuntanathan, A Simple BGN-Type Cryptosystem from LWE 2010) presented a slightly modified BGN cryptosystem where its security was based on the hardness of Learning With Errors (LWE) problem, and it was also capable of supporting a larger message space from its predecessor.

3.4.3 Full Homomorphic Encryption

A relatively recent state-of-the-art and comparison study on the existing FHE schemes (Ahmed and Elkettani 2016) showed that they may indeed prevail in security and privacy terms, but the majority of them have to either improve their runtimes, decrease the size of the produced keys and ciphertexts or change the underlying framework which is responsible for the implementation of their main encryption scheme. The most prominent FHE schemes take advantage of techniques that reduce the noise which is added to plaintext during encryption, or make use of noise-free refreshment techniques. (Xiao, Bastani and Yen 2012, Li and Wang, Noise-free Symmetric Fully Homomorphic Encryption based on noncommutative rings 2015, Y. Wang 2016) proposed symmetric encryption schemes based on a set of homomorphic properties which are derived through matrix-based operations. However, those approaches have proved to be insecure under specific types of attacks (Gjøsteen and Strand 2016). An optimized version of the initially presented FHE schemes was once more proposed by (Gentry, Sahai and Waters, Homomorphic Encryption from Learning with Errors: Conceptually-Simpler, Asymptotically-Faster, Attribute-Based 2013), while (Zhang, et al. 2014) implemented a Ring Learning With Errors (RLWE) FHE scheme which took into account re-linearization techniques and the Brakerski FHE scheme (Brakerski and Vaikuntanathan 2014) to reduce ciphertext's length, noise level, and decryption complexity.

Over the last few years, computational-faster and more secure FHE schemes, suitable for application across a variety of domains were presented. A quite fast FHE scheme based on ring variants of the GSW cryptosystem was proposed by (Chillotti, et al. 2020). The authors of this

paper achieved to improve the running time of HE operations by reducing the bootstrapping key size, while the bootstrapping circuit was also modified to support the conversion of LWE ciphertexts into their low-noise RingGSW counterparts. In contrast to the aforementioned noise-based technique, (Mustafa, et al. 2020) presented a noise-free FHE approach based on non-associative algebra properties. Their solution made use of a novel compression methodology in the dimensional vectors used for the encryption, which could serve as a potential security basis for post-quantum cryptosystems too. Despite the aforementioned works, there are also publicly available third-party libraries which support FHE computations on encrypted data. Microsoft SEAL[18] is such an open-source and easy-to-use research project, which however has been reported to come with security flaws and information leakage under specific use cases (Peng 2019). This reinforces the fact that even if a secure FHE scheme is selected, the implemented application or protocol could easily fail in terms of security.

3.5 Format Preserving Encryption

Format-Preserving Encryption (FPE) (T. R. Mihir Bellare 2009), (P. Rogaway 2010), is a form of deterministic cryptography specifying that one can encrypt data in a manner that the output can maintain the basic properties of the input; the same representation format within a finite lexical set and length (Ben Morris 2009), (M. B. Rogaway 1999). This necessity emerged from the need of operating and storing confidential/critical data with a given length, regardless of their encryption state, mainly in databases of economic, healthcare or military cloud infrastructure (J. Z. Li 2012), (Richard Agbeyibor 2014). Encrypting words of a given alphabet, a TCP/IP payload (Adrián Pérez-Resa, Using a chaotic cipher to encrypt Ethernet traffic 2018) or a credit card number are some of the common usages of FPE (Hoover 2015.), (Zheli Liu 2010), (Mor Weiss n.d.). Taking the latter as a lead, each credit card number consists of a six or eight-digit range called Issuer Identification Number (IIN), then a Major Industry Identifier (MII) digit and the rest are formed accordingly to each individual user account. So, in a bank network, in order to verify a credit card number and forward a payment request, the IIN stays intact and the rest of the target credit card number is encrypted with FPE, as shown on Figure 4.

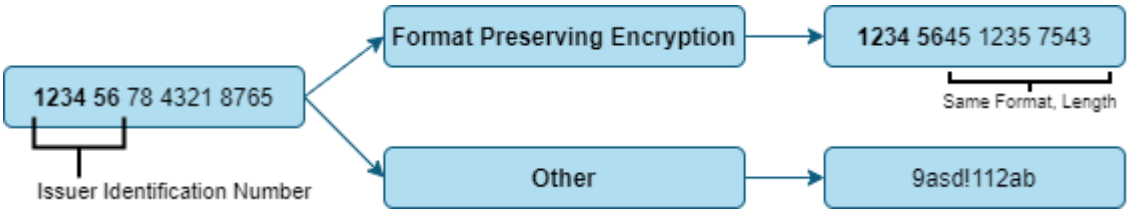

Figure 4. Format Preserving Encryption on Credit Card Numbers

In 2002, three methods (John Black 2002) were proposed for ciphers with arbitrary finite messages: a prefix method, cycle-walking and a Feistel construction. The first two methods operate on small-space messages, though the third method encrypts a greater variety of data. In more recent years, The National Institute of Standards and Technology (NIST) published a standard (Dworkin 2016) consisting of two mechanisms for FPE called FF1 or FFX [Radix] and FF3 (P. R. Mihir Bellare, The FFX mode of operation for format-preserving encryption 2010), short

[18] https://www.microsoft.com/en-us/research/project/microsoft-seal/

for Format-preserving, Feistel-based encryption modes (Eric Brier 2010), (P. R. Mihir Bellare, The FFX mode of operation for format-preserving encryption 2010). FF1 and FF3-1 divide plain text into let's say smaller components and after 10 rounds of using the Encryption Standard Encryption (AES) function FK with some concatenations and mod radix calculations the string is encrypted preserving its initial properties. In 2016, Bellare et.al proposed a message-recovery attack against FF3 on small messages (V. T. Mihir Bellare 2016). Their attack consisted of 3 sub-attacks. The LHR attack, recovers the left half of the message when the right half is known. The RHR attack is the opposite of the first but more logically sophisticated. The FMR attack recovers the entire target message, which is the combination of the two. A year later, Betul Durak et.al. also, broke FF3. Their contributions were a total-break attack (i.e. the attacker obtains the secret key) and a new known-plaintext attack on 4-round Feistel networks, but also provided a prevention technique to mitigate this attack (F. Betül Durak 2017).

FPE also addresses IoT environments. From an older work encoding common Base64 data (Steven R. Hart 2018) to more complex but real-time examples such as encrypting the Global Positioning System (GPS) information encoded within an image. A common user does not have the knowledge that while uploading an image online, may also provide geolocation metadata, thus such a project enhardens the privacy of each user (Changhyun Lee 2019). Others showed that FPE could be useful in encrypting the traffic exchanged between a car equipped with an electronic control unit (ECU) and its sensors (Insu Oh 2019). This could prevent possible denial-of-service or replay attacks which could actually put in risk the control of the car of even the life of a targeted driver. Another research created a smart image partial encryption method using FPE. This mechanism consisted of two basic steps, first select an area of pixels in a video or image to encrypt; an area that contains sensitive data that need to be transformed to preserve anonymity or privacy (e.g. a face in a video or a car license plate) and then encrypt it with FF1 and FF3-1. They ported that mechanism in small embedded devices that do not have the hardware capabilities of executing such an intensive task (Wonyoung Jang, Partial image encryption using format-preserving encryption in image processing systems for Internet of things environment 2020). Also, an alternate research suggested a modified version FF1 called FF1+, designed precisely for IoT devices, that outperformed the vanilla version due to the dynamic round and key-size selection. Their testbench was a common 1.2 GHz RaspberryPi 3 in order to prove that this algorithm is suited for this kind of environments (Alessandro Baccarini 2019). Since no other stream ciphers algorithms existed in the literature implementing FPE encryption, Adrian Perez-Reza et.al. created CTR-MOD, a stream-cipher version of FPE for high throughput flows. The encryption takes place in one of the sublayers of physical layer, where 8b/10b symbols exist (i.e. 8-bit words to 10-bit symbols). This encoding adds some properties to the data stream such as transition density or DC balance etc. They tested it on a FPGA (Field Programmable Gate Array) encrypting raw traffic data streams and their solution reached a better encryption rate from both FF1 and FF3 (Adrián Pérez-Resa, A new method for format preserving encryption in high-data rate communications 2020). Other approaches recommended to exchange the internal block cipher AES of FF1 and FF3 with LEA (Deukjo Hong 2013) and SPECK (Ray Beaulieu 2015), two lightweight block ciphers (Wonyoung Jang, A format-preserving encryption FF1, FF3-1 using lightweight block ciphers LEA and, SPECK 2020). LEA is a fast encrypting algorithm with a small code size. It consists only of ARX operations (modular Addition, bitwise rotation and bitwise XOR) and is secure-proof from previous key recovery attacks (Khovratovich 2009), (Andrey Bogdanov 2011). LEA with a 128-bit key for one-block encryptions was faster in comparison to AES. SPECK is implemented in an elastic way so that it can be easily executed on low-end devices. It was first written in 2011 and published after two years of cryptanalysis. SPECK use simple round functions in contrast to the more complex of AES. It was shown that this algorithm has the highest throughput on 64- bit processors of any other block cipher. In CyberSANE, it is essential to employ format preserving encryption in order to maintain information that will allow us to extract knowledge from a specific security event. For instance, we will need to correlate the IP addresses

of the attackers and/or the attack targets in order to better identify an ongoing attack in the network or identify an APT. FPE will also allow to provide better attack descriptions and provide more accurate IOCs over ShareNet. The algorithms that are currently used and can be exploited in CyberSANE are the IoT-specific FFX1+ and the lightweight SPECK.

4 Blockchain Technologies

Over the last years, the blockchain technology has raised significant interest and has been adopted by several organizations as an advanced security solution to enable trusted transactions between untrusted participants. Blockchain is a Distributed Ledger Technology (DLT) which makes use of a database to record transactions of value using a resilient cryptographic signature. This public or private ledger comes in the form of continuously interconnected blocks, timestamped, and secured in order to prevent any tampering or malicious revision attempts. (Salman, et al. 2018) highlighted the inefficiencies met in most centralized architectures and denoted the advantages of blockchain-based methodologies in authentication, confidentiality, privacy, access control, data provenance and integrity assurance areas. Therefore, this chapter aims to identify the latest studies which utilize blockchain approaches in the context of cyber-security domain.

4.1 Building Blocks

Any blockchain-based framework can be seen as a digital network consisted of an interconnected chain of computers, which follow a specific set of rules defined within a telecommunication protocol. Most of these blockchain solutions are organized as decentralized peer to peer (P2P) networks, where there is not a single central authority to govern the system, but instead all network clients are connected to one or more peers to share resources. The first P2P networks were developed exclusively for file sharing purposes (e.g. Napster[19] and BitTorrent[20]), but the resilient and scalable nature of this infrastructure gave birth to additional P2P applications. Nowadays, such blockchain applications are either public, private, or permissioned, and can be found on most domains of major importance including file storage (Wang and Zhang 2018), asset management (Verma, et al. 2017), insurance (Raikwar, et al. 2018), medical (Azaria, et al. 2016), and finance (Treleaven, Brown and Yang 2017) services.

All the information being exchanged on a blockchain network has to be recorded for validation purposes, therefore a ledger is deployed as a mean to track and verify such transactions between peers (Deshpande, et al. 2017). This distributed ledger has to be maintained and reconciled by each peer of the network, before proceeding with a new resource sharing from his side. There are also occasions where the ledger is not necessarily distributed, but it is instead governed by a single central authority. Doing so, the aforementioned reconciliation task does not fall into each network participant, but the generic concept of organizing data into append-only blocks, remains the same as before.

Additionally, blockchains take advantage of several security mechanisms to secure and prevent the malicious or unintentional access to third-party peers and threat-actors (Moubarak, Filiol and Chamoun 2018). Most of blockchain solutions adopt a public-key cryptography scheme to generate a peer's keys (which are essentially his network addresses), a hashing algorithm to

[19] https://en.wikipedia.org/wiki/Napster
[20] https://en.wikipedia.org/wiki/BitTorrent

protect and detect potential tampering or revision attempts, and a set of digital signatures methodologies to efficiently authenticate and verify the integrity of each generated block.

Despite the affiliated network, ledger and security mechanisms, the most critical building block of any blockchain is definitely its consensus mechanism. Consensus is in fact an agreement between blockchain' parties, which describes the methodology followed to reach unanimity during a group decision making process (Bach, Mihaljevic and Zagar 2018). There are also cases where consensus is applied from a single only individual based on a set of predefined but immutable rules. Over the years, many consensus algorithms and protocols have been presented (Nguyen and Kim 2018) to either surpass the restrictions met in various industries, or serve as an alternative technological solution towards the satisfaction of the requirements addressed from them.

Byzantine Fault Tolerance (BFT): This consensus lies its foundations on the PoW but makes use of an entirely different principle. The computation power of each peer in this occasion is used to evaluate a block's validity, considering that the maximum number of "faulty" peers (malicious peers from the perspective of cyber-security) should not be greater than -or equal to- the 1/3 of all blockchain peers (Castro and Liskov 1999). It is evident that as the blockchain network grows and additional peers are added, then the security is also enhanced thanks to this admission.

Delegated Byzantine Fault Tolerance (DBFT): It is a consensus which combines the features met in DPoS and BFT (NEO Team 2014). It enables the voting of a responsible delegate who is going to create the next block in the chain, but his validity must be verified by at least 2/3 of the remaining peers.

Direct Acyclic Graph (DAG): This is a special type of consensus which takes advantage of a directed graph data structure to imitate the effect of sidechains instead of a single chain of blocks (Pervez, et al. 2018). Doing so, it allows the simultaneously execution of multiple transactions on a set of different sidechains.

Proof of Authority (PoA): It is a reputation-based consensus mechanism which supports a quite higher number of transactions per second compared with the traditionally used consensus (Barinov and Baranov 2018). Such a thing is feasible by assigning the verification of blocks to trusted peers known as validators.

Proof of Elapsed Time (PoET): This is a fair-lottery consensus algorithm developed by Intel Corporation[21], where any peer of the system is equally like to mine and win. This type of consensus has found a wide acceptance among permissioned blockchain networks and is one of the supported consensus in Hyperledger Sawtooth (Dhillon, Metcalf and Hooper, The Hyperledger Project 2017).

Proof of Identity (PoI): It is a consensus mechanism focused on the authorization of the identity of a peer based on his private key (Azouvi, Al-Bassam and Meiklejohn 2017). Each block of the chain can only be related with an identified participant, providing in this way a secure, reliable, and trusted environment of transactions for smart systems.

Proof of Luck (PoL): This is another provably fair consensus algorithm (Milutinovic, et al. 2016) where each peer receives -and is represented by- a lucky number and contributes with its own

[21] https://www.intel.com/content/www/us/en/homepage.html

block in the chain. However, only the chain with the highest value of lucky numbers is chosen and the rest are rejected.

Proof of Stake (PoS): It is a consensus algorithm developed specially to overcome the limitations met in the PoW consensus (Bitcoin Wiki 2019). It offers a low-energy and lightweight mechanism to verify the blocks of a blockchain solution by allocating the ledger's update to the peer who holds the most "stakes". A more recent variation of the PoS consensus is the Delegated Proof of Stake (DPoS) consensus (Bitcoin Wiki 2020), where each peer has the right to "vote" and ultimately "elect" a set of peers responsible for the further maintenance of the ledger.

Proof of Work (PoW): This is the first consensus mechanism ever existed and it was initially introduced in Bitcoin's whitepaper (Nakamoto 2019) back in 2008. It still remains the most widely used consensus, despite the fact that block verification is a process which requires a significant amount of energy, resources, and time. Its consensus is based on peers' competition to solve a complex mathematical problem (that is however easily verified once it is solved from the rest of peers).

It is worth noticing that there are additional consensus mechanisms which have not been included in the list above. However, their development and practicability have been focused into specific only areas (e.g. cryptocurrencies, government efficiency and voting services, financial avenues, etc.) which are out-of-scope of this report. Typical samples of such consensus are the Proof of Activity, Proof of Capacity, Proof of Importance, Leased Proof of Stake, Proof of Burn, Proof of History, Proof of Importance and Proof of Space, where according to (Nguyen, et al. 2019) PoS variations keep gaining ground compared to the rest of consensus protocols.

Last but not least, incentives have been found to be another building block which is thoroughly taken into consideration during the implementation of a blockchain framework (Huang, et al. 2019). Incentives are a set of methodologies embedded within the blockchain itself, aiming to influence and encourage the participation of peers into the system. Even though that this layer of blockchain has not yet find practicability on domains relative with CyberSANE's CIs (He, et al. 2018), it is evident that the development and integration of novel incentive mechanisms could definitely contribute to the wider adoption of blockchain frameworks in the near future.

4.2 Smart Contracts

Smart contracts are executable pieces of code stored in the blockchain itself, which make use of specific protocol rules to facilitate, verify, and enforce a contract between two parties (Franco 2014). (Szabo 1994) seems to be the first one who coined the concept of "smart contract" as we know it today for the sake of law, economics, and physical objects prone to contractual conditions. Over the previous two decades, this concept found quite limited applicability in both the research and business sectors, but the advent of blockchain solutions over the last few years gave shape and an actual practicability to smart contracts. Smart contracts are built on top of a blockchain system and, depending on the application area, can be configured to self-execute and self-enforce their predefine contract without any human interaction. However, smart contracts always incorporate the business logic followed by the underlying blockchain, and their output is embedded along with the rest of transactions in a subsequent block of the chain. Hyperledger Fabric (Cachin 2016) is the most well-known and robust framework for the development of permissioned blockchains which require the support of smart contracts. Its versatile architecture not only enables the distributed application programming of smart contracts in domain-specific languages, but also allows the integration of industry-standard identity management solutions in order to deal with resource-related and performance-related cyber-attacks (Androulaki, et al. 2018). Following the same pattern, (Mendi, et al. 2019) presented their own smart contract application framework which inherited all the core features met in a blockchain implementation,

and allowed the authoring of automatically enforced electronical agreements. It is worth noticing that this framework was deployed and tested in several real-world use cases by HAVELSAN[22], a Turkish company dealing with simulation, ICT and cyber-security applications.

Apart from the aforementioned blockchain frameworks, smart contracts have been also used to define access control policies and authorization solutions (Ouaddah, Abou Elkalam and Ouahman 2017) in IoT environments, as well as to ensure data integrity between two participants for the needs of producer and consumer agreement (Liu, et al. 2017). Another recent study (Unal, Hammoudeh and Kiraz 2020) also aimed at specifying and verifying the policies required in smart contracts for the wide-scale adoption in the upcoming 5G network technology, but this is an area that still requires more investigation. Even though we have not discovered any smart contract approaches used in today's CIIs, their potential adoption could definitely change the current landscape on information exchange, once they are able to overcome a few identified vulnerability aspects and additional general-purpose languages are introduced (Singh, et al. 2020).

4.3 Supply Chain Solutions & Enterprise Applications

The increasingly digitization and automation of the energy sector emerged new cyber-security challenges which had to be addressed in order to efficiently protect the corresponding CIs (Andoni, Robu, et al., Blockchain technology in the energy sector: A systematic review of challenges and opportunities 2019). Such CIs are nowadays consisting of power grids that combine several Industrial Control Systems (ICS) in the form of hardware or software components, as well as a set of network services associated with energy operations. All these components and services ultimately compose an energy supply chain which must be monitored and secured against malicious actors. (Mylrea and Gourisetti 2018) presented a blockchain-based supply chain security solution for the detection, protection and response to anomalies and cyber-threats that take place in CIs related with the energy domain. Their solution took advantage of a cryptographically signed distributed ledger capable of providing the necessary data provenance, attribution and auditability capabilities issued by (FERC 2016). Their permissioned PoA blockchain architecture ensured the data integrity throughput of the chain of custody by verifying both the sender's and signer's identities, while data privacy was preserved by adopting a Merkle tree with root hashes as the system's consensus algorithm and verification mechanism.

The robustness, efficiency and security requirements met in the modern energy CIs also concerned (Liang, et al. 2019), where their main objective was to detect and prevent cyber-threats that could lead to a false system manipulation from false data injection attacks (Liu, Ning and Reiter 2011, Yang, et al. 2014). For that reason, they proposed a data protection framework which attributes self-defensive capabilities to any power system by exploiting core distributed blockchain technology features. Any cyber-attack was blocked thanks to actor's inability to manipulate the data packets across the majority of network channels, as well as gaining access to sufficient infrastructure meters. The performance analysis of the presented approach showed improved data encryption and verification capabilities, but on the other hand, it also displayed limited practicability in SCADA environments mainly due to the fact that changing meters is an unrealistic costly operation for real-time high-availability systems. Another novel framework named DeepCoin (Ferrag and Maglaras 2019) addressed the aspect of safely exchanging energy

[22] https://www.havelsan.com.tr/en

between a vendor and a buyer within a smart grid network, by harnessing both deep learning and blockchain technologies. The presented framework incorporated a reliable high-throughput P2P mechanism based on the BFT, where ledger's blocks were generated using short signatures schemes and hash functions in order to prevent smart grid attacks and preserve infrastructure's privacy (Boneh and Boyen, Short Signatures Without Random Oracles and the SDH Assumption in Bilinear Groups 2008). Their approach demonstrated high-accuracy in terms of security and privacy compared with the traditional machine learning methodologies (e.g. Support Vector Machine, Random Forest, and Naïve Bayes), but the applicability of framework should be further evaluated against today's edge computing smart grids (Xiong, et al. 2018, Mukherjee, et al. 2017). Last but not least, (Andoni, Robu, et al., Blockchain technology in the energy sector: A systematic review of challenges and opportunities 2019, Lu, et al. 2019) investigated the current blockchain trends, opportunities, challenges and risks in energy sector, and oil and gas industries, respectively. Both of them concluded that the development of such blockchain solutions is still at its early stage, where both hybrid architectures and hybrid consensus mechanisms have to be developed for the necessities met in those energy CIs' landscape.

Most of the aforementioned permissioned blockchains make use of either a PoA or BFT consensus algorithm. PoA can be seen as a variant of the classic BFT consensus, focusing into providing improved performance. However, (De Angelis, et al. 2018) noticed that in real-world scenarios where data integrity is of high-importance, PoA is inferior compared to the practical BFT consensus. An innovative and promising framework for the development of various types of blockchains tailored to the needs of today's enterprises and CIs is the Hyperledger Project (Dhillon, Metcalf and Hooper, The Hyperledger Project 2017). Hyperledger consists of several independent components which are gradually unified into a single codebase, where among them Sawtooth (Olson, et al. 2018) is the most prominent one used for the creation of PoET permissioned blockchain solutions. (Staroletov and Galkin 2019) presented a methodology for the formal verification of the correctness of the adopted PoET consensus, as well as a container virtualization solution based on Docker[23] for the rapid development, integration and testing of the underlying blockchain systems. Since PoET is running within the trusted execution environment of Intel Software Guard Extensions (SGX), a higher level of integrity is guaranteed compared to the traditionally used approaches. Furthermore, (Chen, et al. 2017) proceeded with a security analysis of the PoET mechanism, and proposed a series of mitigate methodologies if the enterprise environment is compromised by a third-party. Last but not least, (Milutinovic, et al. 2016) introduced PoL, a similar to PoET consensus which also supports the SGX instructions' set. Many variations of this algorithm have found practicability in CIs composed of cyber-physical blockchain systems like IoT devices and cooperative vehicular systems (Machado and Fröhlich 2018, Bettín-Díaz, Rojas and Mejía-Moncayo 2018, Boos and Lacoste 2020). The latter area is expected to meet exponential growth over the next years with the introduction of additional CIs related with the Intelligent Transportation Systems (ITS), where advanced blockchain solutions have been already presented (Lei, et al. 2017) to allow the dynamic and secure distribution and management of transport keys.

The first practical application of blockchain was introduced by (Nakamoto 2019) and involved the provision of a P2P electronic cash system. Since then, several financial institutions and banks have set-up or adopted a blockchain approach to efficiently deal with the restrictions met in the current state of international payments like their tracking, their transaction and exchange fees,

[23] https://www.docker.com/

and their processing times. It is evident that such financial services are the backbone of a country's economy, therefore, the uninterrupted, fast, and secure functionality of their CIs is deemed necessary. The two major candidates in this domain are the Ripple[24] and Stellar[25] foundations, and each one employs a different set of techniques and consensus. Ripple is a centralized governance model where the access to the network is permissioned and the privacy of the data remains within the codebase of the CI, while Stellar follows a decentralized governance model where the access is permissionless and the data are stored in a public ledger. The decentralized architecture of Stellar may has enabled its adoption from the masses, but Ripple tends to be the most widely accepted solution for integrated supply ledgers dealing with economics (Armknecht, Karame and Mandal 2015, Swan 2018, Lohmer, Bugert and Lasch 2020).

4.4 Blockchain-as-a-Service

The improved performance and scalability features offered in today's cloud solutions, led several CIs to host a part of their services in cloud-based environments. (MacDermott, et al. 2015) conducted a research regarding the existing protection methodologies and weaknesses met in such systems, denoting the necessity to assure the confidentiality, integrity, and availability of the information being exchanged. Even though the availability of data on-demand is considered a de-facto feature of a cloud infrastructure, in most cases such data have to be encrypted in order to avoid a potential compromise of sensitive information from a third-party. Except for the HE methodologies described in a previous chapter of this report, there was no other practical way to perform computations over encrypted data. However, the introduction of blockchain as well as the latest advances on security methodologies, gave birth to alternative solutions for the manipulation of data in their encrypted form. (Zyskind, Nathan and Pentland 2015) presented a P2P network where all peers jointly store and exchange data with cryptographic privacy guarantees, like the recording of time-stamped events and the hashing of files. Moreover, an optimized version of the multi-party computation algorithm introduced in (Baum, Damgård and Orlandi 2014) was responsible for the assignment of tasks across several peers, enabling thus only the partial access and manipulation of encrypted data.

On the other hand, (Benet 2014) developed a P2P distributed hypermedia protocol known as InterPlanetary File System (IPFS), aiming at decentralizing the information exchanged in versioning control systems by "mutating" them into a DAG data structure. A few years later, their work was adopted by (Vashikar, et al. 2020) in order to create a decentralized cloud storage solution backed by the PoW Ethereum[26] blockchain, without relying on cloud service providers for data storage and sharing operations. Following a similar concept, (Vijayakumar, et al. 2019) presented a client-server architecture framework which deployed the Hyperledger Fabric blockchain to store and retrieve medical healthcare records from a distributed ledger, utilizing tamper-proof data, transactions' transparency and an enhanced peers' trust. The adoption of CI services in private environments, as well as their dependability and provisioning over the cloud-based ones has also concerned (Melo, Dantas and Oliveira, et al. 2018, Melo, Dantas and Maciel, et al. 2019). A first, the authors of (Melo, Dantas and Oliveira, et al. 2018) evaluated the dependability of a blockchain-as-a-service environment, presenting a modelling methodology

[24] https://ripple.com/xrp/
[25] https://www.stellar.org/foundation
[26] https://ethereum.org/en/

based on the Dynamic Reliability Block Diagrams and the deployment of Hyperledger Cello[27] for the creation and management of blockchain systems. About a year later, (Melo, Dantas and Maciel, et al. 2019) proceeded additionally to the evaluation of four different infrastructures which hosted blockchains and found out that the hyper-converged blockchain architecture surpassed its competitors in terms of availability, cost, and security. Last but not least, hybrid blockchain-as-a-service solutions have been also reported and proposed in (Şafak, Furkan and Erol 2019), where blockchain ensures the accuracy, security and integrity of the information being exchanged, and a central database is used to efficiently counter with the productivity and management issues that periodically occur in such systems.

A recent study (Abhishta, van Rijswijk-Deij and Nieuwenhui 2019) regarding the impact of the Distributed Denial of Service (DDoS) cyber-attacks that took place back in 2016 against the managed Domain Name System (DNS) providers NS1 and Dyn, showed the catastrophic outcomes of a successful large scale attack of this kind. DNS is after all a critical Internet infrastructure which needs to be secured against cyber-threats in order to provide its expected range of Internet-based services. (Wei-hong, et al. 2017) analysed the first promising blockchain-based DNS services, namely the Namecoin[28] and Blockstack (Ali, et al. 2016), denoting their built-in durability against such DDoS attacks. IPFS (Benet 2014) is another blockchain which could be used as an alternative DNS solution, while (Aranjo, Adivarekar and Hegde 2019, Wang, Hu and Liu 2019) have presented their own blockchain-based approaches to deal with DDoS, DNS spoofing and DNS amplification attacks. Finally, it is worth noticing that DDoS protection along with PKI-based identity and blockchain-based DNS security mechanisms, have been extensively described in (Gupta 2018), covering a large set of both the existing and the emerging global cyber-threats.

4.5 Blockchain over Wireless Communication

Blockchain has shown a great potential for establishing trust and consensus mechanisms thanks to its distributed feature and the fact that no involvement of any third party is required. Its applications in wireless networks have thus attracted many researchers. Nevertheless, wireless communications are prone to different attacks than wired networks [CAO2020,AHM2018], which reduces the reliability of a traditional blockchain. However, blockchain and different distributed ledgers techniques could be used in a different meaning to ensure that the wireless communications will benefit from enough network resources such as bandwidth, retransmission number, reliability, etc.

To do so, it is important to understand the relationship between wireless communication features and blockchain as well as the performance constraints, especially in terms of reliability and security. Posing on the counterparts can facilitate designing a dedicated blockchain-enabled wireless communications for critical information infrastructures.

In the CyberSANE context, we will analyse and model a new blockchain based routing protocol tailored for wireless communications. We will inspire from TORCOIN [GHO2014] and BALADIN

[27] https://www.hyperledger.org/use/cello
[28] https://www.namecoin.org/

[MES2019] that use blockchain mechanisms to secure bandwidth. However, this could not be directly applicable in CyberSANE CII since they rely on the existence of an operator who acts as a trust reference. In CyberSANE, we will investigate how to create such a secure trust reference depending of the use case and for the cases where this is not a viable solution, we will investigate a fully distributed trust reference.

For this latter case, we will combine cryptography techniques, machine learning techniques and block chain mechanisms. All such techniques will require a slight adaptation since none of them could be directly embedded in hardware constraints devices such as the wearable. Also, encryption generally enlarges the size data to be send over a wireless link, which is not recommended for link reliability, bandwidth and energy consumption. Therefore, we will adapt know techniques such as key exchanges, Bayesian approaches, Thompson sampling and other to make them efficient for the different use cases of CyberSANE and ensure the secure techniques will meet every requirements in terms of security and reliability without jeopardizing the well-functioning of wireless devices.

5 Modelling Language

Cyber threats are rapidly evolving, increasing the number of security incidents, especially for CII. Depending on their severity of impact, security incidents may have devastating effects on the overall continuity of the company. Therefore, it is increasingly recognized that incident handling is a key component of critical infrastructure protection that presents a unique set of activities and clear measures to contain and reduce the impact of a cyber incidents. The process for incident handling involves coordinated and organised approach to identify and consistently analyse cyber incidents – a process that requires diligence and which could be complicated because of the dynamicity and complex nature of CII. The planning and initialisation of incident handling processes require an early definition, awareness, and representation of security and privacy requirements of cyber assets in relation to how assets can be exploited in malicious ways and the measures that can be taken to reduce the impact of incidents. This can be supported by a modelling language that offers a standard way to specify and visually analyse the key elements of incident handling process such as the analysis of vulnerabilities, threats, and risks. The inclusion of a modelling language will also provide the basis for describing security and privacy-related requirements, as well as facilitate a broader understanding of several desirable properties such as technical and procedural actions.

The intended purpose of this chapter is to introduce a graphical modelling language that consists of conceptual constructs and a corresponding process for specifying and expressing cyber incident handling. The main benefit is to enable domain and non-domain experts to effectively elicit and model security and privacy concerns in the early phases of incident handling, including the analysis of cyber incident contexts such as threats, vulnerable assets, their associated security risks, and risk treatments. The capturing, analysis graphical visualisation of these key elements of incident handling will enable developers to model and reason about security and privacy requirements. Hence, the modelling language will support developers to gain better understanding of and perform security incident analysis, identify relevant cyber-security threat/attack patterns within critical infrastructure, as well as reason about conflicts and trade-offs between cyber incident handling requirements and security, privacy, and forensic requirements.

This section reports on the outcome of Task 7.2. In particular, the main contribution to the state-of-the-art is the development of a new Cyber Incident Handling Modelling Language that specifically focuses on the handling of modelling incidents in the context of CIIs. In its approach, the work is novel because it consolidates concepts from different fields, such as security requirements, forensic, threat intelligence, critical infrastructure, and cyber incident handling. The approach allows modelling the phases of incident handling lifecycle from three different views (critical information infrastructure, threat and risk analysis, and incident response). Results based on the context of the case study demonstrated that this new modelling language is a viable solution for handling cyber incidents. The work breaks down the various models that allow visual representation and correlation between threats, risks, incidents and control.

5.1 Objectives

The chapter aims to provide a description of a novel modelling language and accompany process for cyber incident handling that will enable the identification and modelling of incident handling activities. To achieve the objective, a language is developed which combines concepts from requirements engineering and security and privacy engineering with incident handling theory and forensics for modelling and graphical visualisation of incident handling processes.

It will enhance the identification and modelling of the heterogeneous interconnections of a CII and the dependencies with other assets, including stakeholders and operators, and their goals within different domains of CIIs such as healthcare, transport, and energy sectors. It will also provide visualisation capabilities threats, vulnerabilities, and risks analysis, including the modelling and analysing of incident prevention, mitigation, and response strategies for CIIs.

Specifically, the language will contribute to the deliverables of WP3 by enabling a developer to create a graphical representation of threats with respect to the potential consequences, intentions, and characteristics of a malicious actor. In this line, ENISA (ENISA 2016) has developed a version of Threat Taxonomy that intends to aid collection and understanding of threats related information. CyberSANE has proposed a comprehensive and well-structured Threat Taxonomy that aims at improving the understanding of threats related to CIIs. The document is built according to ENISA's Threat taxonomy, and since they are closely related, CyberSANE's Threat Taxonomy is used for threat modelling.

5.1.1 The Approach used to Develop the Modelling Language

This section provides a detailed description of the process used for developing the cyber incident modelling language. Fundamentally, a modelling language is an integral part of software development that attempts to capture the complex behaviour of systems, people, and software agents within a distributed sociotechnical environment (Bresciani, Perini et al. 2004). A modelling language enhances the analysis of systems behaviour by allowing developers to concisely express solutions to well-defined problems and helps to avoid problems like misunderstandings between people and lack of interoperability between tools (Krahn, Rumpe et al. 2008). It also helps in identifying and connecting conceptual ideas to ensure consistency, efficiency, and effectiveness, as well as identifying interrelationships between these components.

The development of any modelling language requires a structured definition, elicitation, and reasoning of domain-related concepts, as well as the application of well-established methodology that focuses on security and privacy aspects such as the Secure Tropos (Mouratidis and Giorgini 2007). This kind of adaptation will enable the integration of new context-specific concepts and consolidation with pre-existing ones in a comprehendible and consistent manner that satisfy the requirements for incident handling. Therefore, to follow a structured format, the cyber incident modelling language is developed according to some important phases as detailed afterwards.

5.1.1.1 Identification of Concepts

The first phase involves the identification of concepts according to the existing literature on security and privacy engineering. In particular, we focused on specific domains relevant for incident handling to define the concepts such as digital forensics, cyber resiliency, and cyber threat intelligence. The underlying motive is to define concepts that are integral for the development of the model. Some of the domains explored include:

5.1.1.1.1 Security and Privacy Engineering

Existing security and privacy engineering methodologies are leveraged for developing the modelling language. In particular, Secure Tropos, which is "*a security-aware software systems development methodology, which combines requirements engineering concepts, such as actor, goal, plan together with security engineering concepts such as threat, security constraint and security mechanism, under a unified process to support the analysis and development of secure and trustworthy software systems* (Mouratidis and Giorgini 2007)*"* are used and further extended.

The reason for choosing Secure Tropos is that it is well suited for modelling security requirements and provides an in-depth analysis of security issues from organization and social settings.

5.1.1.1.2 Digital Forensics

Digital forensics entails the process and methods towards the investigation of a cyber incident or crime. It provides procedures and techniques for the systematic identification, collection, and preservation of evidences derived from digital sources for the purpose of facilitating the analysis of cyber incidents (Flaglien 2017). The inclusion of concepts for digital forensics such as evidence, offer several benefits during the extraction and analysis of cyber incidents. For instance, ISO/IEC 27037:2012 (27037:2012 2016) provides guidelines for specific activities in the handling of digital evidence, such as the identification, collection and acquisition of potential evidence.

5.1.1.1.3 Cyber Resiliency

Cyber resiliency "Cyber resiliency is the ability to anticipate, withstand, recover from, and adapt to adverse conditions, stresses, attacks, or compromises on cyber resources" (Bodeau, Graubart et al. 2011). The concepts of cyber resiliency present a set of design principles that are used to inform activities and processes for managing cyber incidents. MITRE (Barnum 2012) provides a working knowledge of cyber resiliency concepts, such as cyber course of actions, that are essential for aligning relevant cyber resiliency design principles with other concepts from systems and security engineering disciplines for ensuring incident handling.

5.1.1.1.4 Cyber Threat Intelligence

Cyber threat intelligence provides evidence-based knowledge representation of threat information that an organization uses to understand threat landscape and potential threats faced by the organization. This information is then used to identify, prepare, and prevent cyber threats. In this regard, STIX model "*provides a language for the specification, capture, characterisation and communication of standardised cyber threat information* (Barnum 2012)*"*. STIX model is important because it provides a unifying architecture that consists of concepts such as indicators, incidents, adversary tactics, techniques, procedures, etc. which are useful in developing the proposed modelling language.

5.1.2 Development of Conceptual Model and a Process

Based on the output of concepts identified in the previous phase, a conceptual model will be defined in this second phase of the method, which deals with describing physical or sociotechnical aspects of a system in an abstract form (Embley and Thalheim 2012). It provides the foundation for the specification and representation of abstract ideas, as well as capturing the features of a computer program using a collection of relevant concepts. The main reason of adopting a conceptual model lies to its ability of providing a high-level understanding and representation of the concepts for the different developers and users of the CyberSANE system, remaining expressive enough to handle the varying levels of complexity. The conceptual model for the language is developed using UML class diagram, which uses a graphical notation to construct and visualize object-oriented systems by representing system's classes, their attributes, operations and the relationships among objects (Fuentes-Fernández and Vallecillo-Moreno 2004).

Each concept is presented as a class with a list of attributes, and the concepts are related to each other using relationships such as association and generalization, while a glossary is provided to elucidate the meaning of the concepts. In addition, a process is included to serve as a guide for developers while implementing conceptual model. The process consists of activities and steps and it encompasses various techniques, methodologies, and industry standards for ensuring validity, comprehensibility, and compliance to generally accepted guidelines.

5.2 Proposed Concepts for the Modelling Language

This section presents in detail the three steps outlined in the previous section (Section 5.1). Hence, the review of literature from different domains mentioned above has led to the identification and extraction of concepts that will be used to develop the language. The concepts identified include:

- **Critical Information Infrastructure (CII):** implies communication networks, information-based facilities, cyber-physical assets or systems that support the operations of critical infrastructure, which if damaged, would have a serious impact on the expected functionality of critical public, government or industry services. CII can also be considered as those systems that provide resources or services on which essential functions depend upon, of which possible incapacitation or destruction would result in a significant effect on the economy, security and or health of the society as a whole.
- **Actor:** represents an entity that intentions, goals, and objectives within a system who participates in a process, performs a task, or carries out an action within an organisational environment. Actor is categorised according to type (such as a developer) and the role performed (such as system development and administration)
- **Asset:** are cyber resources that can be used by multiple actors to critical functions such as systems, software, data, network devices, or other components that enable information-related activities, management, service delivery. Assets are characterized by varying attributes such as categorisation and criticality. Asset can be categorised according to network, software, or data. An asset's criticality expresses its importance or the degree to which the asset is relied upon for the delivery of critical functions.
- **Goal:** represents a state of affair or strategic interest that an actor aims to achieve or the needs of an asset. Goals are mainly introduced to achieve possible security constraints that are imposed to an actor or that exist within a CII. Goal consists of attributes as type and purpose, for example, authentication and authorisation controls could be the goal of an asset whose purpose is to ensure security protection.
- **Constraint:** a set of restrictions related to security and privacy, which that must be satisfied for a specific asset or actor goal to be achieved. It consists of 'type' attribute that distinguishes security and privacy constraints.
- **Malicious Actor:** represents an individual, group or organisation that participates in hostile actions or operates with malicious intents to cause harmful effects at a CII. It is imperative to identify and represent different types of threat actors within the language based on their distinctive characteristics and motives (such as goals, motivation, tactics, and procedure) to compromise a CII. Therefore, threat actors can be characterised by their goals, the tactics, techniques, and procedures that they use.
- **Cyber Incident:** implies any kind of security-related event that produces unanticipated consequences, unwanted occurrences or instances that could likely compromise, breach, or violate the security of assets and CIIs. Example, cyber incident can include but not limited to unauthorised disclosure of classified information, unauthorised modification of classified information, and malicious disruption, use or processing of CII's resources.

- **Impact*:*** used to determine the measurable implications or consequences caused by a security incident to assets within CIIs. The intention is to measure the potential severity of adverse effect caused by a security incident to a CII. Impact contains attributes as *description, type, affected, affected infrastructure,* and *severity.*
- **Vulnerability:** is the weakness of an asset or security mechanism that can be exploited by a threat that could result in degradation or loss (incapacity to perform its designated function).
- **Threat:** implies any cyber-event with the potential to cause unwanted effect or harm to asset because of vulnerabilities being exploited by a threat. The attributes of *threat* include *category* that describe the class of threat (such as denial of service), the *severity* of the threat with regards to its potential impact, and *affected assets* to identify the assets affected by the threat.
- **Risk:** is the potential consequences of an incident, threat or vulnerability that can result in a range of negative consequences, loss, damage, or undesirable change to assets. Risk is associated with attributes as *likelihood* that measures the possibility of a risk occurring, and *impact,* that estimates the potential losses associated with an identified risk. In addition, any risk can be further characterised as a cascading risk or residual risk.
    - **Cascading Risk:** represents a situation in which a security incident propagates within assets or component.
    - **Residual Risk:** is the amount of remaining risk that remains, after control mechanisms have been applied to control or mitigate a risk. Also, the residual risk implies the component of risk which actors are willing to accept.
- **Control Mechanisms:** represents any technical safeguards, systems, or processes that are used to safeguard assets, manage risk, control threats, security incidents and mitigate vulnerabilities. The concept is characterised by attributes according to *type, goals,* and *measure of effectiveness* to either remove, counter, or mitigate risks or cyber-incidents. Control mechanisms are associated with three distinct types:
    - **Detective Mechanisms**: include security control measures implemented to detect and send alert about impending threats or incidents.
    - **Preventive Mechanisms**: are designed to stop a security incident, a threat or risk from occurring, as well as reduce or avoid the likelihood and potential impact to CII.
    - **Corrective Mechanisms**: include control measures that are taken to address existing damages or restore CII to their prior state following a security incident.
- **Evidence:** represents electronic data about observable patterns, artefacts or behaviours from different sources that can be used to analyse a security incident. Evidence is generated from various sources such as log files, error messages, intrusion detection systems, firewalls. For example, evidence of an incident may be captured in several logs that each contains different types of data. It contains attributes such as *type* to indicate the evidence type, and *source,* to indicate where evidence is extracted from such as intrusion detection system logs.
- **Cyber Course of Action:** the set of security controls employed by actors in response to cyber incidents. It is characterised by *procedural* and *technical* courses of action that are applied within an operational setting in response to the impact of cyber-incident. In contrast to *Control Mechanism,* Cyber Course of Action is intended to integrate a combination of technologies and administrative procedures to recover from and adapt to adverse security incidents, risks and impacts on those CIIs that have not been sufficiently prevented by Control Mechanism.

5.3 Conceptual Model and Implementation Process

In this section, the conceptual model of the language is presented, which provides an interpretation of -and highlights- the relationship between the concepts for expressing cyber incident handling process (as shown in Figure 5). Concepts in the metamodel are represented as boxes, attributes as properties inside the boxes, and the relationship between the concepts is created using arrowed lines.

According to the metamodel, a CI provides vital functions and operations within a specific sector such as health and energy, whose disruption could result in serious disruption on economic, wellbeing, security, or safety of the society. A CI is ordinarily operated, managed, controlled, and used by a collection of *actors* who have different types of *goals* that they aim to achieve. Also, each actor goal is attached to a security and privacy *constraints* that must be satisfied for any specific goal to be achieved. In addition, each CI consists of and requires a wide range of cyber assets to achieve its goals and deliver critical functions. Hence, CI is appraised to recognize the underlying domain of operations and its boundary in terms of the critical functions being provided, the actors whose interest and goals must be considered, the specific security and privacy constraints imposed on actor goals, as well the supporting cyber assets.

Assets have varying levels of criticality and are usually associated with various forms of *vulnerabilities.* Vulnerabilities can be introduced by misconfigurations or inadequate access control and authentication in an asset that subjects to exploitation by a *malicious actor.* A malicious actor possesses different kinds of skills and *goals* for compromising a cyber asset or critical functions, such as breaching data protection mechanisms for individual or third-party's financial gains. The activities of a malicious actor to exploit vulnerabilities could result in a *threat.* A threat entails different characteristic and is categorised according to type and severity. Also, the manifestation of a threat could result in a *risk* such as the interruption of critical functions that combinedly leads to a *cyber incident* and subsequently causes a variation of *impact* to one or more assets. Fundamentally, a prioritised set of *control mechanisms* in the form of policies, procedures or technically safeguards are typically implemented to address vulnerabilities and threats, prevent risks, and ultimately mitigate the impact of cyber incidents against the CI. These control mechanisms are implemented according to *detective, preventive and corrective* mechanisms, to serve various purposes like detecting threats, averting the potential impact of a threat, and to restore cyber assets to prior state, respectively. However, control mechanisms such as firewalls, Intrusion Detection Systems (IDS) and Intrusion Prevention Systems (IPS) are not always effective enough to serve their intended purposes. They are usually compromised or subverted by evolving and sophisticated Advanced Persistent Threats (APTs), *cascading risks* that spread beyond the instance of a cyber asset, and even more exacerbated by *residual risks* thereby resulting thus in one or more *cyber incident.* In addition, *evidence* is generated and collected by security mechanisms containing information about threat patterns and cyber incident. The evidence collected can then be aggregated and analysed for detecting patterns and trends, as well as responding to cyber incidents.

The occurrence of a cyber incident triggers the process for incident handling, which has the goal of mitigating the potential impact of a cyber incident and risk, as well as eradicating the root cause of a cyber incident. *Cyber course of action* expresses the necessary measures to address and respond to impending incident by means of *procedural course of action* and *technical course of action*, and is initialised by an actor such as incident response team. The cyber course of action also improves the existing control mechanism and the overall secure posture of a CI.

A distinctive facet of the proposed modelling language that is worthy of mention is that it embraces the notion of decomposing the conceptual model into three main artefacts: *critical infrastructure analysis*, *threat analysis*, and *incident response*. The goal of this decomposition is to devise a graphical view of the different phases of incident handling that can be easily understood by all stakeholders and which is expressive enough to translate the any perceivable level of complexity in the conceptual model. It will also facilitate the ability of developers to completely utilize and

implement the concepts and their relationships. Therefore, Table 5, Table 6 and Table 7 provide a summary of the different views, the motivations behind creating the views, concepts that can be utilized to generate the views, and perceived outcome of the views.

MODEL 1: Analysis Of CII				
Motive: The basis of this model is to create a view of Critical Information Infrastructure (CII) with regards to the boundary of a critical infrastructure. The model will enhance developer's awareness and understanding of the connection between CII and assets, critical functions being provided, and consideration of the human elements that influence the operations of CII.				
Key Elements (Concepts in the Model)				
Critical infrastructure	***Asset***	***Goal***	***Constraint***	***Actor***
The inclusion of this concept provides an understanding and identification of critical infrastructure and associated functions. The goal is to ensure that the CII used to provision, manage or control critical services are modelled according to the significance they have on predetermined services or operations.	Identifies all the ICT systems that are essential for the operation of CII and can be modelled according to the criticality level or support to CII.	The goal of actors are included in the modelling in order to analyse and reason about privacy and security requirements from CII point of view, as well as actors, strategic interest from organizational, operational and technical perspective.	Indicates the security and privacy constraints that are imposed, and which must be met towards the satisfaction of security and privacy goals.	The model will aim to identify and map the different views of actors involved or have strategic interest in CII from organisational, technical and operational viewpoint (such as owners, users, operators, and regulators).
Perceived Result: the main result is to provide an awareness of the CII from organisational context and for identifying and assessing potential vulnerabilities, threats, and risks that could lead to a cyber-incident, as well as incident response activities.				

Table 5: CII Analysis View

MODEL 2: Threat Analysis					
Motive: Provides a general analysis and representation of potential threats, vulnerabilities and risks that could lead to a cyber-incident, including the analysis of potential impact. The model will allow the description and decomposition of cyber-incidents according to existing threat taxonomy, facilitate the measurement of cyber-incident scenarios, and characterize the behaviour of a malicious actor that.					
Key Elements (Concepts in the Model)					
Vulnerabilities	***Threat***	***Risk***	***Critical information infrastructure***	***ThreatActor***	***ControlMechanism***
The underlying and emerging vulnerabilities associated with CII are included in the model to	Provides a clear articulation and granular, decomposition and characteristics of the most significant threats, covering	Potential Risks are identified through modelling the threat scenarios within the context of	The model includes the CII whose vulnerabilities are targeted and exploited by a	Captures the different threat actor types that could potentially compromise	The existing control mechanisms that perform certain functionalities such as to remove, identify, or mitigate cyber-incident are modelled.

highlight those elements of CII that can be circumvented or exploited by a ThreatActor, which could lead to a threat.	the relationship with various ThreatActor and ControlMechanism.	relevant vulnerabilities, as well as the representation of CascadingRisk and ResidualRisk.	MaliciousActor, and the CII that are affected by the impact of a CyberIncident.	CII, including characteristics such as the commonly used tactics, techniques and procedures.	Hence, ControlMechanism is modelled according to CorrectiveMechanism (the measures taken to restore a damage caused by a CyberIncident), PreventiveMechanism (measures taken to eradicate an incident), and DetectiveMechanism (measures to identify an incident is occurring). In addition, its inclusion in the model assists in determining the degree of effectiveness controls which translates into identifying these gaps are technological, process or organisational-dimensions.

Table 6: Threat Analysis View

MODEL 3: Incident Response					
Motive: aims to capture incident response strategies that can be used to identify cyber-incidents, contain and minimize the impact, and recover from cyber-incidents. It will enhance the understanding of relevant response strategies that are suited to an organization to effectively and efficiently contain or mitigate the impact of potential threats, vulnerabilities, risks and cyber-incidents in particular					
Key Elements (Concepts in the Model)					
CyberCourseOfAction (CCoA)	***Critical information infrastructure***	***Impact***	***Cascading/Residual Risk***	***Actor***	***ControlMechanism***
Identifies a combination of operational processes and technologies that provide the ability to respond, protect and recover from cyber-incidents. CCoA consists of such strategies as Procedural and Technical CCoA. Procedural CCoA models cyber-incident handling strategies by human elements (including security	The CCoA are comprehensively mapped to each CII in order to highlight and correlate CCoA strategies (procedural or technical strategy) are most suitable or applicable to the security and privacy contexts of a CII as far as handling	The efficiency and scope of CCoA strategies are included in the model to highlight the extent to which the specific impacts of a	The model will provide a view of the cascading and residual impacts that are mitigated or reduced or prevented by the CCoA strategies.	Similarly, actors (such as incident response team) are included in this model to identify the role each actor plays in the direction, implementation and achievement of the different	The existing control mechanisms that perform certain functionalities such as to remove, identify, or mitigate cyber-incident are modelled.

awareness and management oversight), policies and plan, and regulatory compliance. Technical CCoA are those actions that enable the orchestration and automation of incident response mechanisms for ensuring that the desired security and privacy posture of CII are maintained during an incident. Technical CCoA are categorised according to key elements as protection actions, recovery actions and recovery actions.	incident is concerned.	cyber-incident that can be mitigated from security and privacy.		CCoA strategies.	
Perceived Result: Specification of the relevant incident handling strategies that are applicable to a given context of cyber-incident within CII, including the actors involved in the initialisation and maintenance of incident handling process.					

Table 7: Incidents Response View

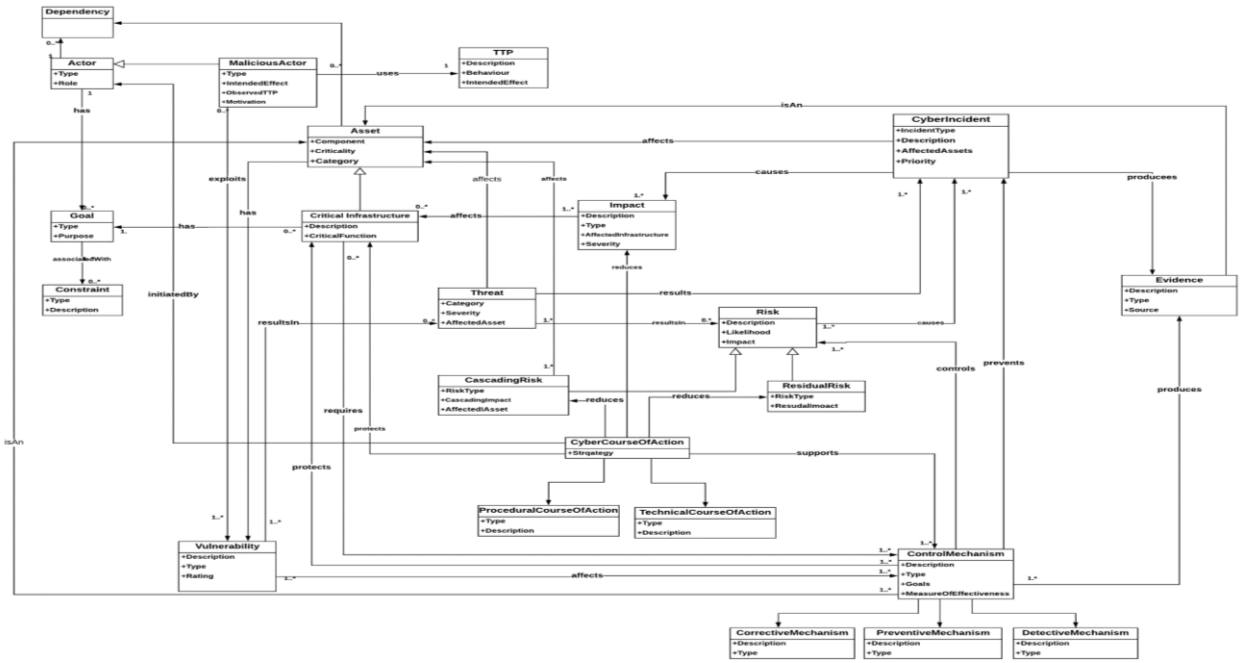

Figure 5: Conceptual Model for Cyber Incident Response Modelling

5.4 Method for the Modelling Language

In the previous section, a conceptual model for the proposed language was presented and, in this section, the underlying process for the modelling language is presented. A process establishes a strong relationship between multiple steps for effective delivery of expected outcome. An activity

deals with interdependently linked tasks that receive and convert one or more input into an output artefact. Essentially, the process aims to introduce a prerequisite guidance in different set of activities and steps that developers can follow to analyse, specifying, and graphically modelling incident handling processes. The process will also strengthen developers' understanding of predefined incident handling procedures that can be tailored to specific context of a CI. Based on the three modelling views of the proposed model, the process proposes 3 main activities and steps to support detailed analysis and extraction of deliverables from the concepts. Each activity specifies the steps that need to be followed, and each step identifies the needful inputs, and final output. Primarily, the output of each activity serves as the input to the next activity that follows it.

Activity One (Analysis of CII) deals with defining the context of CI to support a comprehensive understanding of critical functions, asset, goals and security and privacy constraints. Activity Two analysis deals with the carefully orchestrated process of identifying and assessing vulnerabilities, threats, and risks. The last activity focuses on defining an incident response process in terms of managing, containing, and minimizing the impact of cyber incidents, including recovery actions. It is worth noting that the activities and steps of the process are formed according to various industry best practices, frameworks, guidelines, and standards such as ISO 27000 (Irwin 2019), ENISA guidelines (Mattioli and Levy-Bencheton 2014), NIST(NIST 2012), and OWASP (OWASP 2014). Table 8 shows a summary of the different activities and steps involved in the process, including the input and output for each step, as well as the techniques used. In addition, a matrix is provided at the end of each activity to support developers/security analysts to record the output of each step.

Activity	Steps	Input	Technique	Output
Activity 1: Analysis of CII	Identify Critical Sector and Functions	Enumerated list of critical sectors and critical functions provided by EPCIP (EPCIP 2008) and ENISA (Mattioli and Levy-Bencheton 2014)	Analysis and mapping of organisational operating environment according to EPCIP and ENISA guidelines	Speicfiication of critiical sector and criticall functions relevant to an organisation.
	Identify Actors, Goals, and Security and Privacy Constraints	Stakeholder profile whose interests and needs affect or are affected by critical information infrastructure, including the security and privacy restrictions that must be met to fulfil these needs	Consideration and analysis of diverse stakeholder attributes such performed functions, hierarchical levels, roles, influence and interest analysis	A list of actors, actor goals and security and privacy constraints are conditions that must be satisifed to meet actor goals.
	Determine Assets and Criticality	Existing asset profile	Use of a Fuzzy Asset Criticality System, as well as employing asset categorisation and criticality ranking using impact value and weight score proposed by NIST, ENISA and FIPs (Wunder, Halbardier et al. 2011), (Federal Information Processing Standards Publication 2004, Mattioli and Levy-Bencheton 2014)	Categorisation of assets and determination of asset critical level.
Activity 2:	Identify and Analyse Threats	CyberSANE's Threat Taxonomy (see D3.1)	Application of OCTAVE, DREAD and STIRDE for vulnerability and threats analysis (Meier 2003, Caralli,	A comprehensive profile detailing potential threats to assets

Threat Analysis			Stevens et al. 2007, Microsoft 2007)	categorisation and severity of threats
	Identify Vulnerabilities	Vulnerability databases such as NVD and CVE. (MITRE 2008, Booth, Rike et al. 2013)	Determination of vulnerability severity rating and prioritisation using Common Vulnerability Scoring System (NVD 2019)	Identification, and rating of vulnerabilities according to severity of impact
	Identify Risks	Perceivable lists of risks to critical assets and functions	Determination of impact and likelihood of risk according to NISTs guidance for conducting risk assessment (NIST 2012), OWASP Risk Rating Methodology (OWASP 2014). Identification residual risk and identification of cascading risk.	A detailed risk register highlighting risks, impact, likelihood, and rating, as well as residual and cascaded risks.
	Identify techniques for cyber incident detection and analysis	Cyber-attacks detection and using LiveNet component of CyberSANE system	Prioritisation and impact rating of cyber incidents according to NIST incident prioritisation matrix and impact rating (NIST 2012)	A detailed list of cyber incidents, incident priority, impact rating, affected assets
Activity 3: Identify	Define Incident Containment, Eradication and Recovery Actions	Cyber incident profile		

Table 8: Activities and Steps of the Process

5.4.1 Analysis of CII

Organisations that provide critical functions are usually distinctive in nature and deliver services within a defined scope. The identification of operational context that influences an organisation's services and functions is key to successful incident response process. Along this line, the analysis of CII is concerned with identifying and modelling a CI from organisational and operational context, for the purpose of establishing a clear awareness of the current factors that may influence an organisation. The goal is to represent the sector of CI, functions and assets that are used to manage, control, and support the provisioning of criterial services, as well as the actors that operate and those that are served by the CII. The concepts that support the creation of modelling view in this activity include *Critical information infrastructure, Asset, Goal, Constraint* and Goal.

Therefore, an actor such as a developer or security analyst with significant familiarity and knowledge of an organisation's operational context could initiate this activity according to critical service-dependent approach proposed by ENISA (Mattioli and Levy-Bencheton 2014). The approach begins with the identification of critical sectors, followed by the identification of critical functions, and then the critical information assets that are used to support these critical functions. In principle, CI is identified by looking at critical functions being provided, based on the presumption that a disruption could result in disastrous impact on the vital functions of an organisation or the society. This approach provides a comprehensive way of identifying, understanding, and analysing the complex and operational characteristics of CII.

In practice, the approach consists of three steps namely identification of critical sectors, identification of critical services and identification of CII supporting critical services. Identification

of critical services as the most critical step consists of two different techniques namely state-driven and operator-driven. In the state-driven approach, the process for identifying CII is guided by governmental agencies that have the "mandate to identify and protect CII" and it is more relevant for scenarios where governmental agencies are involved in the process of identifying CII from generic context. On the other hand, operator-driven approach is more specific whereby the leading role of identifying a CII is assigned to the operators or asset owners of CIIs within an organisation. It is more context-specific and more suited for supporting stakeholders within an organisation who are knowledgeable about their infrastructure and the critical sector within which an organisation operates.

Therefore, the developer may consider adopting the operator-driven approach in this activity, owing the fact that actors such as owners or operators of CII are more involved in the process. Hence, the operator-driven approach is fine-tuned and formulated according to three essential steps those compromises of the identification of: critical sector and functions, actors and goals, and assets and components.

5.4.1.1 Step 1 – Identify Critical Sector and Functions

CI extends across many sectors such as healthcare, transport, energy, etc. An effective identification of critical sector that applies to an organisation's operational setting and the critical functions being provided are fundamental key points for the analysis of CII activity. This implies the understanding of the critical sector under which the organisation operates to clear the path for performing subsequent activities.

A viable technique that can be used to identify critical sector and functions is to explore strategic and operational objectives of the organisation, which will support the understanding of pertinent critical sector to the organisation. This can be supported by following the guidance provided by the European Programme for Critical information infrastructure Protection framework (EPCIP) Council Directive 2008/114EC for the Identification of Critical Sectors in Europe (EPCIP 2008). EPCIP guidance identified a total of 10 sectors that are defined based on various impact assessments and studies carried out by relevant stakeholders. A diverse range of critical functions are provided to relate these critical sectors, and the critical functions they support. In addition, ENISA (Mattioli and Levy-Bencheton 2014) has provided an indictive list of critical sectors, associated sub-sectors and services that could be consulted by developers to identify critical sectors depending on their specific characteristics as shown in Table 9. This classification provides a clear channel that guides the modelling of critical sectors and functions.

Similarly, an alternative way to identify critical functions is to consider which functions will result in significant adverse impact on the organisation such as loss or destruction or interruption to function or data. These categories can be further expanded with respect to the requirements of the critical sector, and the organisation's goals and objectives.

In summary, this step enables the representation of a critical sector pertinent to an organisation based on CII concept in the conceptual model. An organisation that provides critical functions is represented as a CII within a defined boundary. Essentially, the output of this step provides an overview and understanding of CII, and the critical functions whose interruption could lead to serious damages or consequences.

Sector	Subsector	Functions
Energy	Electricity	Generation

		Transmission/distribution
		Electricity market
	Petroleum	Extraction
		Refinement
		Transport
		Storage
	Natural Gas	Extraction
		Transport/distribution
		Storage
Information, Communication and Technologies, ICT	Information technologies	Web services
		Datacentre/cloud services
		Software-as-a-Service
	Communications	Voice/data communication
		Internet connectivity
Water	Drinking water	Water storage
		Water distribution
		Water quality assurance
	Waste water	Wastewater collection & treatment
Transport	Aviation	Air navigation services
		Airport operation
	Road transport	Bus/tram services
		Maintenance of road network
	Train transport	Management of public railway
		Railway transport services
	Maritime transport	Monitoring and management of shipping traffic
		Ice-breaking operations
Industry	Chemical/nuclear industry	Storage and disposal of hazardous materials
		Safety of high-risk industrial units

Food		Agriculture/food production
		Food supply
		Food distribution
		Food quality/safety
Health		Emergency healthcare
		Hospital care (impatient and outpatient)
		Supply of pharmaceuticals, vaccines, blood, medical supplies
		Infection/epidemic control
Financial		Banking
		Payment transactions
		Stock exchange
Public Order and Safety		Maintenance of public order and safety
		Judiciary and penal systems
Civil Administration		Government functions
Space		Protection of space-based systems
Civil Protection		Emergency and rescue services
Environment		Air pollution monitoring and early warning

Table 9: ENISA'S List of Critical Sectors and Related Critical Functions

5.4.1.2 Step 2 – Identify Actors, Goals, and Security and Privacy Requirements

It is imperative to identify and analyse the various actors who perform certain roles or functions within the boundary of a CI. Creating an actor profile can be helpful for the initial characterisation of actors with regards to their roles and goals within an organisation, which can be supported by engaging people involved at various positions. Hence, this is the next step that involves the identification of actors such as internal stakeholders. Actors, goals, and constraints can be identified using the *Actor, Goal* and *Constrain* concepts, respectively. Therefore, an approach is proposed to support developers in performing this step. It consists of:

- Specifying actor according to types (such as developers, users, operators, regulators), and strategic hierarchies within the organisation (such as managers, directors, providers), etc.
- Specifying the role of actors by presenting details of the associated influence, responsibilities, and participation in critical infrastructure operations
- Associating actors with the goals they pursue such as delivery of critical functions

- Specifying security and privacy constraint. Constraints can be determined by identifying essentially relevant non-functional requirements (with emphasis on security and privacy) such as data encryption and authentication.

Therefore, the combination of these concepts will help developers to model and have a better understanding of the specific role of actors and their intentions within an organizational setting.

5.4.1.3 Step 3 – Determine Assets and Criticality

To manage the modelling activity effectively, it is crucial to identify and determine the criticality of assets (such as networks and systems) that are essential to sustain critical functions. This step involves the identification and analysis of assets that support critical functions. The step is enabled by the *Asset* concept. The aim is to support the analysis and modelling of assets according to specific category (enabled by *Category* attribute of *Asset* concept), including asset components and criticality level using *Criticality* attribute. The step will also support the analysis and modelling of threats, risks, and vulnerabilities in the next activity.

Therefore, the first task in this step is to identify and categorise assets according to a classification scheme. A developer could consider an existing cyber asset inventory that provides a detailed list of all cyber assets used by an organisation. Accordingly, assets can be categorised according to different types of identification elements such as literal identifies, relationship identifiers, synthetic identifiers, and extension identifiers. Each identification element takes into consideration different types of information. For instance, the relationship identifiers are used when assets are to be identified based on their relationship with another asset. Therefore, assets can be categorised and modelled according to categorisation as system, software, database, network, service, and data.

The next task of this step is the determination of asset criticality. Asset criticality is imperative for prioritising and developing actions that will reduce and respond to cyber incidents. A developer can determine the criticality of an asset using predefined criticality criteria based on (i) the existing criticality rating used by an organisation, or (ii) a decision support system.

5.4.1.4 Using Criticality Rating

Different impact factors can be used to determine criticality such as (a) *service impact* - the impact on loss or degradation of critical function, (b) *population affected* - the percentage of the population affected from the disruption of critical functions, (c) *economic impact* – the financial cost of service disruption (Mattioli and Levy-Bencheton 2014). CII owners always decide to make use of those criteria that comply with a finite set of necessary requirements. Therefore, using the service impact criteria, a table of indicative impact criteria is provided, which could be used as a reference to determine asset criticality in conjunction to potential levels of impact provided by FIPS 199 (Federal Information Processing Standards Publication 2004) impact rating as shown in Table 10.

Potential Impact	Definition	Impact Rating
Low	The loss or damage of an asset is expected to have a limited adverse effect that causes: (i) degradation to an extent that critical functions are provided but effectiveness of the functions is noticeably reduced; (ii) results in minor disruption to other assets (iii) results in minor financial loss	1

Medium	The loss or damage of an asset will cause (i) significant degradation of critical functions to an extent critical functions be provided but effectiveness is significantly reduced (ii) results in significant damage to other asses and components (iii) result in significant financial loss	2
High	The potential loss or damage of an asset will cause a (i) severe degradation to an extent that critical functions cannot be provided (ii) results in severe damage or loss of other assets (iii) results in major financial loss	3

Table 10: The impact on loss of services due to the failure or malfunction of an asset

5.4.1.5 Asset Criticality using Fuzzy System

Alternatively, the criticality of assets can be determined using service criteria, and by the application of an impact matrix. The impact matrix assigns ranking to the criteria based on the potential consequences of occurrence. To ensure validity and consistency, a decision support system using Fuzzy Set Theory is created.

5.4.1.6 Fuzzy Asset Criticality System

A Fuzzy Asset Criticality System (FACS) is developed, which uses Impact on loss of Critical information infrastructure (IoCA) and Impact on malfunction of Asset Dependencies (IoAD) as two fuzzy inputs for assessing Level of Criticality (LoC) for each asset. FACS contains two fuzzy events that serve as input i.e. IoCA and IoAD, and an inference engine. The inference engine contains 20 IF-THEN rules based on Mandari (Cordón 2011) and Segumo (Sugeno 1993) approaches with one crisp output after the de-fuzzification. The rules presented in a matrix and mainly are used to correlate the input values for IoCA and IoAD process.

5.4.1.7 Fuzzy Inputs and Outputs

IoAC and IoAD are used as two fuzzy inputs, assigned to five fuzzy labels respectively as shown in Table 11 and Table 12. The five labels for IoAC are "*very high (VH), high (HG) medium (MD), low (LW), and very low (VL)"*, while the labels for IoAD include " *no impact (NI), low impact (LI), medium impact (MI), significant impact (SI), and catastrophic impact (CI)* respectively. For comparison reasons corresponding scores are illustrated in the left-hand column of the tables to show how they are used in FACS.

Score	Impact on Business Process (IBP)	Fuzzy Labels
0	No impact on critical functions	NI
1	Low impact on critical functions	LI
2	Moderate impact on critical functions	MI
3	Serious impact on critical functions	SI
4	Catastrophic impact on critical functions	CI

Table 11: Fuzzy Labels for IoAD

Score	Impact on Asset Dependencies	Fuzzy Labels
0	Very High	VH
1	High	HG
2	Medium	MD
3	Low	LW
4	Very Low	VL

Table 12: Fuzzy Labels for IoAC

Crisp Score	Fuzzy Score	Level of Criticality	Fuzzy Labels
0	≤ 0.5	Low	L
1	0.5 < ≤ 1.5	Medium	M
2	1.5 < ≤ 2.5	High	H
3	2.5 <	Very High	H

Table 13: Fuzzy Labels for Levels of Criticality (LoC)

The IF-Then rules are translated in a matrix form, which uses the labels of one input in rows and the label of another input variable in columns. Cells in the matrix contain output labels that indicate the possible output resulting from a specific combination of rows and columns. Therefore, using IoAC and IoAD as inputs, LoC is generated as output, as shown in Table 14.

IoG / IBP	VH	HG	MD	LW	VL
NI	L	L	L	M	H
LI	L	L	L	M	H
MI	L	L	M	M	H
SI	L	M	M	H	H
VI	L	M	H	H	(VH)

Table 14: Matrix for Asset Criticality Classifications

The LoC for each asset (according to Very High, High, Medium, Low) is mainly obtained using minimum-maximum inference, which considers the minimum of the antecedents of the maximum

for aggregation and defuzzification. Hence, the LoC for each asset falls under one of the four categories from low to very high as defined in Table 13.

5.4.2 Activity 2 – Threat Analysis Model

This activity takes as input the analysis of CII from the previous activity. It comprises of techniques for identifying and assessing vulnerabilities, threats and risks that could result in cyber incident and potentially impact CII from an attacker's viewpoint. The activity requires a structured representation of threat information that expresses valuable situational and contextual threats that are specific to the organisation. For example, threats to assets and components are identified and modelled so that CII operators can determine the impact of a threat on targeted assets. In addition, the concepts for the incidents handling meta-model used in this activity include *Critical information infrastructure, Asset, Goal, Constraint, Actor and Impact.*

To support developers understanding, two different methods by which this activity can be approached have been identified namely: *threat classification approach* and cyber *incident operationalisation approach*. On the one hand, threat classification approach focuses on the analysis of the commonly listed threats and vulnerabilities found in threat taxonomies, classification, and information sources (such as CyberSANE Threats taxonomy) that are likely to affect the CII. This approach is broad-ranging, and it involves the identification, review, and assessment of extensive list of potential threats, and the likely impact they have on the CII. However, as threats vary over time and the techniques used by cyber criminals continue to evolve, this approach could be proved to be resource consuming and difficult to use by non-security experts.

On the other hand, cyber incident operationalisation is more specific to the assessment of peculiar threats, vulnerabilities and risks that have materialized and resulted to a cyber incident from a holistic viewpoint of threat actor. In particular, it focuses on the cyber incidents that are caused by a threat actor in order to systematically explore, characterise, and determine the strategies used to operationalise the incident scenarios, the vulnerabilities, threats, and risks, as well as the impact. However, a limitation of this approach is that it potentially overlooks a vast pool of threat information that developers can use to understand the threats that currently targeting the organisation so that pre-emptive can be taken. Both approaches are suitable depending in assisting developers to have a better understanding and assessment of a cyber incident in detail. Therefore, the following essential steps have been defined for performing this activity.

5.4.2.1 Step 2.1 – Identify and Analyse Threats

The initial step of this activity deals with the identification of potential threats, vulnerabilities, and risks. In other words, based on the assets identified in the previous task, all possible threats that could negatively impact the assets are profiled. The development or security analyst requires a sound approach that enables the gathering of valuable insights based on the analysis of situational and contextual threats that can be tailored to the organisation-specific threat landscape. Therefore, using the threat classification approach, threat taxonomies and models can be leveraged to identify potential threats that can compromise assets including exploitable vulnerabilities, which will improve the developers' ability to understand the nature of threats in a more structured manner.

Accordingly, this step is enabled by the *Threat* concept. At this juncture, the first attempt in which focuses into identifying threats is to consider information sources that provide a comprehensive list of threats. There are many sources of threat data that provide timely and applicable threat information, such as cyber threat intelligence platforms, tools and standards which can be used to obtain threat information. In this context, CyberSANE provides a comprehensive and well-

structured Threat Taxonomy that aims at improving the understanding of threats related to CII. The document is developed according to ENISA (ENISA 2016) threat taxonomy, and therefore, it can be adopted by the developer as a reliable source for threat information.

Once threat information source is identified, the next task is to methodically analyse the threats in terms of classification and severity, and create an association with the assets that are most affected by these threats. Such analysis of threats is enabled by the *Category* and *Severity* attributes. It is imperative to perform such analysis according to standard methodologies. In this vein, threat evaluation models such as OCTAVE (Caralli, Stevens et al. 2007) and STRIDE Model (Microsoft 2007) can be used in the analysis of threats. Particularly, STIRDE model is important because it categorises threats according to exploits as Spoofing Identity, Tampering, Repudiation, Information Disclosure, Denial of Service and Elevation of Privilege. In addition, Microsoft's DREAD model (Meier 2003), provides a framework for rating, comparing, and prioritising the severity of various threats by rating them on an ordinal scale. The model consists of five main categories: Damage, Reproducibility, Exploitability, Affected user, and Discoverability. These two models can be utilized to categorise and determine the severity of threats.

Therefore, by using CyberSANE Threat Taxonomy in conjunction to STRIDE and DREAD models, a threat analysis matrix can be created reflecting the severity and category of potential threats. In particular, the threats listed in sources such as CyberSANE taxonomy can be mapped to or modelled to represent the intention of a threat actor based on STRIDE shown in Table 15. Also, threats can be rated by following the customized and accompanying questions shown in Table 16 and Table 17.

Category	Consideration
Spoofing (S)	Attackers masquerade as a legitimate user, system or application element
Tampering (T)	Attackers modify or tamper assets in transit or in-store
Repudiation (R)	Attackers perform actions that cannot be traced
Information Disclosure (I)	Attackers disrupt or interrupt normal operations of the asset
Elevation (E)	Attackers obtaining access privilege to an asset without legitimate authority

Table 15: Threat Categorisation Matrix

Category	Question
Damage Potential (D)	How extensive is the damage potential?
Reproducibility (R)	How easy it is for the threat to be repeated or reoccur?
Exploitability (E)	How easy is it to launch the treat?
Affected Users (A)	What is the estimate of users that will be affected?
Discoverability (D)	How easy is it to discover the vulnerabilities?

Table 16: DREAD Model

Category	3 (High)	2 (Medium)	1 (Low)
Damage Potential (D)	Complete system or data destruction, and unavailability of assets and critical functions	Compromise or impact a subset of assets and critical functions	Minor impact to a small number of assets and critical functions
Reproducibility (R)	A threat could be reproduced to compromise assets and critical functions	The threat can be reproduced, but only by an authorised user	The threat is very unlikely to be replicated
Exploitability (E)	A novice threat actor can easily compromise assets and bring down critical function	Attack tools freely available, or an exploit is easily performed using novice tools	Advanced programming and deep knowledge, with custom or advanced tools
Affected Users (A)	All users	Some users but not all	None
Discoverability (D)	Vulnerabilities in the asset are very noticeable and can be easily exploited	Weaknesses in the assets are rarely discovered	Vulnerabilities are hardly present and rarely discovered

Table 17: Threat Rating Matrix

Values	Rating
12 to 15	High
8 to 11	Medium
5 to 7	Low

Table 18: Threat Severity Matrix

The above scales can be used to rate and determine the severity of every threat according to the DREAD model. The questions can also be modified or extended accordingly. To apply the model, a rating table is used with corresponding values of 3, 2 and 1 to represent (3) high, (2) medium and (1) and low, respectively. The outcome can fall within the scope of 5 to 15 to denote threat severity with from low to high. Threats with overall ratings of 12-15 can be treated as having 'High Severity', 8-11 as 'Medium Severity', and 5-7 as 'Low Severity' as shown in Table 18.

5.4.2.2 Step 2.2 – Identify Vulnerabilities

The second step involves the identification of vulnerabilities which can be exploited by a threat to compromise one or more assets. Numerous vulnerabilities exist at different levels and they need to be properly identified and modelled to provide a consolidated view of the possible threats in relation to the vulnerabilities. The identification and modelling of vulnerabilities is supported by *Vulnerability* concept, whereby the different types of vulnerabilities associated with assets are identified using *Type* attribute.

At this point, it is critical for the developer to explore databases to efficiently identify vulnerabilities. The National Vulnerability Database (NVD), is a "U.S government repository of standards based vulnerability management data" that reports known vulnerabilities (Booth, Rike et al. 2013). It helps developers to access information and understand the nature of the common. Similarly, the Common Vulnerabilities and Exposures (MITRE 2008) provides a glossary that uses Security Content Automation Protocol for collecting information about security vulnerabilities and exposures. The main purpose of the glossary serves as a standardized way by which each known vulnerability or exposure is identified, as well as an industry baseline for communicating and dialoguing around given vulnerabilities.

Accordingly, vulnerabilities need to be rated according to severity, which is enabled by the *Rating* attribute. To ensure consistency, a scoring system for indicating the severity rating for each security vulnerabilities can be used. The security severity rating helps developers to determine how best to approach a vulnerability based on CVSS (NVD 2019) rating, which consists of a formula made up of three main metric groups: base, temporal and environmental. The Base metric assess the severity of a vulnerability based on its intrinsic characteristics, which are mostly constant over time. The Temporal Metrics is based on factors that change over time, such as availability of exploit code. The Environmental metrics considers factors such as the presence of mitigations in the cyber environment. The rating system also consist of numerical score that produces a score ranging from 0 to 10, which can be mapped to qualitative ratings as shown in Table 19.

Once vulnerabilities are identified and assigned a severity score, an association is created in the model between the potential threats that could exploit the vulnerability, as well as the assets associated with the vulnerability itself.

Rating	Score
Low	0.0
Low	0.1-3.9
Medium	4.0-6.9
High	7.0-8.9
Critical	9.0-10.0

Table 19: Vulnerability Rating

5.4.2.3 Step 2.3 – Identify Risks

The output of threat analysis provides a list of potential security threats and the potential severity on cyber assets. The second step provided a method by which vulnerabilities can be identified and assigned a severity rating. A threat can have multiple impacts to the critical information infrastructure, particularly assets and critical functions, and therefore, it is important to identify the resulting risks as a result of threats exploiting vulnerabilities.

Therefore, the goal of this step is to identify and assess the potential outcomes of a successful threat on cyber asset, such as the possibilities for destruction, modification, or interruptions to assets or critical functions. This can be instantiated using the *Risk* concept of the meta-model. To meet the goal of this step, the developer can assess the likelihood and impact of risk, using the *Likelihood* and *Impact* properties of *Risk* concept.

In addition, there are many approaches to perform risk analysis which can be utilized for this purpose. However, it is essential for the developer to define an approach that makes the identification of risks as accurate as possible. This will help in ensuring that major risks are not prioritized overlooked. The basic factors that are considered for estimating risk likelihood include *threat agent factors* and *vulnerability factors*, while factors for estimating risk impact include *technical factors* and *business impact factors*.

Threat factors for estimating risk likelihood involve assigning each factor a set of options, and each option has a likelihood rating from 0 to 9 associated with it (as shown in Table 20). Similarly, technical impact factors are used for determining impact. Each factor is assigned a set of options, and each option is associated with an impact rating from 0 to 9 as shown in (Table 21). Therefore, the developer can be able to determine the severity of risks to assets and business functions, as well as ensure that priority is given to more serious risks.

Threat Factor		**0 to < 3 (Low)**	**3 to < 6 (Medium)**	**6 to 9 (High)**
Factor	**Description**			
Ease of Discovery	How easy is it for this group of threat agents to discover this vulnerability?	Practically impossible	Difficult	Substantially easy
Ease of Exploit	How easy is it for this group of threat agents to exploit this vulnerability?	Theoretical	Difficult	Substantially easy
Awareness	How well known is this vulnerability to this group of threat agents?	Unknown	Obvious	Public knowledge
Intrusion Detection	How likely is an exploit to be detected?	Active detection mechanisms	Logged & reviewed	Not reviewed

Table 20: Risk Likelihood

Technical Impact		**2**	**6**	**6**	**7**	**9**
Factor	**Question to ask**					
Loss of Confidentiality (C)	How much data could be disclosed and how sensitive is it?	Minimal non-sensitive data disclosed	Minimal critical data disclosed	Extensive non-sensitive data disclosed	Extensive critical data disclosed	All data disclosed
Loss of Integrity (I)	How much data could be corrupted and how damaged is it?	Minimal slightly corrupt data	Minimal seriously corrupt data	Extensive slightly corrupt data	Extensive seriously corrupt data	Extensive seriously corrupt data
Loss of Availability (A)	How much service could be lost and how vital is it?	Minimal primary services interrupted	Extensive secondary services interrupted	Extensive primary services interrupted	Extensive primary services interrupted	All services completely lost

Loss of Accountability (AC)	Are the threat agents' actions traceable to an individual?	All services completely lost	Possibly traceable	Possibly traceable	Possibly traceable	Completely anonymous

Table 21: Risk Impact to Technical Impact

Threat														**Vulnerability**		**Risk**						
Threat Type	Description	Threat Category						Target Assets	Threat Severity					Type	Rating	Type	Residual	Cascading	Technical impact			
		S	*T*	*R*	*I*	*D*	*E*		*D*	*R*	*E*	*A*	*D*						*C*	*I*	*A*	*AC*

Table 22: Threat, Vulnerability and Risk Register

5.4.3 Activity 3 – Incident Response

This activity focuses on the specification of the essential aspects of incident response such as managing, containing, minimising the impact of a cyber incident, as well as recovery actions. It consists of concepts as *CyberCourseOfAction, Critical Infrastructure, Impact, Actor, Control Mechanism.* The activity aims to capture incident response strategies that can be used to identify cyber-incidents and support the understanding of the relevant response strategies for effective and efficient analysis of a cyber incidents in terms of containment, eradication, and actions. Therefore, the activity mainly entails the detection, analysis, containment, eradication, and recovery aspects of incident response process. This activity's output aims to represent incident response activities tailored to the specific needs of a CI. Hence, the activity is decomposed to enable the creation of sub-models, which are described in the following section.

5.4.3.1 Step 1 – Identify Techniques for Detection and Analysis

This step provides a meticulous analysis of one or multiple incidents. The analysis considers attributes such as the severity and priority of incidents. Primarily, the step consists of two tasks namely cyber incident *detection* and *analysis*, which pave way for the subsequent step for containment, eradication, and recovery. The concepts that enable this step are *CyberIncident*, *Impact, Assets,* and *Actor.* A developer can initiate the step from *CyberIncident* concept that consists of attributes as: *Incidenttype, Description, AffectedAssets,* and *Severity.* Other concepts included in this step include *Actor* role, *Impact,* and *CriticalInfrastructure.*

5.4.3.1.1 Incident Detection

This phase entails the application of different techniques and tools for detecting cyber incidents. It is imperative for the developer to collect and log security event data for detecting incidents and supporting incident analysis using the *Evidence* and *Incident Type* attribute of *CyberIncident* concepts. The *Evidence* concept enables a set of automated detection capabilities which are used to identify a cyber incident. Hence, incidents can be detected through various means, with varying levels of details fidelity. Automated detection capabilities such as log management tools, antivirus software, intrusion detection systems, intrusion detection systems, and vulnerability scan data can be used to detect incidents. Incidents may also be detected through manual means such as user reports, especially because some incidents can be easily detected manually, whereas others can go undetected without automated processes. Moreover, established logging standards and procedures that ensure adequate collection and analysis information by logs and security

software is essential. In this direction, the CyberSANE system provides a combination of active approaches to detect and analyse anomaly activities and attacks in real-time. Specifically, the LiveNet component of CyberSANE can be leveraged for identifying such incidents because it integrates security monitoring sensors with network-based intrusion detection systems, anomaly detection modules, and endpoint protection solutions for accessing and extracting information in order to detect complex attacks.

5.4.3.1.2 Incident Analysis

Furthermore, the analysis part evaluates and validates a cyber incident in order to determine the scope of an incident, the methods used and the targeted vulnerabilities, which are enabled using concepts and attributes of the modelling language as *Priority, AffectedAssets Impact, Threat, Vulnerabilities, Risks,* and *Control Mechanism*. Some incidents are relatively more important and require urgent response compared to others. Hence, a developer should assign an incident priority scheme based on its impact and urgency for resolution. This "*Priority"* attribute enables a developer to determine incident priority according to a prioritization matrix. The attribute "*AffectedAssets"* is used to identify the assets that have been affected by a cyber incident by creating an associated between the cyber incident and the assets perceived to be affected.

Similarly, the adverse effects in terms of the consequences of an incident caused to assets is quantified through the *Impact* Concept using qualitative or quantitative values. The *Control Mechanism* concept enables the developer to represent and understand the existing control actions, processes and mechanisms being used to prevent or mitigate potential incidents, which are categorized according to *Corrective Mechanism, Preventive Mechanism,* and *Detective Mechanism*. Furthermore, it is worth noticing that control mechanisms do not always provide complete security and protection of assets as desired, hence, the attribute *Measure of Effectiveness* enables the assessment of the effectiveness of existing control measures in terms of relevance and robustness to control mechanisms to address cyber incident. Therefore, these considerations will provide the developer enough insight for assessing the subsequent containment and mitigation strategies according to the order based on which cyber incidents should be handled.

To ensure consistency, it is important to use an incident prioritization matrix for determining incident priority. Cyber incident prioritization can be performed according to three criterion: (a) functional impact of the incident (such as current and likely future negative impact to critical functions), (b) information impact of the incident (such as the confidentiality, integrity and availability of assets), (c) recoverability from the incident (such as time and types of resources that are required to recover from the incident) NIST (NIST 2012). The purpose for this prioritization lies on the presumption that highly rated incidents must be handled and resolved before low rated incidents. Although the developer can decide the desired and appropriate criterion, the functional impact criteria is more suitable for prioritizing incidents according to negative impacts on critical functions, thereby it is considered in this process and presented in Table 23.

Category	Rating	Definition
None	0	No effect to the ability to provide critical functions to all users
Low	1	Minimal effect, critical functions can be provided to all users but with limited efficiency
Medium	2	The inability to provide critical functions to a subset of users

High	3	Complete incapacity to provide critical functions to any users

Table 23: Functional Impact Categories for Incident Prioritization

Furthermore, as mentioned earlier, the impact or magnitude of harm resulting from a cyber incident is estimated through the *Impact* concept. In particular, the *Severity* attribute of the concept is used to determine an impact in terms of loss, failure or damage that could result in adverse effect to critical functions or assets. To support the determination of incident impact, a useful matrix for the determination of impact to organizational assets and functions can be used. The assessment scales shown in the impact matrix can be tailored according to organization-specific conditions as shown in Table 24.

Qualitative Values	Semi-Qualitative Values		Description
Very High	95-100	10	The impact of an incident is sweeping, affecting almost all of assets and critical functions
High	80-95	8	The impact of an incident is extensive, affecting most of assets, including many critical functions
Moderate	21-79	5	The impact of an incident is substantial, affecting a signification portion of assets including some critical functions
Low	5-20	2	The impact of an incident is limited in nature, affecting some assets but not involving critical functions
Very low	0-4	0	The impact of an incident is minimal and negligible, involving a few if any assets and involving no critical functions

Table 24: Incident Impact Rating

Also, the effectiveness of control can be determined using a standard quality metrics for each of the control categories. ISO/IEC 27004:2016-12-15 provides guidelines intended to assist organizations to evaluate the information security performance and the effectiveness of the ISMS" (27004:2016 2016). The guideline identified a measurement method and four groups of controls that can be measured: (a) management controls such as security policy, security procedures, business continuity plans, (b) business processes such as risk assessment and risk management process, (c) operational controls such as operational procedures, change control, problem management, back up, and secure disposal, (d) technical controls as patch management, anti-virus controls, IDS, firewall and content filtering. Ultimately, this ISO guideline can be used to assess the effectiveness of controls using the attribute *Measure of Effectiveness.*

5.4.3.2 Step 2 – Define Incident Containment, Eradication and Recovery Actions

The eventual end goal after progressing through the preceding step is to apply actions that successfully contain and eradicate an incident. It is crucial to implement strategies to contain and remove incidents in order to avoid overwhelming assets and increase the potential impacts to critical functions. Hence, this step focuses on the analysis of the appropriate and implementable incident response strategies to address cyber incidents identified in the preceding step. The goal here is to allow a developer to create an independent model that captures the essential strategies for containing and reducing the potential impacts of an incident, as well as the strategies for the actual restoration of affected assets. Like the previous models, the modelling activity in this step uses concepts from the incident handling modelling language, as well as the integration of various techniques and practices to support the modelling activity.

Fundamentally, the central concept that enable modelling at this level is the *CyberCourseOfAction,* which entails a combination of processes or measures for responding to or mitigating the potential impacts of predefined or anticipated cyber incidents. As the control actions for containing and removing an incident may vary according to incident types, strategies for cyber course of actions consider these variations to enable the implementation of separate strategies for each major incident type. Therefore, *CyberCourseOfAction* strategies can be implemented from to two different perspectives namely (i) procedural course of actions dealing with control actions such as security policies and awareness and training, and (ii) technical course of actions as cryptography and access control. These two categories of *CyberCourseOfAction* are modelled according to *ProceduralCourseOfAction* and *TechnicalCourseOfAction* inherence.

Therefore, in defining and modelling technical and procedural course of actions, it is essential to consider a standard set of actions for cyber defence that provide specific and actionable practices and mechanisms to contain, mitigate and eradicate most of pervasive and dangerous cyber-attacks. For example, CIS CSC (Centre for Internet Security 2018) proposed a set of 20 controls categorised into 3 prioritized defence-in-depth and best practices (as shown in Table 25), and they are generally regarded as effective because they are derived from the "most common attack patterns highlighted in the leading threat reports and vetted across a very broad community of government and industry practitioners". In addition, the controls are mapped to regulations commitments (such as GDPR, PCI DSS, HIPAA, FISMA) and compliance commitments (such as NIST 800-53, ISO 270000 series, ITIL) that are applicable to most CIIs.

Therefore, the controls provided by CIS can be adopted and mapped both to the procedural and technical course of actions. In particular, the Organisational CIS Controls and Basic CIS controls can be aligned to *ProceduralCourseOfAction,* whereas Fundamental CIS controls are associated with *TechnicalCourseOfAction.* This mapping will support the incident response team to have a better understanding of modelling the containment, eradication, and recovery efforts following the best practices.

Control Category	Control Types
Basic CIS Controls	Inventory and Control of Hardware Assets
	Inventory and Control of Software Assets
	Continuous Vulnerability Management
	Controlled use of Administrative Privileges
	Secure Configuration for Hardware and Software on Mobile Devices, Laptops, Workstations and Servers
	Maintenance, Monitoring and Analysis of Audit Logs
Foundational CIS Controls	Email and Web Browser Protections
	Malware Defences
	Limitation and Control of Network Ports, Protocols and Services
	Data Recovery Capabilities

	Secure Configuration for Network Devices (such as firewalls, Routers and Switches)
	Boundary Defences
	Data Protection
	Controlled Access Based on the Need to Know
	Wireless Access Control
	Account Monitoring and Control
Organisational Controls	Implement a Security Awareness and Training Program
	Application Software Security
	Incident Response and Management
	Penetration Tests and Red Team Exercises

Table 25: CIS Control Category and Types

Qualitative Values	Semi-Qualitative Values		Description
Very High	95-100	10	The impact of an incident is sweeping, affecting almost all of assets and critical functions
High	80-95	8	The impact of an incident is extensive, affecting most of assets, including many critical functions
Moderate	21-79	5	The impact of an incident is substantial, affecting a signification portion of assets including some critical functions
Low	5-20	2	The impact of an incident is limited in nature, affecting some assets but not involving critical functions
Very Low	0-4	0	The impact of an incident is minimal and negligible, involving a few if any assets and involving no critical functions

Table 26: Incident Response Matrix

5.5 Evaluation

In this section, we present an example of the modelling activities using Lightsource Lab scenario from the CyberSANE project. The example assumes a Jigsaw ransomware cyber-attack took place to enable a brief demonstration of the conceptual model and process of the modelling language.

5.5.1 Pilot Study

Lightsource Labs provides solar energy production, storage and distribution services, and is collaborating with various distribution points. In this context, LSE operates an integrated Smartly Integrated Distributed Energy (SIDE) platform and a number of digital services on top, helping energy “prosumers”, utilities and grid operators to optimize power flows, secure the electricity grid and finally reduce the cost of electricity. The SIDE platform constitutes a smart software-hardware solution optimised for Grid 2 Home / Home 2 Grid optimization of a distributed generation system. In order to meet its objectives, this platform incorporates a bundle of components such as: (i) A range of web apps for the end user (SIDE UIs) that enable users to see in real time the power flow between the solar system, the battery and the grid of their household; (ii) the SIDE gateway which is an intermediate device between sensors, smart meters, inverters, the battery and appliances and the SIDE Platform that creates value from data collection and control; (iii) the SIDE Virtual Power Plant (VPP) that is a cloud infrastructure and software platform which operates a smart grid network of a population of distributed assets interconnected in a secure way via Side Gateway; (iv) the SIDE CRM that is a bespoke back-office CRM application which automates the entire process of the business; (v) the SIDE Panel that is an electric panel specially designed to accelerate the installation process of the system and eliminate connectivity errors; and the SIDE IoT platform which is our abstract software framework running. Various combined cyber-attacks on the “Solar Energy Production, Storage and Distribution Service” may affect the examined solar energy service. From the cyber part, attacks against back-end SIDE Platform such as gaining unauthenticated remote access to IoT components and other entities to disrupt services and change their data set points or state. Other cyber-attacks may target against the IT and communication systems that are used to process the sensed data and transmit them to the corresponding IT systems.

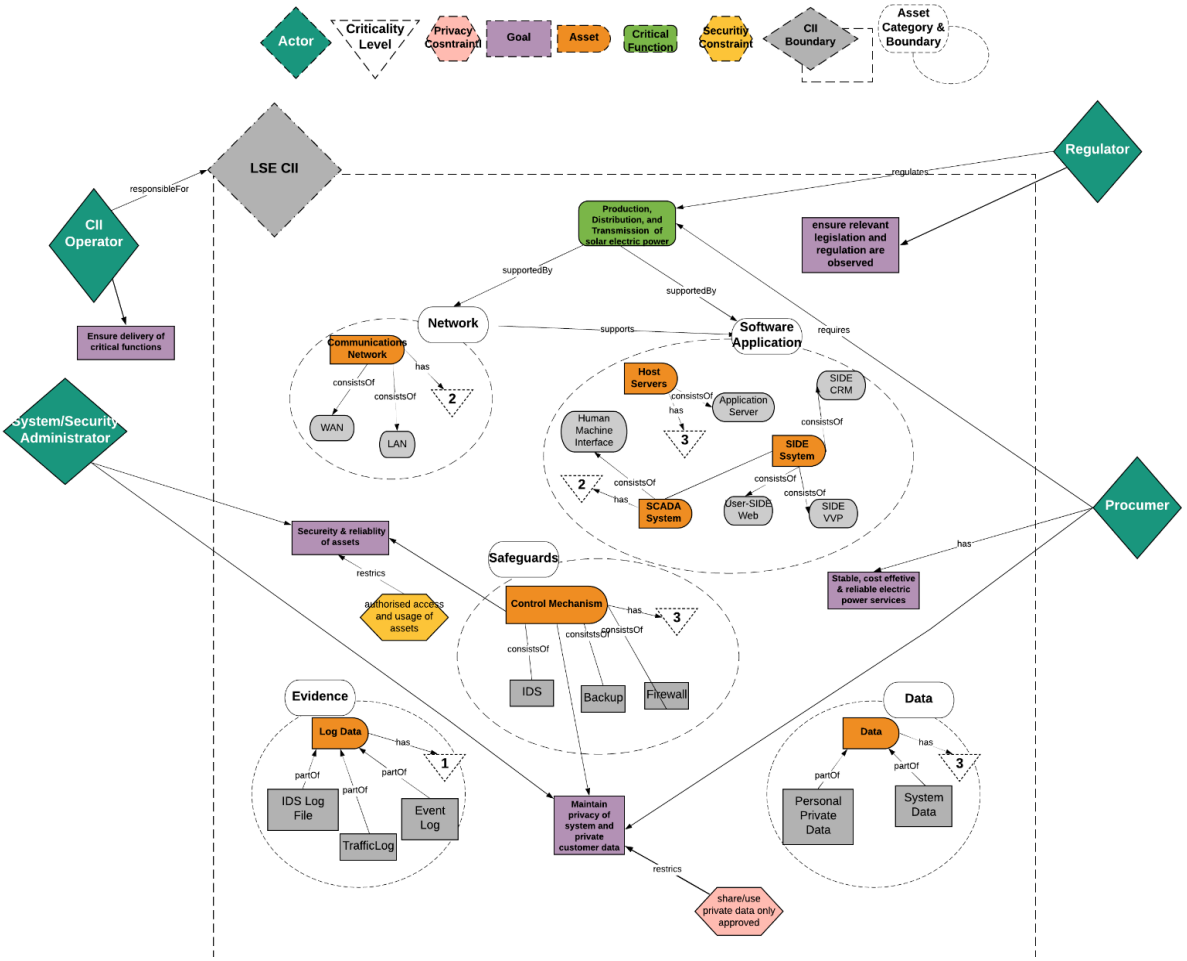

Figure 6: CII Model

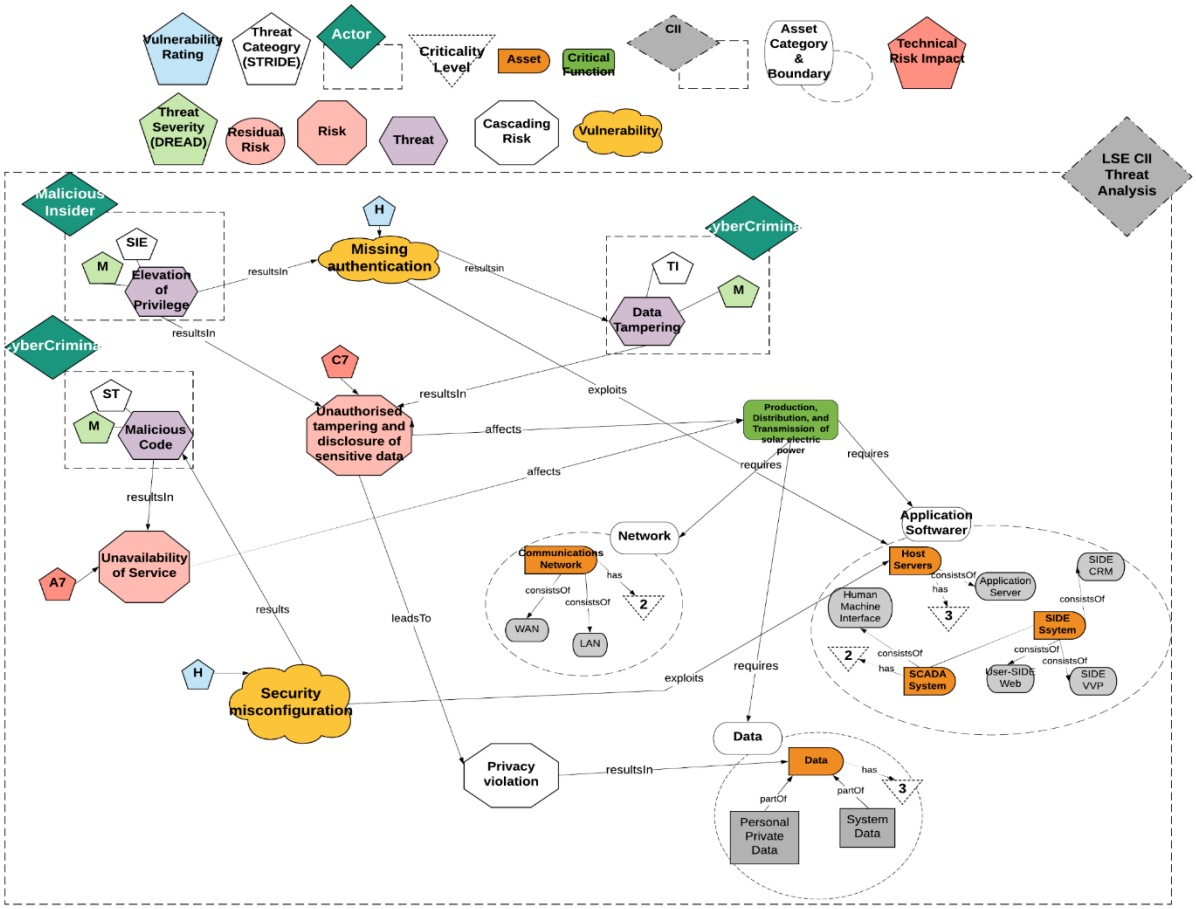

Figure 7: Threat Analysis

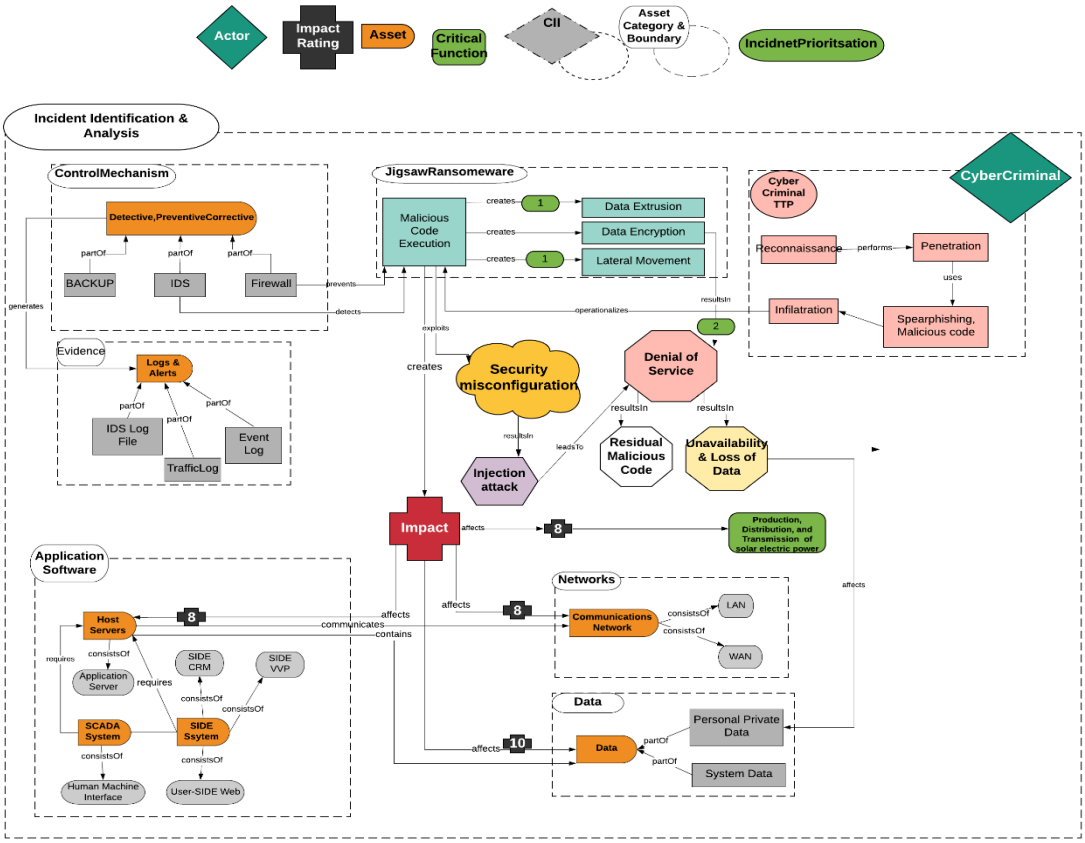

Figure 8: Incident Detection and Analysis

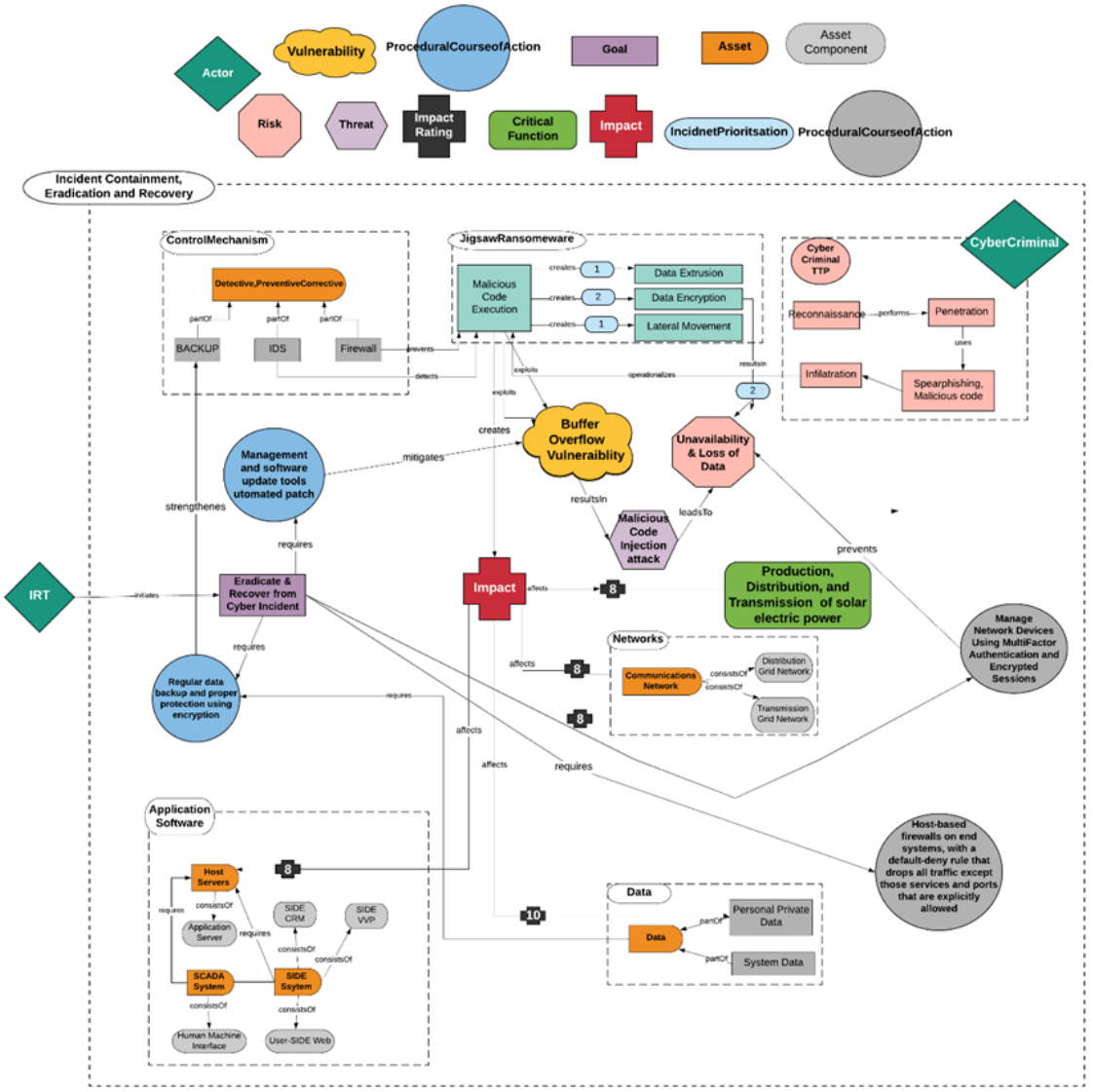

Figure 9: Incident Containment, Eradication and Recovery

6 Integrating Sharing and Anonymization

Since information of the CI and CII is critically sensitive and may describe vulnerabilities of a system, sharing this information become an error prone process accompanied by multiple challenges. Therefore, protecting sensitive information from potential abuse is one of the main task in achieving strong defence capabilities. This chapter describes potential integration of the C3ISP platform into the CyberSANE ecosystem to provide advanced and secure data sharing capabilities. Furthermore, the chapter describes the communication between the C3ISP platform and the PrivacyNet component to enable the enforcement of privacy-preserving operations according to fine-grained security policies continuously enforced.

6.1 C3ISP Collaborative Framework

C3ISP is a collaborative and confidential information sharing and analysis framework which was funded under H2020-EU.3.7. (European Commission CORDIS 2016). The framework can be deployed as a service to enhance the cyber-security protection of various types of organizations, by acting as a flexible and controllable medium for the sharing of data between them. The main advantage of the C3ISP framework comparing to other approaches is the secure sharing, storing and analysis of information. This is achieved through the combination of security mechanism, including continuous enforcement of security policies. Data owners can define security constraints through security policies written in the human-readable Data Sharing Agreements (DSAs).

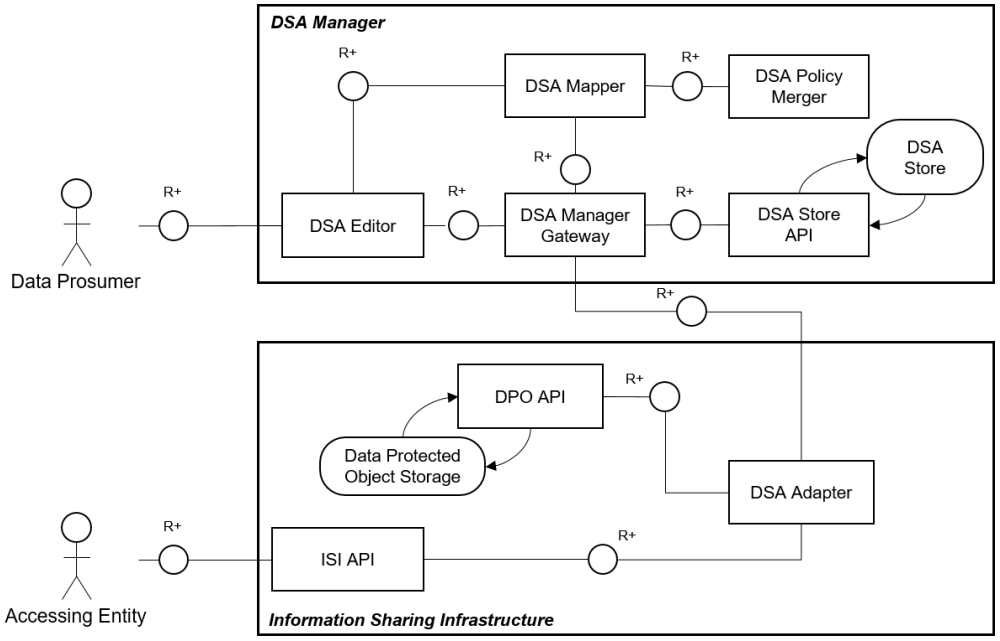

Figure 10: High level architecture of the C3ISP platform integrated into ShareNet component

Figure 10 depicts the high level architecture of the C3ISP platform as a part of the ShareNet component. It includes two main components, namely DSA Manager and Information Sharing Infrastructure (ISI). The DSA Manager supports data prosumers with an infrastructure for defining, storing and editing security policies. Data owners may also map defined DSA to their information, thus enabling data access and usage control. The DSA Manager interacts with the ISI using DSA Manager Gateway. The ISI component of the C3ISP framework is used to provide information

sharing infrastructure. In particular, it ensures that only authorised entities may access information. This is achieved through the enforcement of fine-grained security policies. Furthermore, the ISI component can execute specific operations, called obligations, to preserve sensitive information from it potential disclosure. The following section discusses the interactions between the C3ISP platform integrated into the ShareNet component and PrivacyNet.

6.1.1 Enforcing Data-Manipulation Operations

To enable continuous enforcement of security policies the C3ISP platform implements UCON paradigm described in Chapter 2. Security policies may specify certain obligations, which may require system to perform certain operations on the corresponding dataset before, after or during the usage. For example, obligations may require from the system to notify a data owner whenever others access his/her dataset. Furthermore, security policies may require the C3ISP platform to enforce one or multiple Data Manipulation Operations (DMOs) on the corresponding information before providing access. For example, the U-XACML obligations may specify that the C3ISP platform must execute an encryption technique on the dataset if the subject requesting data access belongs to a particular organization. In the CI and CII security context, considering the criticality of information shared by organizations, enabling ongoing control and enforcement of operations to prevent the potential abuse of sensitive data is considered as an essential feature of any data-sharing platform.

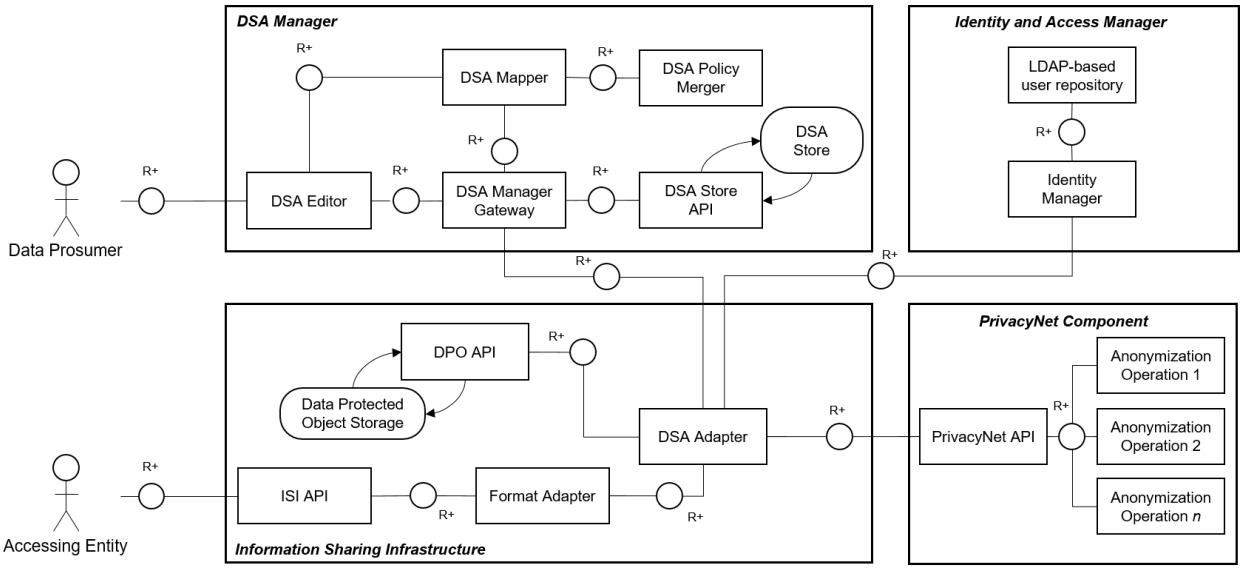

Figure 11: Interactions between C3ISP platform and PrivacyNet

Therefore, to allow the CyberSANE framework to share information and, if necessary, enforce anonymization operations to preserve privacy, the ShareNet and the PrivacyNet systems will interact with each other. Figure 11 depicts a high-level architecture of C3ISP platform integrated into ShareNet component of the CyberSANE ecosystem and PrivacyNet component that offers the enforcement of anonymization operations.

Hence, the ShareNet component will enforce security policies and provide data usage features to enable continuous control over data usage. On the other hand, if the security policy specifies the need of an anonymization operation on the corresponding dataset, then the ShareNet component will forward the corresponding information together with the specified data anonymization operation to the PrivacyNet system, which in turn will execute the operation to preserve privacy and return the sanitized version of data. For example, the security policy written in the U-XACML

format may include the obligation element. The obligation element could specify that before sharing the dataset with the organization belonging to the financial sector (this could be defined through attributes), the system must execute the pseudonymization operation on IP addresses specified in that dataset. In this case, the ShareNet component of the CyberSANE framework will pass the dataset with the requested anonymization operation and the corresponding attribute to the PrivacyNet system.

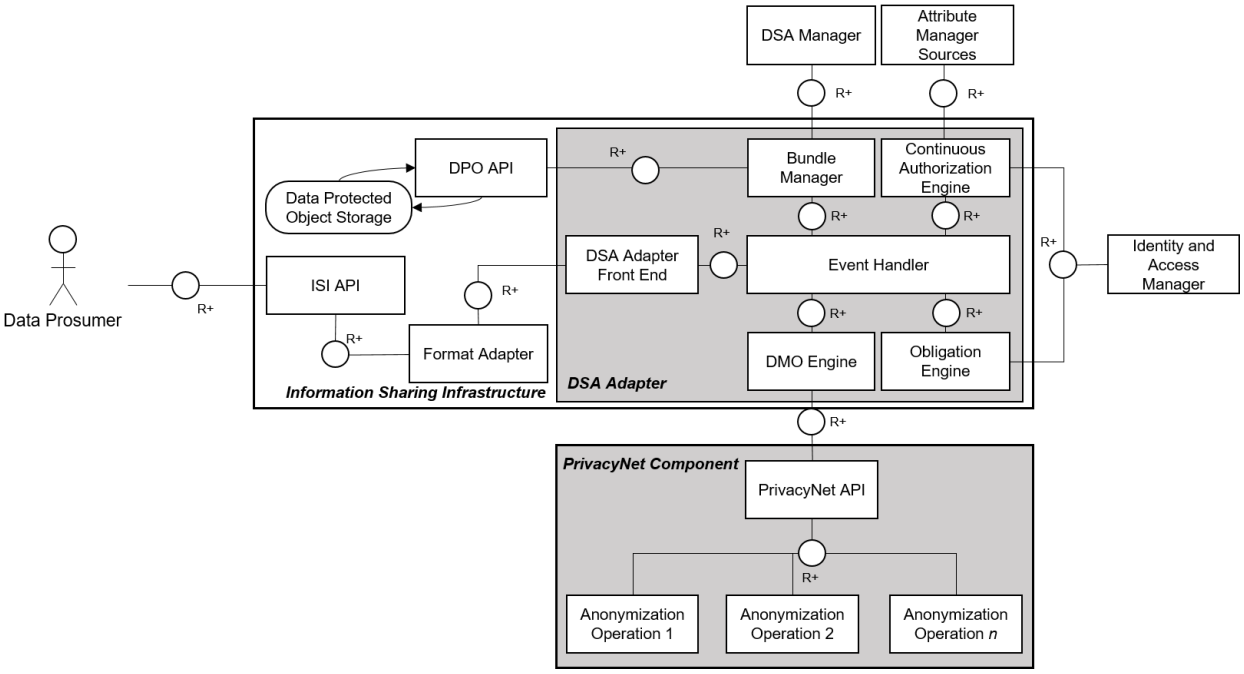

Figure 12: Interactions between ISI and PrivacyNet

Figure 12 depicts detailed view of the interaction between ISI element of the C3ISP platform and PrivacyNet system. As shown on the Figure 12, the main component of the ISI is a Data Sharing Agreement (DSA) Adapter, which includes multiple elements. Thus, to enable continuous enforcement of UCON policies, the DSA Adapter includes the Continuous Authorization Engine (CAE), which is also capable of retrieving attributes from different sources (e.g., action attributes, resource attributes, etc.) using multiple Attribute Managers. As mentioned, UCON policies may require the execution of obligations. For this reason, the module called Obligation Engine is responsible for the execution of specific operations if certain conditions take place. Finally, the Data Manipulation Operation (DMO) Engine component is in charge of executing the Data Manipulation Operation (DMO) prescribed as part of the decision process on the dataset or by any obligation prescribed by the DSA paired with that dataset. In fact, besides determining whether the dataset can be accessed by the entity, the decision process also determines a set of operations, which must be executed on that dataset before being released to the requestor. For example, a DSA paired to a system log could require that all the IP addresses present in that log must be anonymized before releasing this log to a third party. Therefore, to satisfy these needs, the DMO Engine interacts with the PrivacyNet component by exploiting its API. Following section describes the interactions between ISI as a part of the ShareNet component and PrivacyNet system.

6.1.2 ShareNet and PrivacyNet operation Workflow

This section describes the operational workflow between ShareNet and PrivacyNet components of the CyberSANE platform. Since organizations share CTI reported through the STIX standard

to enable automated sharing with other platforms, to facilitate this requirement, we have designed a format-adapter toolbox. The toolbox converts multiple data samples, including the Netflow dataset and email messages to the format of the STIX standard. After performing this operation, it then forwards the STIX package to the DSA Adapter element. In fact, entities can upload both the Netflow and spam email datasets without changing format. However, in this case, it will not be possible to share those datasets with other platforms.

The designed format-adapter toolbox converts each network-traffic flow sample to the existing STIX Cyber-observable Object (SCO), which can be also extended with additional features. Furthermore, for each source IP address the tool can generate the Indicator STIX Domain Object (SDO), which is used to detect suspicious or malicious cyber activity. Figure 13 depicts the complete workflow between components.

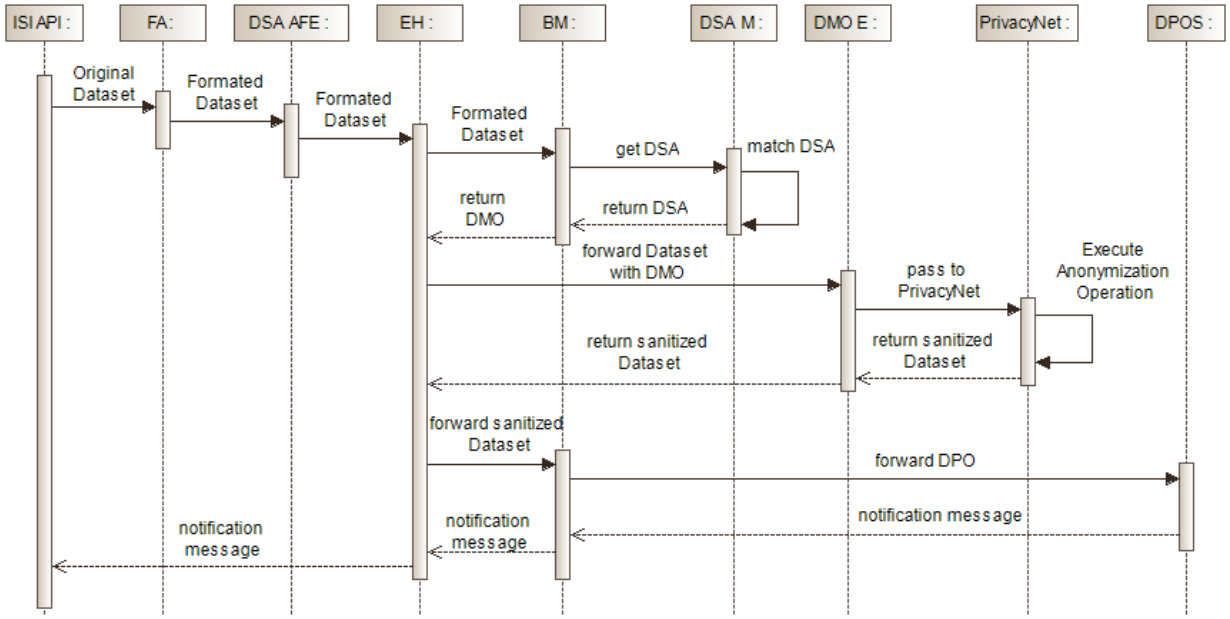

Figure 13: Complete workflow diagram

Data Prosumers can upload data by using ISI API. After that, the designed format-adapter toolbox will initiate the operation for formatting the uploaded dataset. Once the toolbox formats the input dataset to the STIX standard it forwards the formatted dataset to the DSA adapter invoking DSA Adapter Frontend (DSA AFE) in order to create a Data Protected Object (DPO) by using the Bundle Manager. The Bundle Manager is used for both packing and unpacking operations. Each DPO is the encrypted and compressed bundle that contains uploaded data, related metadata, and the ID of corresponding security policy defined by the data-owner. We consider that the security constrains defined with the DSA requires platform to execute DMO operation once data has been uploaded to the system. Therefore, the Event Handler (EH) element of the DSA Adapter will invoke the DMO Engine, depicted as DMO E, by forwarding the encoded dataset together with the required operation. In turn, the DMO Engine (depicted on Figure 13 as DMO E) invokes the PrivacyNet component of the CyberSANE platform exploiting available PrivacyNet API. The PrivacyNet component execute the requested operation on the encoded dataset and return its sanitized version to the DMO Engine of the DSA Adapter. Once the Event Handler receives the sanitized version of the dataset it invokes the BM for packing data and store it in the Data Protected Object Storage (DPOS). Finally, the notification message is produced informing data owner about successful operations. Thus, the platform will provide access only to the sanitized version of the dataset if security policies have been satisfied. Otherwise, the final decision for

providing access will result in denial. The following section provides two examples of enforcing anonymization operations on two different datasets.

6.1.3 Enforcing anonymization operation on CTI data

In this part we report an example of enforcing anonymization functions on different datasets. In particular, we use publicly available Netflow datasets[29] for training Machine Learning-based Network Intrusion Detection Systems (NIST) (Mohanad Sarhan 2020) and spam email datasets available online[30]. Each network-traffic sample of the Netflow dataset is characterized by a set of features (see Table 27). The features were extracted from the publicly available pcap files. Furthermore, each network-traffic was labelled with its respective attack categories. Hence, the total number of data flows is more than 600k, where more than 97% related to attack samples and less than 3% are benign.

Feature	Description
IPV4_SRC_ADDR	Indicates source IP address
IPV4_DST_ADDR	Indicates destination IP address
L4_SRC_PORT	Defines the source port
L4_DST_PORT	Defines the destination port
PROTOCOL	IP protocol identifier byte
TCP_FLAGS	Cumulative of all TCP flags
L7_PROTO	Layer 7 protocol (numeric)
IN_BYTES	Incoming number of bytes
OUT_BYTES	Outgoing number of bytes
IN_PKTS	Incoming number of packets
OUT_PKTS	Outgoing number of packets
FLOW_DURATION_MILLISECONDS	Flow duration in milliseconds

Table 27: Netflow features

In fact, the Netflow dataset also provide two additional features, namely label and attack type. However, these features are used for ML training, and thus out of the scope for anonymization.

The security policy that we consider for the Netflow dataset requires the platform to anonymize the destination IP addresses using pseudonymization method. Table 28 reports an example both of the Netflow sample and its sanitized version after invoking the PrivacyNet system.

[29] https://staff.itee.uq.edu.au/marius/NIDS_datasets/#RA1
[30] http://untroubled.org/spam/

Netflow data in STIX format	Sanitized version of the Netflow data in STIX format

{
"type": "bundle",
"spec_version": "2.1",
"objects": [
{
"type": "observed-data",
"spec_version": "2.1",
"id": "observed-data--88afc9ed-efdb-4e67-8516-5acc…",
"created": "2021-02-06T17:58:13.000Z",
"modified": "2021-02-06T17:58:13.000Z",
"first_observed": "2021-02-06T17:58:13.000Z",
"last_observed": "2021-02-06T17:58:13.000Z",
"number_observed": 1,
"object_refs": [
"ipv4-addr--2f201dba-0bd8-4cdc-b36c-1ebaca91b5be",
"ipv4-addr--cdc68824-3a92-4ef7-8b49-d5a8f34dc1e5",
"network-traffic--12d23621-54ee-4826-b66e-aa1ab…"
]
},
{
"type": "ipv4-addr",
"spec_version": "2.1",
"id": "ipv4-addr--2f201dba-0bd8-4cdc-b36c-1eba…",
"value": "192.168.100.6",
},
{
"type": "ipv4-addr",
"spec_version": "2.1",
"id": "ipv4-addr--cdc68824-3a92-4ef7-8b49-d5a8…",
"value": "192.168.100.149"
},
{
"type": "network-traffic",
"spec_version": "2.1",
"id": "network-traffic--12d23621-54ee-4826-b66e-aa1ab…",
"src_ref": "ipv4-addr--2f201dba-0bd8-4cdc-b36c-1eba…",
"dst_ref": "ipv4-addr--cdc68824-3a92-4ef7-8b49-d5a8…",
"protocols": [
"ipv4",
"tcp"
],
"src_byte_count": 217753000,
"src_packets": 4521,
}
]
}
	{
	"type": "bundle",
	"spec_version": "2.1",
	"objects": [
	{
	"type": "observed-data",
	"spec_version": "2.1",
	"id": "observed-data--88afc9ed-efdb-4e67-8516-5acc…",
	"created": "2021-02-06T17:58:13.000Z",
	"modified": "2021-02-06T17:58:13.000Z",
	"first_observed": "2021-02-06T17:58:13.000Z",
	"last_observed": "2021-02-06T17:58:13.000Z",
	"number_observed": 1,
	"object_refs": [
	"ipv4-addr--2f201dba-0bd8-4cdc-b36c-1ebaca91b5be",
	"ipv4-addr--cdc68824-3a92-4ef7-8b49-d5a8f34dc1e5",
	"network-traffic--12d23621-54ee-4826-b66e-aa1ab…"
	]
	},
	{
	"type": "ipv4-addr",
	"spec_version": "2.1",
	"id": "ipv4-addr--2f201dba-0bd8-4cdc-b36c-1eba…",
	"value": "192.168.0.0",
	},
	{
	"type": "ipv4-addr",
	"spec_version": "2.1",
	"id": "ipv4-addr--cdc68824-3a92-4ef7-8b49-d5a8…",
	"value": "192.168.100.149"
	},
	{
	"type": "network-traffic",
	"spec_version": "2.1",
	"id": "network-traffic--12d23621-54ee-4826-b66e-aa1ab…",
	"src_ref": "ipv4-addr--2f201dba-0bd8-4cdc-b36c-1eba…",
	"dst_ref": "ipv4-addr--cdc68824-3a92-4ef7-8b49-d5a8…",
	"protocols": [
	"ipv4",
	"tcp"
	],
	"src_byte_count": 217753000,
	"src_packets": 4521,
	}
	]
	}

Table 28: Netflow data sample

The format-adapter toolbox converts 600k network-traffic samples in 5.1 seconds. In fact, this operation introduces an overhead, however, it is tolerant to requirements and this operation performed only once. Furthermore, without this operation it will not be possible to share CTI with other sources, since the data CSV format is not standardized for CTI representation.

On the other hand, the total time required to forward the formatted dataset to the PrivacyNet system, enforcement of the anonymization operation, and generating the DPO is less than 25 seconds for the dataset of more than 600k network-traffic samples.

Another example of DMO execution is anonymization of emails. According to several studies, including Symantec[31] and Kaspersky Lab[32], adversaries increasingly used spam emails to distribute malware or for other malicious objectives. Furthermore, according to these reports, more than 50% of emails received by users is spam. Therefore analysis of email messages is essential in achieving overall security. However, since apart from the content and attached files email messages also include information both of recipients and sender, enforcing anonymization functions considers as a must for any further analysis.

Each email message of the considered spam email dataset contains unstructured data. This data comprises personal information, including email addresses of recipients, IP addresses, the email subject with its length, etc. Furthermore, other information that might be useful for performing analyses contains the email body, its length, and URLs in the text.

For this reason, the designed format-adapter toolbox can also convert email messages to STIX format making it possible to share and analyse those messages as well as their attachments. Furthermore, we used the extended version of the STIX proposed in (Fabio Martinelli 2018), which provides more description of the email message. Since email addresses of recipients can be treated as sensitive information, we consider security policy that requires the system to anonymize recipients email addresses. Therefore, once the format adapter toolbox converts spam emails to the STIX format, the DSA Adapter forwards the encoded dataset to the PrivacyNet together with the requested DMO. After anonymizing the recipients' email addresses, the PrivacyNet system returns the sanitized dataset to the DSA Adapter, which in turn invokes the DPOS API to save the generated DPO. In this way, data owners may define policies, which will allow data sharing only sanitized version of their datasets, thus preserving sensitive information from potential misuses.

In our particular case, more than 22k of email messages reported in STIX format were anonymized in 0.223 seconds. Nevertheless, additional time is required to convert email messages to STIX format, the format adapter toolbox has converted this particular dataset in less than 15 seconds.

[31] https://www.broadcom.com/support/security-center
[32] https://securelist.com/wannacry-ransomware-used-in-widespread-attacks-all-over-the-world/78351/

7 Conclusions

Secure information sharing among different parties is a prone task accompanied by multiple challenges. Considering the criticality of information produced and shared within the CI and CII, sensitive data protection is one of the main aspects of the overall security of those infrastructures.

Summarizing, this report presents a holistic point of view on the problem of sensitive data protection. In this document, the key technologies and methods for data protection are discussed. At first, the document covers the findings on the latest approaches and initiatives for sensitive data protection and storage. It addresses anonymization techniques and access control models, tackling at the same time the principle of privacy by design and by default regarding EDPB guidelines. Secondly, it provides an overview regarding the encryption technologies aiming to highlight the best approaches used for data security. Afterwards, our report provides an overview of the latest studies which use blockchain technology in the context of cyber-security domain. It also discusses smart contracts and describes the application scenario for achieving trust between organizations using blockchain.

Furthermore, this report also introduces a graphical modelling language that enables security experts to effectively detect and model security and privacy concerns in the early phases of incident handling. Doing so, it supports the analysis of cyber incident contexts, including threats, vulnerable assets and associated security risks, and risk treatments. Last but not least, in addition to the theoretical findings, the outcomes of this report add practical value by providing insight into secure and privacy-aware information sharing. In particular, Chapter 6 provides initial results for achieving secure and privacy-aware information sharing using both ShareNet and PrivacyNet systems of the CyberSANE framework.

7 List of Abbreviations

Abbreviation	Translation
ABAC	Attribute Based Access Control
ABE	Attribute Based Encryption
ACL	Access Control List
ACM	Access Control Mechanism
AES	Advanced Encryption Standard
AM	Attribute Manager
APTs	Advanced Persistent Threat
BFT	Byzantine Fault Tolerance
C3ISP	Collaborative and Confidential Information Sharing and Analysis for Cyber Protection
CA	Central Authority
CAAC	Context Aware Access Control
CCoA	Cyber Course of Action
CH	Context Handler
CI	Critical Infrastructure
CII	Critical Information Infrastructure
CIS	Centre for Internet Security
CP-ABE	Ciphertext Policy ABE
CSS	Common Security Services
CTI	Cyber Threat Intelligence
CVE	Common Vulnerabilities and Exposures
CVSS	Common Vulnerability Scoring System

DAC	Discretionary Access Control
DAG	Directed Acyclic Graph
DBFT	Delegated Byzantine Fault Tolerance
DDoS	Distributed Denial of Service
DES	Data Encryption Standard
DLT	Distributed Ledger Technology
DMO	Data Manipulation Operation
DNS	Domain Name System
DPbDD	Data Protection by Design and by Default
DPIAs	Data Protection Impact Assessments
DSA	Data Sharing Agreement
ECC	Elliptic Curve Cryptography
EDPB	European Data Protection Board
ENISA	European Union Agency for Cybersecurity
EPCIP	European Programme for Critical information infrastructure Protection
FACS	Fuzzy Asset Criticality System
FFX	Format-preserving, Feistel-based encryption X
FHE	Full Homomorphic Encryption
FPE	Format-Preserving Encryption
GDPR	General Data Protection Regulation
GSW	Gentry, Sahai and Waters
HE	Homomorphic Encryption
HiSea	Hybrid Cubes Encryption Algorithm
IAI	Information Analysis Infrastructure
IBE	Identity-Based Encryption
ICS	Industrial Control System

ICT	Information and Communications Technology
ID	Identifier
IDEA	International Data Encryption Algorithm
IDSs	Intrusion Detection Systems
IIN	Issuer Identification Number
IoCA	Impact on loss of Critical information infrastructure
IPFS	InterPlanetary File System
IPS	Intrusion Prevention System
ISI	Information Sharing Infrastructure
ISO	International Organization for Standardization
KP-ABE	Key Policy ABE
LEA	Lightweight Encryption Algorithm
LHR	Left-Half Recovery
LOC	Levels of Criticality
LWE	Learning With Errors
MAC	Mandatory Access Control
MD	Message Digest
MII	Major Industry Identifier
NIST	National Institute of Standards and Technology
NSA	National Security Agency
NVD	National Vulnerability Database
OWASP	Open Web Application Security Project
P2P	Peer-to-Peer
PAP	Policy Administration Point
PCI	Public Sector Information
PDP	Policy Decision Point

PEP	Policy Enforcement Point
PHE	Partially Homomorphic Encryption
PIP	Policy Information Point
PoA	Proof of Authority
PoET	Proof of Elapsed Time
PoI	Proof of Identity
PoL	Proof of Luck
PoS	Proof of Stake
PoW	Proof of Work
RAdAC	Risk-Adaptable Access Control
RBAC	Role Based Access Control
RC	Rivest Cipher
RHR	Right-Half Recover
RIPEMD	RIPE Message Digest
SCADA	Supervisory Control And Data Acquisition
SHA	Secure Hash Function
SIDE	Smartly Integrated Distributed Energy platform
SM	Session Manager
SME	Small and Medium Enterprise
SSL	Secure Sockets Layer
STIX	Structured Threat Information Expression
SWHE	Somewhat Homomorphic Encryption
TBAC	Task Based Access Control
TCP/IP	Transmission Control Protocol and the Internet Protocol
UCON	Usage Control
UCS	Usage Control System

UML	Unified Modelling Language
U-XACML	UCON-XACML
VDL	Virtual Data Lake
VPP	Virtual Power Plant
WEP	Wireless Equivalent Privacy
XACML	eXtensible Access Control Markup Language

8 Bibliography

Abadi, Martín. 2003. "Logic in access control." *In 18th Annual IEEE Symposium of Logic in Computer Science, Proceedings, IEEE*, 228-233.

Abhishta, A., R. van Rijswijk-Deij, and L.J. Nieuwenhui. 2019. "Measuring the impact of a successful DDoS attack on the customer behaviour of managed DNS service providers." *ACM SIGCOMM Computer Communication Review* 48 (5): 70-76.

Acar, A., H. Aksu, A.S. Uluagac, and M. Conti. 2018. "A Survey on Homomorphic Encryption Schemes: Theory and Implementation." *ACM Computing Surveys (CSUR)* (Association for Computing Machinery) 51 (4): 1-35.

Adrián Pérez-Resa, Miguel Garcia-Bosque, Carlos Sánchez-Azqueta, and Santiago Celma. 2020. "A new method for format preserving encryption in high-data rate communications." *IEEE Access 8, 21003-21016.*

—. 2018. "Using a chaotic cipher to encrypt Ethernet traffic." *In 2018 IEEE International Symposium on Circuits and Systems (ISCAS), IEEE*, 1-5.

Agrawal, R., J. Kiernan, R. Srikant, and Y. Xu. 2004. "Order Preserving Encryption for Numeric Data." *Proceedings of the 2004 ACM SIGMOD International Conference on Management of Data.* Paris: Association for Computing Machinery. 563-574.

Ahmed, E.Y., and M.D. Elkettani. 2016. "Fully homomorphic encryption: state of art and comparison." *International Journal of Computer Science and Information Security (IJCSIS)* 14 (4): 159-167.

Albrecht, M.R., P. Farshim, J.C. Faugere, and L. Perret. 2011. "Polly Cracker, Revisited." *Advances in Cryptology – ASIACRYPT 2011. ASIACRYPT 2011. Lecture Notes in Computer Science* 7073: 179–196.

Alessandro Baccarini, and Thaier Hayajneh. 2019. "Evolution of Format Preserving Encryption on IoT Devices: FF1+." *In Proceedings of the 52nd Hawaii International Conference on System Sciences.*

Alexandru, A.B., K. Gatsis, Y. Shoukry, S.A. Seshia, P. Tabuada, and G.J. Pappas. 2020. "Cloud-based Quadratic Optimization with Partially Homomorphic Encryption." *IEEE Transactions on Automatic Control.*

Ali, M., J. Nelson, R. Shea, and M.J. Freedman. 2016. "Blockstack: A global naming and storage system secured by blockchains." *2016 USENIX ANNUAL TECHNICAL CONFERENCE (USENIX ATC 16).* Denver: USENIX Association. 181-194.

Aliaksandr Lazouski, Gaetano Mancini, Fabio Martinelli, and Paolo Mori. 2012. "Usage control in cloud systems." *In 2012 International Conference for Internet Technology and Secured Transactions, IEEE*, 202-207.

Andoni, M., V. Robu, D. Flynn, S. Abram, D. Geach, D. Jenkins, P. McCallum, and A. Peacock. 2019. "Blockchain technology in the energy sector: A systematic review of challenges and opportunities." *Renewable and Sustainable Energy Reviews* 100: 143-174.

Andoni, M., V. Robu, D. Flynn, S. Abram, D. Geach, D. Jenkins, P. McCallum, and A. Peacock. 2019. "Blockchain technology in the energy sector: A systematic review of challenges and opportunities." *Renewable and Sustainable Energy Reviews* 100: 143-174.

Andrey Bogdanov, Dmitry Khovratovich, and Christian Rechberger. 2011. "Biclique cryptanalysis of the full AES." *In International conference on the theory and application of cryptology and information security, Springer, Berlin, Heidelberg*, 344-371.

Androulaki, E., A. Barger, V. Bortnikov, C. Cachin, K. Christidis, A. De Caro, D. Enyeart, et al. 2018. "Hyperledger Fabric: A Distributed Operating System for Permissioned Blockchains." *Proceedings of the Thirteenth EuroSys Conference (EuroSys '18).* Porto: Association for Computing Machinery. 1-15.

Aranjo, S., T. Adivarekar, and D. Hegde. 2019. "Blockchain Name Service." *Pramana Research Journal* 9 (2).

Armknecht, F., G.O. Karame, and A. Mandal. 2015. "Ripple: Overview and Outlook." *Trust and Trustworthy Computing. Trust 2015. Lecture Notes in Computer Science* 9229: 163-180.

Attrapadung, N., and H. Imai. 2009. "Dual-Policy Attribute Based Encryption." *Applied Cryptography and Network Security. ACNS 2009. Lecture Notes in Computer Science* 5536: 168-185.

Attrapadung, N., and S. Yamada. 2015. "Duality in ABE: Converting Attribute Based Encryption for Dual Predicate and Dual Policy via Computational Encodings." *Topics in Cryptology – - CT-RSA 2015. CT-RSA 2015. Lecture Notes in Computer Science* 9048: 87-105.

Attrapadung, N., B. Libert, and E. de Panafieu. 2011. "Expressive Key-Policy Attribute-Based Encryption with Constant-Size Ciphertexts." *Public Key Cryptography – PKC 2011. PKC 2011. Lecture Notes in Computer Science* 6571: 90-108.

Azaria, A., A. Ekblaw, T. Vieira, and A. Lippman. 2016. "MedRec: Using Blockchain for Medical Data Access and Permission Management." *2016 2nd International Conference on Open and Big Data (OBD).* Vienna: IEEE. 25-30.

Azouvi, S., M. Al-Bassam, and S. Meiklejohn. 2017. "Who Am I? Secure Identity Registration on Distributed Ledgers." *Data Privacy Management, Cryptocurrencies and Blockchain Technology. DPM 2017, CBT 2017. Lecture Notes in Computer Science* 10436: 373-389.

Bach, L.M., B. Mihaljevic, and M. Zagar. 2018. "Comparative analysis of blockchain consensus algorithms." *2018 41st International Convention on Information and Communication Technology, Electronics and Microelectronics (MIPRO).* Opatija: IEEE.

Baker, T., M. Asim, Á. MacDermott, F. Iqbal, F. Kamoun, B. Shah, O. Alfandi, and M. Hammoudeh. 2020. "A secure fog-based platform for SCADA-based IoT critical infrastructure." *Software: Practice and Experience* 50 (5): 503-518.

Baldwin, Robert W. 1990. "Naming and Grouping Privileges to Simplify Security Management in Large Databases." *In IEEE Symposium on Security and Privacy*, 116-132.

Barinov, P.K.I., and V. Baranov. 2018. "POA Network Whitepaper." 9 28. Accessed 8 7, 2020. https://github.com/poanetwork/wiki/wiki/POA-Network-Whitepaper.

Basel Katt, Xinwen Zhang, Ruth Breu, Michael Hafner, and Jean-Pierre Seifert. 2008. "A general obligation model and continuity: enhanced policy enforcement engine for usage control." *In Proceedings of the 13th ACM symposium on Access control models and technologies*, 123-132.

Baum, C., I. Damgård, and C. Orlandi. 2014. "Publicly Auditable Secure Multi-Party Computation." *Security and Cryptography for Networks. SCN 2014. Lecture Notes in Computer Science* 8642: 175-196.

Ben Morris, Phillip Rogaway, and Till Stegers. 2009. "How to encipher messages on a small domain." *In Annual International Cryptology Conference, Springer, Berlin, Heidelberg*, 286-302.

Benet, J. 2014. "IPFS - Content Addressed, Versioned, P2P File System." *arXiv preprint arXiv:1407.3561.*

Bethencourt, J., A. Sahai, and B. Waters. 2007. "Ciphertext-Policy Attribute-Based Encryption." *2007 IEEE Symposium on Security and Privacy (SP '07).* Berkeley: IEEE. 321-334.

Bettín-Díaz, R., A.E. Rojas, and C. Mejía-Moncayo. 2018. "Methodological Approach to the Definition of a Blockchain System for the Food Industry Supply Chain Traceability." *Computational Science and Its Applications – ICCSA 2018. ICCSA 2018. Lecture Notes in Computer Science.* Melbourne: Springer. 19-33.

Bitcoin Wiki. 2020. *DPoS.* Accessed 8 7, 2020. https://en.bitcoinwiki.org/wiki/DPoS.

—. 2019. *Proof of Stake.* Accessed 8 7, 2020. https://en.bitcoin.it/wiki/Proof_of_Stake.

Boneh, D., and M. Franklin. 2001. "Identity-Based Encryption from the Weil Pairing." *Advances in Cryptology — CRYPTO 2001. CRYPTO 2001. Lecture Notes in Computer Science* 2139: 213-229.

Boneh, D., and X. Boyen. 2008. "Short Signatures Without Random Oracles and the SDH Assumption in Bilinear Groups." *Journal of Cryptology* 21 (2): 149-177.

Boneh, D., E.J. Goh, and K. Nissim. 2005. "Evaluating 2-DNF Formulas on Ciphertexts." *Theory of Cryptography. TCC 2005. Lecture Notes in Computer Science* 3378: 325-341.

Boneh, D., X. Boyen, and E.J. Goh. 2005. "Hierarchical Identity Based Encryption with Constant Size Ciphertext." *Advances in Cryptology – EUROCRYPT 2005. EUROCRYPT 2005. Lecture Notes in Computer Science* 3494: 440-456.

Boos, P., and M. Lacoste. 2020. "Networks of Trusted Execution Environments for Data Protection in Cooperative Vehicular Systems." *Vehicular Ad-hoc Networks for Smart Cities. Advances in Intelligent Systems and Computing* 1144: 99-109.

Božović, V., D. Socek, R. Steinwandt, and V.I. Villány. 2012. "Multi-authority attribute-based encryption with honest-but-curious central authority." *International Journal of Computer Mathematics* 89 (3): 268-283.

Brakerski, Z., and V. Vaikuntanathan. 2014. "Efficient Fully Homomorphic Encryption from (Standard) LWE." *SIAM Journal on Computing* (Association for Computing Machinery) 43 (2): 831-871.

Brucker, A.D., H. Petritsch, and S.G. Weber. 2010. "Attribute-Based Encryption with Break-Glass." *Information Security Theory and Practices. Security and Privacy of Pervasive Systems and Smart Devices. WISTP 2010. Lecture Notes in Computer Science* 6033: 237-244.

Cachin, C. 2016. "Architecture of the Hyperledger Blockchain Fabric." *Workshop on distributed cryptocurrencies and consensus ledgers* 310 (4).

Castro, M., and B. Liskov. 1999. "Practical Byzantine fault tolerance." *Proceedings of the Third Symposium on Operating Systems Design and Implementation (OSDI '99).* New Orleans: USENIX Association.

Changhyun Lee, Yeonju Choi, Hyeongmin Park, Kangbin Yim, and Sun-Young Lee. 2019. "Novel Encryption Method of GPS Information in Image File Using Format-Preserving

Encryption." *In International Conference on Innovative Mobile and Internet Services in Ubiquitous Computing, Springer, Cham*, 815-823.

Chase, M. 2007. "Multi-authority Attribute Based Encryption." *Theory of Cryptography. TCC 2007. Lecture Notes in Computer Science* 4392: 515-534.

Chase, M., and S.S. Chow. 2009. "Improving Privacy and Security in Multi-Authority Attribute-Based Encryption." *Proceedings of the 16th ACM Conference on Computer and Communications Security.* Chicago: Association for Computing Machinery. 121–130.

Chen, L., L. Xu, N. Shah, Z. Gao, Y. Lu, and W. Shi. 2017. "On Security Analysis of Proof-of-Elapsed-Time (PoET)." *Stabilization, Safety, and Security of Distributed Systems. SSS 2017. Lecture Notes in Computer Science* 10616: 282-297.

Chillotti, I., N. Gama, M. Georgieva, and M. Izabachène. 2020. "TFHE: Fast Fully Homomorphic Encryption Over the Torus." *Journal of Cryptology* 3 (1): 34-91.

Christian Schläger, Manuel Sojer, Björn Muschall, and Günther Pernul. 2006. "Attribute-based authentication and authorisation infrastructures for e-commerce providers." *In International Conference on Electronic Commerce and Web Technologies, Springer, Berlin, Heidelberg*, 132-141.

Cocks, C. 2001. "An Identity Based Encryption Scheme Based on Quadratic Residues." *Cryptography and Coding. Cryptography and Coding 2001. Lecture Notes in Computer Science* 2260: 360-363.

D. Richard Kuhn, Edward J. Coyne and Timothy R. Weil. 2010. "Adding attributes to role-based access control." *Computer 43, no. 6* 79-81.

David Ferraiolo, Ravi Sandhu, Serban Gavrila, D. Richard Kuhn, and Ramaswamy Chandramouli. 2001. "Proposed NIST standard for role-based access control." *ACM Transactions on Information and System Security (TISSEC) 4, no. 3*, 224-274.

De Angelis, S., L. Aniello, R. Baldoni, F. Lombardi, A. Margheri, and V. Sassone. 2018. "PBFT vs proof-of-authority: applying the CAP theorem to permissioned blockchain." *Second Italian Conference On Cybersecurity (ITA-SEC 2018).* Milan: University of Southampton Institutional Repository.

Deshpande, A., K. Stewart, L. Lepetit, and S. Gunashekar. 2017. *Distributed Ledger Technologies/Blockchain: Challenges, opportunities and the prospects for standards.* The British Standards Institution (BSI).

Deukjo Hong, Jung-Keun Lee, Dong-Chan Kim, Daesung Kwon, Kwon Ho Ryu, and Dong-Geon Lee. 2013. "LEA: A 128-bit block cipher for fast encryption on common processors." *In International Workshop on Information Security Applications, Springer, Cham*, 3-27.

Dhillon, V., D. Metcalf, and M. Hooper. 2017. "The Hyperledger Project." In *Blockchain Enabled Applications*, 139-149. Apress, Berkeley, CA.

Dhillon, V., D. Metcalf, and M. Hooper. 2017. "The Hyperledger Project." *Blockchain Enabled Applications* 139-149.

Dusse, Ronald Rivest and S. 1991. ""The MD5 message-digest algorithm."." 330-344.

Dworkin, Morris. 2016. "Recommendation for block cipher modes of operation: methods for format-preserving encryption." *NIST Special Publication 800 (2016): 38G.*

ElGamal, Taher. 1985. ""A subexponential-time algorithm for computing discrete logarithms over GF (p^ 2)." *IEEE transactions on information theory 31, no. 4*, 473-481.

Eli Biham, Orr Dunkelman, Nathan Keller, and Adi Shamir. 2015. "New attacks on IDEA with at least 6 rounds." *Journal of Cryptology 28, no. 2* 209-239.

Enrico Carniani, Davide D'Arenzo, Aliaksandr Lazouski, Fabio Martinelli, and Paolo Mori. 2016. "Usage control on cloud systems." *Future Generation Computer Systems 63*, 37-55.

Eric Brier, Thomas Peyrin, and Jacques Stern. 2010. "BPS: a format-preserving encryption proposal."

European Commission CORDIS. 2016. *Collaborative and Confidential Information Sharing and Analysis for Cyber Protection.* Accessed 6 6, 2020. https://cordis.europa.eu/project/id/700294.

Evered, Mark. 2003. "Supporting parameterised roles with object-based access control." *In 36th Annual Hawaii International Conference on System Sciences, Proceedings of the IEEE*, 9.

F. Betül Durak, and Serge Vaudenay. 2017. "Breaking the FF3 format-preserving encryption standard over small domains." *In Annual international cryptology conference, Springer, Cham*, 679-707.

Farroha, Bassam Farroha and Deborah. 2012. "Challenges of "operationalizing" dynamic system access control: Transitioning from ABAC to RAdAC." *In 2012 IEEE International Systems Conference SysCon 2012, IEEE*, 1-7.

Fellows, M., and N. Koblitz. 1994. "Combinatorial cryptosystems galore!" *Contemporary Mathematics* 168: 51-51.

FERC. 2016. *Revised Critical Infrastructure Protection Reliability Standards.* Washington, DC: Federal Energy Regulatory Commission (FERC).

Ferrag, M.A., and L. Maglaras. 2019. "DeepCoin: A Novel Deep Learning and Blockchain-Based Energy Exchange Framework for Smart Grids." *IEEE Transactions on Engineering Management* 1-13.

Franco, P. 2014. *Understanding Bitcoin: Cryptography, engineering and economics.* John Wiley & Sons.

Gao, W., W. Yu, F. Liang, W.G. Hatcher, and C. Lu. 2018. "Privacy-Preserving Auction for Big Data Trading Using Homomorphic Encryption." *EEE Transactions on Network Science and Engineering* 7 (2): 776-791.

Gentry, C. 2009. "A Fully Homomorphic Encryption Scheme." Ph.D. Dissertation. Stanford University.

Gentry, C., A. Sahai, and B. Waters. 2013. "Homomorphic Encryption from Learning with Errors: Conceptually-Simpler, Asymptotically-Faster, Attribute-Based." Edited by R. Canetti and J.A. Garay. *Advances in Cryptology – CRYPTO 2013. CRYPTO 2013. Lecture Notes in Computer Science* (Springer) 8042: 75-92.

Gentry, C., and A. Silverberg. 2002. "Hierarchical ID-Based Cryptography." *Advances in Cryptology — ASIACRYPT 2002. ASIACRYPT 2002. Lecture Notes in Computer Science* 2501: 548-566.

Gentry, C., S. Halevi, and V. Vaikuntanathan. 2010. "A Simple BGN-Type Cryptosystem from LWE." *Advances in Cryptology – EUROCRYPT 2010. EUROCRYPT 2010. Lecture Notes in Computer Science* 6110: 506-522.

Gjøsteen, K., and M. Strand. 2016. "Fully homomorphic encryption must be fat or ugly?" *IACR Cryptol. ePrint Arch.* 105.

Goyal, V., A. Jain, O. Pandey, and A. Sahai. 2008. "Bounded Ciphertext Policy Attribute Based Encryption." *Automata, Languages and Programming. ICALP 2008. Lecture Notes in Computer Science* 5126: 579-591.

Goyal, V., O. Pandey, A. Sahai, and B. Waters. 2006. "Attribute-Based Encryption for Fine-Grained Access Control of Encrypted Data." *Proceedings of the 13th ACM Conference on Computer and Communications Security.* Alexandria: Association for Computing Machinery. 89–98.

Guangsen Zhang, and Manish Parashar. 2004. "Context-aware dynamic access control for pervasive applications." *In Proceedings of the Communication Networks and Distributed Systems Modeling and Simulation Conference*, 21-30.

Gupta, R. 2018. *Hands-On Cybersecurity with Blockchain: Implement DDoS protection, PKI-based identity, 2FA, and DNS security using Blockchain.* Packt Publishing Ltd.

He, X., M.O. Pun, and C.C.J. Kuo. 2012. "Secure and efficient cryptosystem for smart grid using homomorphic encryption." *2012 IEEE PES Innovative Smart Grid Technologies (ISGT).* Washington: IEEE. 1-8.

He, Y., H. Li, X. Cheng, Y. Liu, C. Yang, and L. Sun. 2018. "A Blockchain Based Truthful Incentive Mechanism for Distributed P2P Applications." *IEEE Access* 6: 27324-27335.

Herranz, J. 2017. "Attribute-based encryption implies identity-based encryption." *IET Information Security* 11 (6): 332-337.

Hoover, Douglas Neil. 2015. " Format-preserving encryption via rotating block encryption." *U.S. Patent 8,948,376*, February 3.

Horwitz, J., and B. Lynn. 2002. "Toward Hierarchical Identity-Based Encryption." *Advances in Cryptology — EUROCRYPT 2002. EUROCRYPT 2002. Lecture Notes in Computer Science* 2332: 466-481.

Hu, C., Y. Huo, L. Ma, H. Liu, S. Deng, and L. Feng. 2017. "An Attribute-Based Secure and Scalable Scheme for Data Communications in Smart Grids." *Wireless Algorithms, Systems, and Applications. WASA 2017. Lecture Notes in Computer Science* 10251: 469-482.

Huang, J., K. Lei, M. Du, H. Zhao, H. Liu, J. Liu, and Z. Qi. 2019. "Survey on Blockchain Incentive Mechanism." *Data Science. ICPCSEE 2019. Communications in Computer and Information Science* 1058: 386-395.

Hur, J. 2013. "Attribute-Based Secure Data Sharing with Hidden Policies in Smart Grid." *IEEE Transactions on Parallel and Distributed Systems* 24 (11): 2171-2180.

Iglio, Luigi Giuri and Pietro. 1997. "Role templates for content-based access control." *In Proceedings of the second ACM workshop on Role-based access control, pp. 153-159. 1997.*, 153-159.

Insu Oh, Taeeun Kim, Kangbin Yim, and Sun-Young Lee. 2019. "A novel message-preserving scheme with format-preserving encryption for connected cars in multi-access edge computing." *Sensors 19, no. 18 (2019): 3869.*

Isabel F. Cruz, Rigel Gjomemo, Benjamin Lin, and Mirko Orsini. 2008. "A constraint and attribute based security framework for dynamic role assignment in collaborative environments." *In International Conference on Collaborative Computing: Networking, Applications and Worksharing, Springer, Berlin, Heidelberg*, 322-339.

Ishai, Y., and A. Paskin. 2007. "Evaluating Branching Programs on Encrypted Data." *Theory of Cryptography. TCC 2007. Lecture Notes in Computer Science* 4392: 575-594.

Jeffrey Fischer, Daniel Marino, Rupak Majumdar, and Todd Millstein. 2009. " Fine-grained access control with object-sensitive roles." *In European Conference on Object-Oriented Programming, Springer, Berlin, Heidelberg*, 173-194.

John Black, and Phillip Rogaway. 2002. "Ciphers with arbitrary finite domains." *In Cryptographers' track at the RSA conference, Springer, Berlin, Heidelberg*, 114-130.

Joux, A. 2000. "A One Round Protocol for Tripartite Diffie–Hellman." *Algorithmic Number Theory. ANTS 2000. Lecture Notes in Computer Science* 1838: 385-393.

Khambhammettu, Jason Crampton and Hemanth. 2008. "Delegation in role-based access control." *International Journal of Information Security 7, no. 2* 123-136.

Khayat, Ali E. Abdallah and Etienne J. 2004. "A formal model for parameterized role-based access control." *In IFIP World Computer Congress, TC 1, Springer, Boston, MA*, 233-246.

Khovratovich, Alex Biryukov and Dmitry. 2009. "Related-key cryptanalysis of the full AES-192 and AES-256." *In International Conference on the Theory and Application of Cryptology and Information Security, Springer, Berlin, Heidelberg*, 1-18.

Kumar J., P.V., and R.K. Aluvalu. 2015. "Key Policy Attribute Based Encryption (KP-ABE): A Review." *International Journal of Innovative and Emerging Research in Engineering (IJIERE)* 2 (2): 49-52.

Lampson, Butler W. 1974. "Protection." *ACM SIGOPS Operating Systems Review 8, no. 1*, 18-24.

Lars R. Knudsen, Vincent Rijmen, Ronald L. Rivest, and Matthew JB Robshaw. 1998. "On the design and security of RC2." *In International Workshop on Fast Software Encryption, Springer, Berlin, Heidelberg*, 206-221.

Lei, A., H. Cruickshank, Y. Cao, P. Asuquo, C.P.A. Ogah, and Z. Sun. 2017. "Blockchain-Based Dynamic Key Management for Heterogeneous Intelligent Transportation Systems." *IEEE Internet of Things Journal* 4 (6): 1832-1843.

Levy-dit-Vehel, F., and L. Perret. 2004. "A Polly Cracker System Based on Satisfiability." *Coding, Cryptography and Combinatorics. Progress in Computer Science and Applied Logic* 23: 177-192.

Lewko, A., A. Sahai, and B. Waters. 2010. "Revocation Systems with Very Small Private Keys." *2010 IEEE Symposium on Security and Privacy.* Berkeley/Oakland: IEEE. 273-285.

Li, J., and L. Wang. 2015. "Noise-free Symmetric Fully Homomorphic Encryption based on noncommutative rings." *IACR Cryptol. ePrint Arch.* 641.

Li, J., Y. Shi, and Y. Zhang. 2017. "Searchable ciphertext-policy attribute-based encryption with revocation in cloud storage." *International Journal of Communication Systems* 30 (1): e2942.

Li, Jingwei, Zheli Liu, Li Xu, and Chunfu Jia. 2012. "An efficient format-preserving encryption mode for practical domains." *Wuhan University Journal of Natural Sciences 17, no. 5* 428-434.

Liang, G., S.R. Weller, F. Luo, J. Zhao, and Z.Y. Dong. 2019. "Distributed Blockchain-Based Data Protection Framework for Modern Power Systems Against Cyber Attacks." *IEEE Transactions on Smart Grid* 10 (3): 3162-3173.

Lingyu Wang, Duminda Wijesekera, and Sushil Jajodia. 2004. "A logic-based framework for attribute based access control." *In Proceedings of the 2004 ACM workshop on Formal methods in security engineering*, 45-55.

Liu, B., X.L. Yu, S. Chen, X. Xu, and L. Zhu. 2017. "Blockchain Based Data Integrity Service Framework for IoT Data." *2017 IEEE International Conference on Web Services (ICWS).* Honolulu: IEEE.

Liu, Y., P. Ning, and M.K. Reiter. 2011. "False data injection attacks against state estimation in electric power grids." *ACM Transactions on Information and System Security (TISSEC)* 14 (1): 1-33.

Liu, Z., and D.S. Wong. 2016. "Practical Attribute-Based Encryption: Traitor Tracing, Revocation and Large Universe." *The Computer Journal* 59 (7): 983-1004.

Liu, Z., Z.L. Jiang, X. Wang, and S.M. Yiu. 2018. "Practical Attribute-Based Encryption." *Journal of Network and Computer Applications* 108: 112–123.

Lockman, Naftaly H. Minsky and Abe D. 1985. "Ensuring integrity by adding obligations to privileges." *In Proceedings of the 8th international conference on Software engineering*, 92-102.

Lohmer, J., N. Bugert, and R. Lasch. 2020. "Analysis of resilience strategies and ripple effect in blockchain-coordinated supply chains: An agent-based simulation study." *International Journal of Production Economics* 228: 107882.

Loomis, Alan O'Connor and Ross. 2010. "Economic analysis of role-based access control." *No. RTI Project Number 0211876. RTI International.*

Lu, H., K. Huang, M. Azimi, and L. Guo. 2019. "Blockchain Technology in the Oil and Gas Industry: A Review of Applications, Opportunities, Challenges, and Risks." *IEEE Access* 7: 41426-41444.

Maanak Gupta, Farhan Patwa, and Ravi Sandhu. 2018. "An attribute-based access control model for secure big data processing in Hadoop ecosystem." *In Proceedings of the Third ACM Workshop on Attribute-Based Access Control*, 13-24.

MacDermott, Á., Q. Shi, M. Merabti, and K. Kifayat. 2015. "Hosting critical infrastructure services in the cloud environment considerations." *International Journal of Critical Infrastructures* 11 (4): 365-381.

Machado, C., and A.A.M. Fröhlich. 2018. "IoT Data Integrity Verification for Cyber-Physical Systems Using Blockchain." *2018 IEEE 21st International Symposium on Real-Time Distributed Computing (ISORC).* Singapore: IEEE.

Maurizio Colombo, Aliaksandr Lazouski, Fabio Martinelli, and Paolo Mori. 2010. "A proposal on enhancing XACML with continuous usage control features." *In Grids, P2P and Services Computing, Springer, Boston, MA*, 133-146.

McDaniel, P., and S. McLaughlin. 2009. "Security and Privacy Challenges in the Smart Grid." *IEEE Security & Privacy* 7 (3): 75-77.

Melo, C., J. Dantas, D. Oliveira, I. Fé, R. Matos, R. Dantas, R. Maciel, and P. Maciel. 2018. "Dependability Evaluation of a Blockchain-as-a-Service Environment." *2018 IEEE Symposium on Computers and Communications (ISCC).* Natal: IEEE. 00909-00914.

Melo, C., J. Dantas, R. Maciel, P. Silva, and P. Maciel. 2019. "Models to evaluate service provisioning over cloud computing environments-A blockchain-as-A-service case study." *Revista de Informática Teórica e Aplicada* 26 (3): 65-74.

Mendi, A., T. Erol, E. Safak, and T. Kaym. 2019. "A Blockchain Smart ContractApplication Framework." *2019 International Symposium on Networks, Computers and Communications (ISNCC).* Istanbul: IEEE.

Mihir Bellare, Phillip Rogaway, and Terence Spies. 2010. "The FFX mode of operation for format-preserving encryption." *NIST submission 20 (2010): 19.*

—. 2010. "The FFX mode of operation for format-preserving encryption." *NIST submission 20 (2010): 19.*

—. 2010. "The FFX mode of operation for format-preserving encryption." *NIST submission 20 (2010): 19.*

Mihir Bellare, Thomas Ristenpart, Phillip Rogaway, and Till Stegers. 2009. "Format-preserving encryption." *In International workshop on selected areas in cryptography, Springer, Berlin, Heidelberg*, 295-312.

Mihir Bellare, Viet Tung Hoang, and Stefano Tessaro. 2016. "Message-recovery attacks on Feistel-based format preserving encryption." *In Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security*, 444-455.

Milutinovic, M., W. He, H. Wu, and M. Kanwal. 2016. "Proof of Luck: An Efficient Blockchain Consensus Protocol." *Proceedings of the 1st Workshop on System Software for Trusted Execution (SysTEX '16).* Trento: Association for Computing Machinery. 1-6.

Mohanad Sarhan, Siamak Layeghy, Nour Moustafa, and Marius Portmann. 2020. "NetFlow Datasets for Machine Learning-based Network Intrusion Detection Systems."

Mor Weiss, Boris Rozenberg, and Muhammad Barham. n.d. "Practical solutions for format-preserving encryption." *arXiv preprint arXiv:1506.04113*, 2015.

Morris J. Dworkin, Elaine B. Barker, James R. Nechvatal, James Foti, Lawrence E. Bassham, E. Roback, James F. Dray Jr. 2001. *Advanced Encryption Standard (AES).* NIST Pubs, Federal Inf. Process. Stds. (NIST FIPS) - 197.

Moubarak, J., E. Filiol, and M. Chamoun. 2018. "On blockchain security and relevant attacks." *2018 IEEE Middle East and North Africa Communications Conference (MENACOMM).* Jounieh: IEEE. 1-6.

Mouha, Elaine Barker and Nicky. 2017. *Recommendation for the triple data encryption algorithm (TDEA) block cipher.* NIST Special Publication (SP) 800-67 Rev. 2 (Draft), National Institute of Standards and Technology, . No. NIST Special Publication (SP) 800-67 Rev. 2 (Draft). National Institute of Standards and Technology, 2017.

Mukherjee, M., R. Matam, L. Shu, L. Maglaras, M.A. Ferrag, N. Choudhury, and V. Kumar. 2017. "Security and Privacy in Fog Computing: Challenges." *IEEE Access* 5: 19293-19304.

Muller, S., S. Katzenbeisser, and C. Eckert. 2009. "On multi-authority ciphertext-policy attribute-based encryption." *Bulletin of the Korean Mathematical Society* 46 (4): 803-819.

Murthy, S., and C.R. Kavitha. 2019. "Preserving Data Privacy in Cloud using Homomorphic Encryption." *2019 3rd International conference on Electronics, Communication and Aerospace Technology (ICECA).* Coimbatore: IEEE. 1131–1135.

Mushtaq, Muhammad Faheem, Sapiee Jamel, Abdulkadir Hassan Disina, Zahraddeen A. Pindar, N. Shafinaz Ahmad Shakir, and Mustafa Mat Deris. 2017. ""A survey on the cryptographic encryption algorithms"." *International Journal of Advanced Computer Science and Applications 8, no. 11* 333-344.

Mustafa, I., H. Mustafa, A.T. Azar, S. Aslam, S.M. Mohsin, M.B. Qureshi, and N. Ashraf. 2020. "Noise Free Fully Homomorphic Encryption Scheme Over Non-Associative Algebra." *IEEE Access* 8: 136524-136536.

Mylrea, M., and S.N.G. Gourisetti. 2018. "Blockchain: Next Generation Supply Chain Security for Energy Infrastructure and NERC Critical Infrastructure Protection (CIP) Compliance." *Journal of Systemics, Cybernetics and Informatics (JSCI)* 16 (6): 22-30.

Naehrig, M., K. Lauter, and V. Vaikuntanathan. 2011. "Can Homomorphic Encryption Be Practical?" *Proceedings of the 3rd ACM Workshop on Cloud Computing Security Workshop.* Chicago: Association for Computing Machinery. 113-124.

Nakamoto, S. 2019. *Bitcoin: A peer-to-peer electronic cash system.* Manubot.

Nash, David FC Brewer and Michael J. 1989. "The Chinese Wall Security Policy." *In IEEE symposium on security and privacy, vol. 1989*, 206.

NEO Team. 2014. *The dBFT Algorithm - NEO Documentation.* Accessed 7 8, 2020. https://docs.neo.org/developerguide/en/articles/consensus/consensus_algorithm.html.

Nguyen, C.T., D.T. Hoang, D.N. Nguyen, D. Niyato, H.T. Nguyen, and E. Dutkiewicz. 2019. "Proof-of-Stake Consensus Mechanisms for Future Blockchain Networks: Fundamentals, Applications and Opportunities." *IEEE Access* 7: 85727-85745.

Nguyen, G.T., and K. Kim. 2018. "A Survey about Consensus Algorithms Used in Blockchain." *Journal of Information processing systems* 14 (1): 101-128.

Niels Ferguson, Stefan Lucks, Bruce Schneier, Doug Whiting, Mihir Bellare, Tadayoshi Kohno, Jon Callas, and Jesse Walker. 2010. "The Skein hash function family." *Submission to NIST (round 3) 7, no. 7.5.*

Olson, K., M. Bowman, J. Mitchell, S. Amundson, D. Middleton, and C. Montgomery. 2018. "Sawtooth: An Introduction." 1. Accessed 8 18, 2020. https://www.hyperledger.org/wp-content/uploads/2018/01/Hyperledger_Sawtooth_WhitePaper.pdf.

Ostrovsky, R., A. Sahai, and B. Waters. 2007. "Attribute-Based Encryption with Non-Monotonic Access Structures." *Proceedings of the 14th ACM Conference on Computer and Communications Security.* Alexandria: Association for Computing Machinery.

Ouaddah, A., A. Abou Elkalam, and A.A. Ouahman. 2017. "Towards a Novel Privacy-Preserving Access Control Model Based on Blockchain Technology in IoT." *Europe and MENA Cooperation Advances in Information and Communication Technologies. Advances in Intelligent Systems and Computing* 520: 523-533.

Paillier, P. 1999. "Public-Key Cryptosystems Based on Composite Degree Residuosity Classes." Edited by J. Stern. *Advances in Cryptology — EUROCRYPT '99. EUROCRYPT 1999. Lecture Notes in Computer Science* (Springer) 1592: 223-238.

Pan, M., X. Zhu, and Y. Fang. 2012. "Using homomorphic encryption to secure the combinatorial spectrum auction without the trustworthy auctioneer." *Wireless Networks* 18 (2): 113-128.

Pang, L., J. Yang, and Z. Jiang. 2014. "A Survey of Research Progress and Development Tendency of Attribute-Based Encryption." *The Scientific World Journal* 2014.

papers, Ketu File white. 2003-2004. *Symmetric vs Asymmetric Encryption.* a division of Midwest Research Corporation.

Peng, Z. 2019. *Danger of using fully homomorphic encryption: A look at Microsoft SEAL.* arXiv preprint arXiv:1906.07127.

Pervez, H., M. Muneeb, M.U. Irfan, and I.U. Haq. 2018. "A Comparative Analysis of DAG-Based Blockchain Architectures." *2018 12th International Conference on Open Source Systems and Technologies (ICOSST).* Lahore: IEEE. 27-34.

Pierangela Samarati, and Sabrina Capitani de Vimercati. 2000. "Access control: Policies, models, and mechanisms." *In International School on Foundations of Security Analysis and Design, Springer, Berlin, Heidelberg*, 137-196.

Raikwar, M., S. Mazumdar, S. Ruj, S.S. Gupta, A. Chattopadhyay, and K.Y. Lam. 2018. "A Blockchain Framework for Insurance Processes." *2018 9th IFIP International Conference on New Technologies, Mobility and Security (NTMS).* Paris: IEEE. 1-4.

Ravi Sandhu, Edward J. Coyne, Hal L. Feinstein, and Charles E. Youman. 1996. "Role-based access control models." *Computer 29, no. 2* 38-47.

Ray Beaulieu, Douglas Shors, Jason Smith, Stefan Treatman-Clark, Bryan Weeks, and Louis Wingers. 2015. "The SIMON and SPECK lightweight block ciphers." *In Proceedings of the 52nd Annual Design Automation Conference*, 1-6.

Richard Agbeyibor, Jonathan Butts, Michael Grimaila, and Robert Mills. 2014. "Evaluation of format-preserving encryption algorithms for critical infrastructure protection." *In International Conference on Critical Infrastructure Protection, Springer, Berlin, Heidelberg*, 245-261. .

Rissanen, Erik. n.d. *Oasis extensible access control markup language (xacml) version 3.0.* OASIS committee specification 1.

Rivest, R. L. 1991. "The MD4 message digest algorithm"." 303-311.

Rivest, R.L., L. Adleman, and M.L. Dertouzos. 1978. "On data banks and privacy homomorphisms." *Foundations of secure computation* 4 (11): 169-180.

Rivest, Ronald L. 1994. " The RC5 encryption algorithm." *In International Workshop on Fast Software Encryption, Springer, Berlin, Heidelberg*, 86-96.

Rogaway, Mihir Bellare and Phillip. 1999. "On the construction of variable-input-length ciphers." *In International Workshop on Fast Software Encryption, Springer, Berlin, Heidelberg*, 231-244.

Rogaway, Phillip. 2010. "A synopsis of format-preserving encryption." *In Unpublished Manuscript.*

Ronald L. Rivest, Adi Shamir, and Leonard Adleman. 1978. "A method for obtaining digital signatures and public-key cryptosystems." *Communications of the ACM 21, no. 2*, 120-126.

Ronald L. Rivest, Matthew JB Robshaw, Ray Sidney, and Yiqun L. Yin. 1998. "The RC6TM block cipher." *In First Advanced Encryption Standard (AES) Conference*, 16.

Şafak, E., A. Furkan, and T. Erol. 2019. "Hybrid Database Design Combination of Blockchain And Central Database." *2019 3rd International Symposium on Multidisciplinary Studies and Innovative Technologies (ISMSIT).* Ankara: IEEE.

Sahai, A., and B. Waters. 2005. *Fuzzy Identity-Based Encryption.* Vol. 3494, in *Advances in Cryptology - EUROCRYPT 2005. Lecture Notes in Computer Science*, edited by R. Cramer, 457-473. Springer.

Salman, T., M. Zolanvari, A. Erbad, R. Jain, and M. Samaka. 2018. "Security Services Using Blockchains: A State of the Art Survey." *IEEE Communications Surveys & Tutorials* 21 (1): 858-880.

Samarati, Ravi Sandhu and Pierangela. 1994. "Access control: principle and practice." *IEEE communications magazine 32, no. 9*, 40-48.

Sander, T. , A. Young, and M. Yung. 1999. "Non-interactive cryptocomputing for NC1." *40th Annual Symposium on Foundations of Computer Science (Cat. No.99CB37039).* New York City: IEEE. 554-566.

Sandhu, Jaehong Park and Ravi. 2004. "The UCONABC usage control model." *"The UCONABC usage control model" ACM Transactions on Information and System Security (TISSEC) 7, no. 1 (2004): 128-174.*, 128-174.

—. 2004. "The UCONABC usage control model." 128-174.

Sandhu, Maanak Gupta and Ravi. 2016. "The GURA G Administrative Model for User and Group Attribute Assignment." *In International Conference on Network and System Security, Springer, Cham*, 318-332.

Sandhu, Ravi. 1988. "Transaction control expressions for separation of duties." *In Proc. of the Fourth Computer Security Applications Conference*, 282-286.

Sandhu, Ravi. 1993. "Lattice-based access control models." *Computer 26, no. 11* 9-19.

Sandhu, Roshan K. Thomas and Ravi. 1998. "Task-based authorization controls (TBAC): A family of models for active and enterprise-oriented authorization management." *In Database security XI, Springer, Boston, MA*, 166-181.

Sapiee Jamel, Mustafa Mat Deris, Iwan Tri Riyadi Yanto, and Tutut Herawan. 2011. "The hybrid cubes encryption algorithm (HiSea)." *In Advances in Wireless, Mobile Networks and Applications, Springer, Berlin, Heidelberg*, 191-200.

Schneier, Bruce. 1998. " The Twofish encryption algorithm." *Dr. Dobb's Journal: Software Tools for the Professional Programmer 23, no. 12* 30-34.

—. 1993. "Description of a new variable-length key, 64-bit block cipher (Blowfish)." *In International Workshop on Fast Software Encryption, Springer, Berlin, Heidelberg*, 191-204.

Sethi, K., A. Pradhan, and P. Bera. 2020. "Smart Grid Data Security using Practical CP-ABE with Obfuscated Policy and Outsourcing Decryption." *2020 International Conference on Cyber Situational Awareness, Data Analytics and Assessment (CyberSA).* Dublin: IEEE.

Shirey, Robert. 2007. "Internet security glossary, version 2." *RFC 4949*, August.

Shoukry, Y., K. Gatsis, A. Alanwar, G.J. Pappas, S.A. Seshia, M. Srivastava, and P. Tabuada. 2016. "Privacy-aware quadratic optimization using partially homomorphic encryption." *2016 IEEE 55th Conference on Decision and Control (CDC).* Las Vegas: IEEE. 5053-5058.

Singh, A., R.M. Parizi, Q. Zhang, K.K.R. Choo, and A. Dehghantanha. 2020. "Blockchain smart contracts formalization: Approaches and challenges to address vulnerabilities." *Computers & Security* 88: 101654.

Stallings, William. 2005. "The RC4 stream encryption algorithm." *Cryptography and network security.*

Staroletov, S., and R. Galkin. 2019. "Towards Hyperledger Sawtooth: Formal Verification of Proof-of-Elapsed Time Algorithm and Testing Methods of Enterprise Blockchain Applications."

Steinwandt, R. 2010. "A ciphertext-only attack on Polly Two." *Applicable Algebra in Engineering, Communication and* 21 (2): 85–92.

Steven R. Hart, Eysha S. Powers, and James W. Sweeny. 2018. "Format-preserving encryption of base64 encoded data." *U.S. Patent 10,015,008*, July 3.

Swan, M. 2018. "Blockchain Economics:“Ripple for ERP”." Accessed 8 19, 2020. https://melanieswan.com/documents/RippleERP.pdf.

Sylvia Osborn, Ravi Sandhu, and Qamar Munawer. 2000. "Configuring role-based access control to enforce mandatory and discretionary access control policies." *ACM Transactions on Information and System Security (TISSEC) 3, no. 2*, 85-106.

Szabo, N. 1994. "Smart Contracts." Accessed 8 20, 2020.

Technology, U.S. DEPARTMENT OF COMMERCE/National Institute of Standards and. 1999. "Data Encryption Standard (DES)." 1–22.

Thomsen, Daniel J. 1990. "Role-Based Application Design and Enforcement." *In DBSec*, 151-168.

Tong, Eric Yuan and Jin. 2005. "Attributed based access control (ABAC) for web services." *In IEEE International Conference on Web Services (ICWS'05). IEEE.*

Treleaven, P., R.G. Brown, and D. Yang. 2017. "Blockchain Technology in Finance." *Computer* 50 (9): 14-17.

Unal, D., M. Hammoudeh, and M.S. Kiraz. 2020. "Policy specification and verification for blockchain and smart contracts in 5G networks." *ICT Express* 6 (1): 43-47.

Van Ly, L. 2006. "Polly two: A new algebraic polynomial-based public-key scheme." *Applicable Algebra in Engineering, Communication and Computing* (Ph.D. Dissertation. Ruhr University Bochum) 17 (3): 267–283.

Vashikar, S., P. Mandge, R. Pagar, and A. Jadhav. 2020. "Decentralized Cloud Storage using IPFS and Ethereum." *International Journal of Advance Computational Science and Engineering Technology (IJACSET)* 1 (2): 1-7.

Verma, D., N. Desai, A. Preece, and I. Taylor. 2017. "A block chain based architecture for asset management in coalition operations." *Ground/Air Multisensor Interoperability, Integration, and Networking for Persistent ISR VIII* 10190: 101900Y.

Víctor Gayoso Martínez, Luis Hernández Encinas, and Carmen Sánchez Ávila. n.d. "A survey of the elliptic curve integrated encryption scheme." 2010.

Vijayakumar, V., K.M. Sabarivelan, J. Tamizhselvan, B. Ranjith, and B. Varunkumar. 2019. "Utlization of Blockchain in Medical Healthcare Record using Hyperledger Fabric." *International Journal of Research in Advent Technology* 7 (4): 414-419.

Vincent C. Hu, David Ferraiolo, Rick Kuhn, Arthur R. Friedman, Alan J. Lang, Margaret M. Cogdell, Adam Schnitzer, Kenneth Sandlin, Robert Miller, and Karen Scarfone. n.d. "Guide to attribute based access control (abac) definition and considerations (draft)." *NIST Publications.*

Wang, C., and J. Luo. 2013. "An Efficient Key-Policy Attribute-Based Encryption Scheme with Constant Ciphertext Length." *Mathematical Problems in Engineering* 2013.

Wang, G., Q. Liu, and J. Wu. 2010. "Hierarchical Attribute-Based Encryption for Fine-Grained Access Control in Cloud Storage Services." *Proceedings of the 17th ACM Conference on Computer and Communications Security.* Chicago: Association for Computing Machinery. 735-737.

Wang, S., and Y. Zhang. 2018. "A Blockchain-Based Framework for Data Sharing With Fine-Grained Access Control in Decentralized Storage Systems." *IEEE Access* 6: 38437-38450.

Wang, W., N. Hu, and X. Liu. 2019. "BlockZone: A Blockchain-Based DNS Storage and Retrieval Scheme." *Artificial Intelligence and Security. ICAIS 2019. Lecture Notes in Computer Science* 11635: 155-166.

Wang, Y. 2016. "Octonion algebra and noise-free fully homomorphic encryption (FHE) schemes." *arXiv ePrint Archive Cornell University Library.*

Wang, Z. 2017. "An Identity-Based Data Aggregation Protocol for the Smart Grid." *IEEE Transactions on Industrial Informatics* 13 (5): 2428-2435.

Waters, B. 2011. "Ciphertext-Policy Attribute-Based Encryption: An Expressive, Efficient, and Provably Secure Realization." *Public Key Cryptography – PKC 2011. PKC 2011. Lecture Notes in Computer Science* 6571: 53-70.

Waters, B. 2005. "Efficient Identity-Based Encryption Without Random Oracles." *Advances in Cryptology – EUROCRYPT 2005. EUROCRYPT 2005. Lecture Notes in Computer Science* 3494: 114-127.

Wei-hong, H.U., A.O. Meng, S.H.I. Lin, X.I.E. Jia-gui, and L.I.U. Yang. 2017. "Review of blockchain-based DNS alternatives." *Chinese Journal of Network and Information Security* 3 (3): 71-77.

Wilson, David D. Clark and David R. 1987. "A comparison of commercial and military computer security policies." *In 1987 IEEE Symposium on Security and Privacy, IEEE*, 184-184.

Wonyoung Jang, and Sun-Young Lee. 2020. " A format-preserving encryption FF1, FF3-1 using lightweight block ciphers LEA and, SPECK." *In Proceedings of the 35th Annual ACM Symposium on Applied Computing*, 369-375.

Wonyoung Jang, and Sun-Young Lee. 2020. "Partial image encryption using format-preserving encryption in image processing systems for Internet of things environment." *International Journal of Distributed Sensor Networks 16, no. 3.*

Xiao, L., O. Bastani, and I.L. Yen. 2012. "An Efficient Homomorphic Encryption Protocol for Multi-User Systems." *IACR Cryptol. ePrint Arch* (IACR Cryptol. ePrint Arch.) 193.

Xin Jin, Ram Krishnan and Ravi Sandhu. 2012. "A unified attribute-based access control model covering DAC, MAC and RBAC." *In IFIP Annual Conference on Data and Applications Security and Privacy, Springer, Berlin, Heidelberg*, 41-55.

Xin Jin, Ram Krishnan, and Ravi Sandhu. 2012. "A unified attribute-based access control model covering DAC, MAC and RBAC." *In IFIP Annual Conference on Data and Applications Security and Privacy, Springer, Berlin, Heidelberg*, 41-55.

Xiong, Z., Y. Zhang, D. Niyato, P. Wang, and Z. Han. 2018. "When Mobile Blockchain Meets Edge Computing." *IEEE Communications Magazine* 56 (8): 33-39.

Xu, Z., and K.M. Martin. 2012. "Dynamic User Revocation and Key Refreshing for Attribute-Based Encryption in Cloud Storage." *2012 IEEE 11th International Conference on Trust, Security and Privacy in Computing and Communications.* Liverpool: IEEE. 844-849.

Yang, Q., J. Yang, W. Yu, D. An, N. Zhang, and W. Zhao. 2014. "On False Data-Injection Attacks against Power System State Estimation: Modeling and Countermeasures." *IEEE Transactions on Parallel and Distributed Systems* 25 (3): 717-729.

Yokoo, M., and K. Suzuki. 2002. "Secure Multi-Agent Dynamic Programming Based on Homomorphic Encryption and Its Application to Combinatorial Auctions." *Proceedings of the First International Joint Conference on Autonomous Agents and Multiagent Systems: Part 1.* Bologna: Association for Computing Machinery. 112–119.

Zhang, L., Q. Wu, and Y. Hu. 2012. "Hierarchical Identity-Based Encryption with Constant-Size Private Keys." *ETRI Journal* 34 (1): 142-145.

Zhang, X., C. Xu, C. Jin, R. Xie, and J. Zhao. 2014. "Efficient fully homomorphic encryption from RLWE with an extension to a threshold encryption scheme." *Future Generation Computer Systems* 36: 180-186.

Zheli Liu, Chunfu Jia, Jingwei Li, and Xiaochun Cheng. 2010. "Format-preserving encryption for datetime." *In 2010 IEEE International Conference on Intelligent Computing and Intelligent Systems, vol. 2, IEEE*, 201-205.

Zyskind, G., O. Nathan, and A. Pentland. 2015. "Enigma: Decentralized Computation Platform with Guaranteed Privacy." *arXiv preprint arXiv:1506.03471.*