



ASCLEPIOS

Advanced Secure Cloud Encrypted Platform for Internationally Orchestrated Solutions in Healthcare

Project Acronym: **ASCLEPIOS**

Project Contract Number: 826093

Programme: **Health, demographic change and wellbeing**

Call: **Trusted digital solutions and Cybersecurity in Health and Care
to protect privacy/data/infrastructures**

Call Identifier: **H2020-SC1-FA-DTS-2018-2020**

Focus Area: **Boosting the effectiveness of the Security Union**

Topic: **Toolkit for assessing and reducing cyber risks in hospitals and care
centres**

Topic Identifier: **H2020-SC1-U-TDS-02-2018**

Funding Scheme: **Research and Innovation Action**

Start date of project: 01/12/2018

Duration: 36 months

Deliverable:

D2.3 GDPR-compliant and Privacy-Preserving Analytics for Healthcare Providers

Due date of deliverable: 31/05/2020

Actual submission date: 03/06/2020

WPL: TUNI - Antonis Michalas

Dissemination Level: Public

Version: Final



Executive Summary

This deliverable, titled “GDPR-compliant and Privacy-Preserving Analytics for Healthcare Providers”, constitutes a report on the activities performed in the context of Task T2.3, titled “GDPR-compliant and Functional Encryption-enabled Prescriptive Analytics for Healthcare Providers” and T2.4, titled “Cybersecurity, Encryption and Access Analytics for CSP operation to Healthcare Providers”.

In the scope of T2.3 activities, a detailed state of the art analysis of the functional encryption (FE) cryptographic paradigm is first performed to examine how its capability of allowing the application of a function over encrypted data, revealing only the result of the performed function on the ciphertexts, could be of value for healthcare providers. In this context, the theoretical foundations of functional encryption are explored, its sub-categories are identified and specific approaches for each sub-category are studied and presented. Focus is given on the Inner Product FE schemes, which were identified as the most promising in the ASCLEPIOS context. Applications based on functional encryption are examined and technical limitations are identified and presented. The ASCLEPIOS functional encryption analytics services that allow healthcare providers to perform statistical computations over encrypted data are then designed and implemented. Both the symmetric and asymmetric key settings are leveraged in the implemented services which offer single input and multi input functional encryption functions. A symmetric multi-input functional encryption scheme is developed as part of T2.3 activities and is documented in detail. Sequence diagrams are created to showcase the supported workflows and foreseen user interactions. The role of the functional encryption services within the ASCLEPIOS framework and the ways they can be combined with the framework’s search functionality (implemented through searchable encryption) to offer a more complete user experience are also explored. Furthermore, GDPR considerations regarding the provided services for functional encryption-enabled analytics on healthcare data are discussed.

In the scope of T2.4 activities, a landscape analysis of the cyber threats that healthcare organisations need to face in order to safeguard their assets is performed and extracted insights are presented. Relevant cybersecurity approaches, both methodological and technical, are also explored. The need for targeted solutions offering insights into data handling processes, including access, encryption and decryption patterns, for healthcare providers is identified and a state of the art review of data analysis and visualisation methods that can be used to implement such solutions is presented. The ASCLEPIOS Cybersecurity, Encryption and Access Analytics for Healthcare Providers (CEAA) component is designed and implemented to help healthcare providers gain a deeper understanding on their system regarding these processes through various defined metrics, detect abnormal behaviour and leverage the extracted knowledge to build threat preventive mechanisms. CEAA is implemented using powerful open source technologies and offers the following functionalities (a) ingests and contextualises log data from data access and processing services used by the healthcare provider, (b) performs fast queries and computations over the collected data, (c) detects and highlights potential anomalies in the collected data, (d) provides meaningful insights to the security analyst in an intuitive way and (e) enables the security analyst to adapt the data and analysis representation to the current needs. Finally, the application of CEAA to provide insights about security, access and encryption patterns that can be extracted within the ASCLEPIOS framework, i.e. for a healthcare provider that utilises the ASCLEPIOS services, is explored and showcased.

Table of Contents

Table of Contents.....	3
List of Figures and Tables.....	5
Status, Change History and Glossary.....	7
1 Introduction.....	13
1.1 Scope of the deliverable.....	13
1.2 Structure.....	14
2 Functional Encryption Landscape.....	15
2.1 Introduction.....	15
2.2 Subclasses of Functional Encryption.....	16
2.2.1 Purely Cryptographic Functional Encryption.....	16
2.2.2 Hardware-enabled Functional Encryption.....	24
2.3 Applications of Functional Encryption.....	25
2.3.1 General applications.....	25
2.3.2 Health-related applications.....	27
2.4 Functional Encryption Considerations.....	28
2.4.1 Considerations on Indistinguishability and Simulation Security.....	28
2.4.2 Information Leakage due to Provided Functionality.....	28
2.4.3 Performance Issues.....	30
2.4.4 Hardware-enabled approaches.....	30
3 ASCLEPIOS Functional Encryption System.....	31
3.1 Symmetric MIFE Scheme.....	31
3.1.1 Preliminaries.....	32
3.1.2 Multi-Input Functional Encryption for the ℓ_1 Norm.....	33
3.1.3 From Single-Client to Multi-Client MIFE.....	35
3.2 FE Services Implementation & Workflows.....	36
3.2.1 Symmetric MIFE services.....	37
3.2.2 Asymmetric IPFE services.....	38
3.3 FE Analytics Usage in ASLEPIOS.....	41
3.4 GDPR-considerations.....	41
4 Cybersecurity Landscape in Healthcare.....	44
4.1 Cyber Threats in Healthcare.....	44
4.2 Cybersecurity Frameworks and Tools.....	47
4.3 Current Status and Challenges.....	51
4.4 Data Analytics for Cyber Threat Detection.....	52
4.4.1 Supervised Methods.....	53
4.4.2 Unsupervised Methods.....	54
4.4.3 Semi-supervised Methods.....	54
4.4.4 Ensemble learning Methods.....	55

D2.3 GDPR-compliant and Privacy-Preserving Analytics for Healthcare Providers

4.5	Data Visualisation for Cyber Threat Detection	55
4.6	Privacy Considerations.....	57
5	ASCLEPIOS Cybersecurity, Encryption and Access Analytics for Healthcare Providers	58
5.1	Motivation and Scope	58
5.2	Design Methodology.....	58
5.2.1	Definition of Target Usage.....	58
5.2.2	Identification of Input Sources and Data	59
5.2.3	Definition of Metrics.....	60
5.2.4	Requirements and Considerations	62
5.3	CEAA Design	64
5.3.1	Architecture	64
5.3.2	Functionalities and Workflow.....	65
5.4	CEAA Interface and Usage	67
6	Conclusions	78
	References.....	80

List of Figures and Tables

Figures

Figure 1: Methodological work for T2.3 and T2.4 activities.....	13
Figure 2: one-AD-IND secure MIFE for Inner Products	34
Figure 3: one-AD-IND secure MIFE for the ℓ_1 norm.....	34
Figure 4: Hybrid Games for the proof of Theorem 1	35
Figure 5: Multi Input MIFE for the ℓ_1 norm.....	36
Figure 6: Symmetric MIFE example.....	38
Figure 7: Asymmetric IPFE example	39
Figure 8: Asymmetric IPFE – Encrypt Only	40
Figure 9: Asymmetric IPFE – Invoke Function Only	40
Figure 10: Threat taxonomy for smart hospitals by ENISA [71].....	46
Figure 11: Initial Points of Compromise for Security Incidents [74].....	47
Figure 12: NIST Cybersecurity Framework [75].....	48
Figure 13: CEAA Architecture.....	64
Figure 14: CEAA High-Level Workflow	66
Figure 15: CEAA Dashboard I	70
Figure 16: CEAA Dashboard II	71
Figure 17: Dashboard Filters & Core Metrics.....	72
Figure 18: Dashboard Filters Completed	73
Figure 19: Requests distribution based on invoked services and requestor's role between working and non-working hours of the organisation	73
Figure 20: Annotated requests distribution over time	74
Figure 21: Functions usage across days of the week.....	74
Figure 22: Daily change in number of requests performed per department.....	75
Figure 23: Service heatmap over days of week for internal and external requests	75
Figure 24: Failed requests distribution and policy insights	76
Figure 25: Annotated data granularity distribution of requests over time.....	76
Figure 26: Service distribution (percentages)	77

Tables

Table 1: Status Change History	7
Table 2: Deliverable Change History	7
Table 3: Glossary.....	12
Table 4: Parameter size and operation time complexity analysis for [1].....	19
Table 5: Parameter size analysis.....	20
Table 6: Operation time complexity analysis	20

D2.3 GDPR-compliant and Privacy-Preserving Analytics for Healthcare Providers

Table 7: A taxonomy of the most common algorithmic families used for anomaly detection.53

Status, Change History and Glossary

Status:	Name:	Date:	Signature:
Draft:	Evmorfia Biliri, Nefeli Bountouni	29/05/2020	Evmorfia Biliri
Reviewed:	Dimitris Apostolou	01/06/2020	Dimitris Apostolou
Approved:	Tamas Kiss	03/06/2020	Tamas Kiss

Table 1: Status Change History

Version	Date	Pages	Author	Modification
v0.1	20/11/2019	13	Suite5	Draft table of contents
v0.2	20/12/2019	20	SECURA B.V.	Input on functional encryption state of the art and GDPR perspectives
v0.3	24/01/2020	26	Suite5	Input on functional encryption state of the art
v0.4	10/02/2020	37	Suite5	Input on cybersecurity analytics state of the art
v0.5	27/02/2020	42	Suite5	Input on CEAA design and metrics definition
v0.6	08/05/2020	63	Suite5	Update content in all sections
v0.7	22/05/2020	70	TUNI	Input on ASCLEPIOS FE system
v0.8	29/05/2020	83	Suite5	Final draft version
v0.9	01/06/2020	83	ICCS	Peer review version
v1.0	02/06/2020	86	Suite5	Final version

Table 2: Deliverable Change History

Glossary

AD-IND	Adaptive Indistinguishability
ABAC	Attribute Based Access Control
ABE	Attribute Based Encryption
ADV	Adversary
ANN	Artificial Neural Network
APT	Advanced Persistent Threat
ARIMA	Autoregressive Integrated Moving Average
ARMA	autoregressive moving average
BP-ANN	Back Propagation Artificial Neural Network
CCA2	Adaptive Chosen Ciphertext Attack
CEAA	Cybersecurity, Encryption and Access Analytics
C-FE	Controlled Functional Encryption
CI	Critical Infrastructure
CJEU	Court of Justice of the European Union
CPU	Central Processing Unit
CSP	Cloud Service Provider
CUSUM	Cumulative Sum control chart
Dx.y	Deliverable x.y
DCRA	Decisional Composite Residuosity Assumption

D2.3 GDPR-compliant and Privacy-Preserving Analytics for Healthcare Providers

DDH	Decisional Diffie-Hellman
DevOps	Software Development and Information Technology Operations
DLIN	Decision Linear
DNF	Disjunctive Normal Form
DoS	Denial of Service
DT	Decision Tree
EC	European Commission
ECG	Electrocardiogram
EDM	E-Divisive with Medians
EGADS	Extensible Generic Anomaly Detection System
EHR	Electronic Health Record
EM	Expectation Minimisation
ENISA	European Union Agency for Cybersecurity
EU	European Union
EV	Evaluator
FE	Functional Encryption
FHE	Fully Homomorphic Encryption
FSA	Finite State Automata
GAN	Generative Adversarial Network
GDPR	General Data Protection Regulation

D2.3 GDPR-compliant and Privacy-Preserving Analytics for Healthcare Providers

GRU	Gated Recurrent Unit
HE	Homomorphic Encryption
HMM	Hidden Markov Models
HTTP	Hyper Text Transfer protocol
IBE	Identity Based Encryption
ICT	Information and Communication Technology
ID	Identifier
IDS	Intrusion Detection System
IND	Indistinguishability
IND-CPA	Indistinguishability Chosen Plaintext Attack
iO	Indistinguishability Obfuscation
IoT	Internet of Things
IP	Internet Protocol
IPFE	Inner Product Functional Encryption
IPS	Intrusion Prevention System
IT	Information Technology
JCDE	Java Card Development Kit
JSON	JavaScript Object Notation
K-NN	k-Nearest Neighbors
LA	Los Angeles

D2.3 GDPR-compliant and Privacy-Preserving Analytics for Healthcare Providers

LMS	Log Management System
LOF	Local Outlier Factor
LSTM	Long Short Term Memory
LWE	Learning with Errors
MCD	Minimum Covariance Determinant
MDDH	Matrix Decisional Diffie Hellman
MIFE	Multi Input Functional Encryption
ML	Machine Learning
MSK	Master Secret Key
MUMIFE	Multi User Multi Input Functional Encryption
NC	Nick's Class (complexity)
NHS	National Health Service
NIST	National Institute of Standards and Technology
OS	Operating System
PKE	Public Key Encryption
PP	Master Public Key
RCCA	Replayable Chosen Ciphertext Attack
RNN	Recurrent Neural Network
RSA-OAEP	Rivest-Shamir-Adleman – Optimal Assymmetric Encryption Padding
SARIMA	Seasonal Autoregressive Integrated Moving Average
SGX	Software Guard Extensions
SIEM	Security Information and Event Management
SIM	Simulation
SEL-IND	Selective Indistinguishability

D2.3 GDPR-compliant and Privacy-Preserving Analytics for Healthcare Providers

SOM	Self-Organising Map
SSE	Symmetric Searchable Encryption
SVM	Support Vector Machine
SXDH	Symmetric External Diffie-Hellman
TA	Trusted Authority
Tx.y	Task x.y
TEE	Trusted Execution Environment
UK	United Kingdom
US	United States
UTM	Unified Threat Management
VARMA	Vector Autoregression
WPx	Work Package x
WSGI	Web Server Gateway Interface
3-PDDH	3-Party Decisional Diffie Hellman

Table 3: Glossary

1 Introduction

1.1 Scope of the deliverable

The purpose of the present deliverable spans across two axes which correspond to the two tasks whose activities it reports on, namely Task T2.3, titled “GDPR-compliant and Functional Encryption-enabled Prescriptive Analytics for Healthcare Providers” and T2.4, titled “Cybersecurity, Encryption and Access Analytics for CSP operation to Healthcare Providers”.

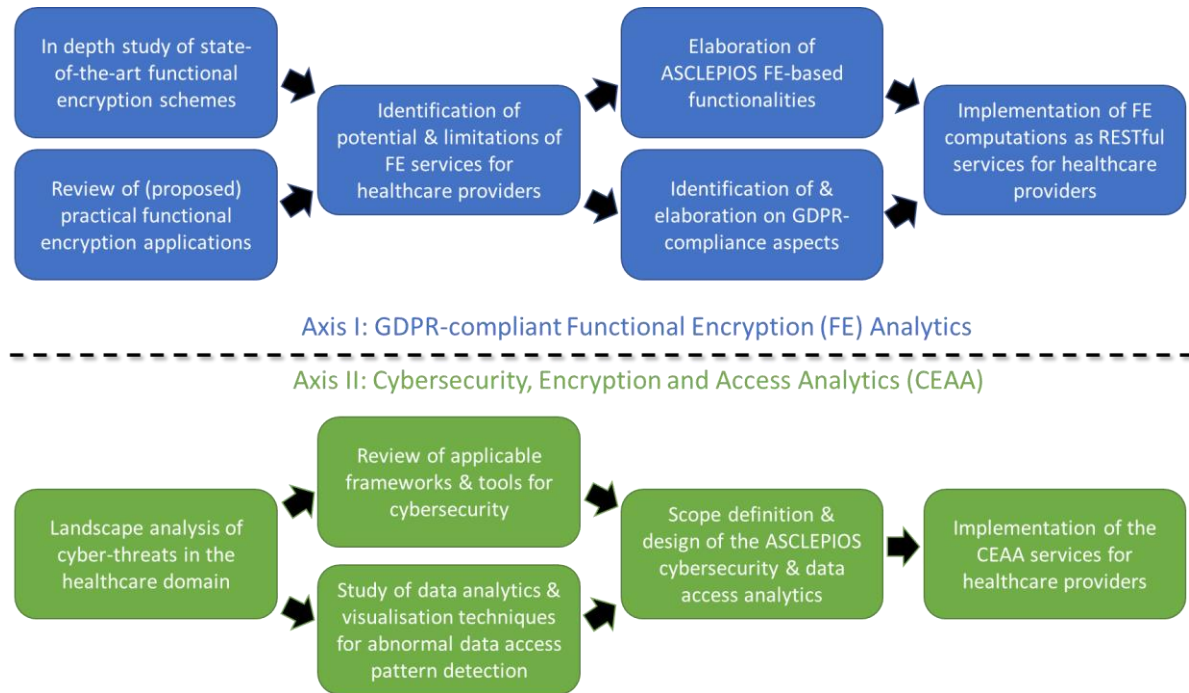


Figure 1: Methodological work for T2.3 and T2.4 activities

Towards the first axis, the scope of the deliverable is to provide data analytics functionalities to healthcare providers, focusing on performing computations over encrypted data, i.e. leveraging the functional encryption (FE) cryptographic paradigm. To this end, the initial purpose of the deliverable is to investigate the true potential and the limitations of functional encryption and the extent to which it can support real-world analytics processes from healthcare providers, ranging from simple mathematical computations to statistics and to more advanced analytics algorithms, including machine learning methods. The second objective in this direction is to use the insights gained from the state of the art review to explore how ASCLEPIOS can leverage functional encryption to provide data analytics services that address concrete needs of healthcare providers, to identify which functionalities would be of added value in this context and what are the GDPR compliance perspectives to consider. The final purpose related to the T2.3 activities, which is reported in this document, is to design, implement and deliver a set of functional encryption-enabled services to be used by healthcare providers.

Towards the second axis, which is related to the T2.4 activities, the purpose of the deliverable is twofold. The first purpose is to define metrics to be monitored and analytics processes to be applied in order to (a) closely follow data access activities, including encryption and decryption, (b) measure security incidents, (c) identify emerging threats and trends, (d) sort out patterns of abnormal and insecure behavior and finally (e) help healthcare providers gain a better understanding of their system which will help them build the necessary preventive mechanisms around it from a cybersecurity perspective. In order to define these metrics, the deliverable first presents insights extracted from a landscape review on cybersecurity perspectives in the healthcare domain, including important cyber-

D2.3 GDPR-compliant and Privacy-Preserving Analytics for Healthcare Providers

threats against which healthcare organisations need to safeguard their infrastructures, applicable frameworks and tools, as well as data analytics and visualisation methods that can contribute in fleshing out more targeted solutions for the analysis of data access patterns in healthcare organisations. The second purpose is to deliver to healthcare providers these analytics and metrics in a packaged solution that will offer insights into current and past status of their systems regarding security, encryption and access, and will thus contribute in securing their infrastructures

1.2 Structure

The remainder of the present deliverable is structured as follows:

- Section 2 provides an in-depth analysis of the functional encryption paradigm as part of T2.3 activities. The section presents the sub-categories of functional encryption and numerous proposed approaches in the domain and provides insights into functional encryption applications, advantages, and limitations.
- Section 3 presents in the context of T2.3 activities the ASCLEPIOS functional encryption analytics services which leverage symmetric and asymmetric functional encryption schemes and offer both single- and multi-input data analytics functionalities for healthcare providers.
- Section 4 presents, in the context of T2.4, a landscape review of cybersecurity threats, challenges and solutions in healthcare and provides a state of the art analysis of data analytics and visualisation methods that can be used to develop targeted solutions towards helping healthcare providers understand data access, encryption and decryption patterns in their infrastructures.
- Section 5 presents, in the context of T2.4, the Cybersecurity, Encryption and Access Analytics (CEAA) component which is responsible for monitoring data access patterns in order to provide meaningful and actionable insights to healthcare providers to help them increase their cybersecurity level.
- Section 6 summarises the content of the deliverable and draws relevant conclusions.

2 Functional Encryption Landscape

The first step towards developing the ASCLEPIOS services that will offer healthcare providers data analytics functionalities applicable over encrypted data is to understand the mechanisms that can be used to enable such solutions. Towards this goal, the project focuses on the application of Functional Encryption (FE). Functional Encryption is a cryptographic paradigm allowing fine-grained access control over data [1] and revealing only partial information, going beyond the ‘all-or-nothing’ approach of traditional public key encryption systems [2]. More specifically, in functional encryption a key generation mechanism – the authority - creates decryption keys of restricted capabilities, allowing the user to learn the result of a function over encrypted data without learning anything else about the data [3].

The present section provides a thorough review of FE schemes and applications to provide insights regarding the ways in which FE can be used to provide valuable services to healthcare providers.

2.1 Introduction

The formal definition of a Functional Encryption scheme for a functionality F defined over (K, X) as a deterministic Turing Machine, is defined as a tuple of four algorithms, namely (FE.Setup, FE.Keygen, FE.Enc, FE.Dec) [3]. This tuple is encountered throughout the relevant literature, with variations in the properties and implementation of the four algorithms to support different classes of functions (ex. quadratic functions) or guarantee additional security properties such as ‘function-hiding’. Nonetheless, the core structure of the functional encryption scheme remains the same, and is the following:

- **FE.Setup.** A probabilistic algorithm that takes as input a security parameter λ and outputs a master public/secret key pair (PP, msk) ,
- **FE.Keygen.** A probabilistic algorithm that takes as input the master secret key msk and a key k and outputs a secret key sk_k for k ,
- **FE.Enc.** A probabilistic algorithm that takes as input the public key PP and a message m and outputs a ciphertext c ,
- **FE.Dec.** A probabilistic algorithm that takes as input sk_k , for some k , and a ciphertext c and outputs a part of the ciphertext in plaintext.

These four algorithms can be formally expressed as:

- $(PP, msk) \leftarrow FE.Setup(1^\lambda)$
- $sk_k \leftarrow FE.Keygen(msk, k)$
- $c \leftarrow FE.Enc(PP, m)$
- $y \leftarrow FE.Dec(sk_k, c)$, we require that $y = F(k, m)$ with probability 1.

The security of functional encryption relies heavily on the classical security notion of *indistinguishability*, which requires that it is impossible for an adversary with access to the secret keys corresponding to functions $F_1 \dots F_n$ to detect which of the input messages m_0, m_1 has been used in a computation, provided that $F_i(x_0) = F_i(x_1)$ for all i [4]. Indistinguishability ensures collusion resistance, meaning that a group of users with different keys sk_F can only learn from the encrypted messages only the union of information they would already get individually and nothing more about the messages [1]. However, it has been proved that this security definition is weak and in cases, inadequate. The stronger notion of simulation-based security of functional encryption has been proposed. Although it has been proved that the use of simulation-based definitions leads to impossibility results [5], other adaptive indistinguishability-based schemes were found to imply simulation-based security in restricted environments [6].

2.2 Subclasses of Functional Encryption

Over the last few years, functional encryption has attracted the attention of the scientific community. With the shift of mentality to outsourcing data storage on external cloud infrastructures, the need has emerged for privacy preserving computations and fine-grained access control. Functional encryption offers an attractive alternative to traditional public key encryption techniques that lack the expressiveness to address either of these issues [7], as the decryption keys provide full visibility of the encrypted data to the users who own the key. There exist other cryptographical approaches that secure computations over data, without imposing an important compromise on the data analysis potential and functionalities. The most known techniques are Homomorphic Encryption (HE) and Fully Homomorphic Encryption (FHE), which allow computations over encrypted data. However, as the computed results are in their turn also encrypted and remain confidential to the server, these approaches are not of great use for the users who need access to open results. This trait of Homomorphic/Fully Homomorphic Encryption imposes also restrictions with regards to practical issues, such as the training of machine learning models with encrypted data [8]. In contrast to HE/FHE, functional encryption outputs the results “in the clear”, thus allowing the interested party to learn the computation information she wants, while the owner of the data can control what is allowed to be leaked from her data and to whom [9].

A wide variety of approaches towards the design and implementation of efficient and secure functional encryption schemes is available in the literature. A first categorisation can be made based on whether the cryptographic primitives and notions used are based on purely cryptographic-based functional encryption schemes, or whether specially designed hardware infrastructure for hardware-based functional encryption instantiation is utilised. It should be noted that the implementation of a purely cryptographic encryption scheme does not preclude usage of hardware assistance, e.g. to generate keys, but it denotes that the core mechanism is based on manipulating encrypted data and not on leveraging hardware properties of trusted execution environments.

Cryptographic approaches can be further analysed based on the class of functionalities supported by the proposed schemes. These can be grouped under the following categories: Predicate Encryption (where the supported functionality is the provision of fine-grained access with the use of predicates), Inner Product Encryption (functional encryption schemes for the computation of inner products), Functional Encryption for Element-wise Operations (supporting basic arithmetic operations such as addition and subtraction), Quadratic Functional Encryption (for the computation of quadratic polynomials), General Polynomial Functions (aiming to construct schemes for any polynomial-time function), and Functional Encryption for Randomised Functionalities (using randomness for the computation of results).

Several techniques have been employed for the realisation of functional encryption. Indicative examples include among others the use of cryptographic pairings [9][10][11], indistinguishability obfuscation (iO) [12][13][14], lattices [15] or even the introduction of additional players in the standard functional encryption scheme [16]. The security of the proposed schemes has been based on related security assumptions, such as the Decisional Diffie-Hellman assumption (DDH) for pairing-based implementations, the Learning with errors assumption (LWE) for lattice-based schemes and the Decisional Composite Residuosity assumption (DCRA) that is behind Paillier cryptosystems.

2.2.1 Purely Cryptographic Functional Encryption

This section refers to functional encryption implementations that are based on cryptographic notions. Purely cryptographic approaches constitute the main body of existing research around functional encryption. A multitude of different schemes have been constructed, aiming to provide an improved solution in terms of performance, supported functionalities and security.

2.2.1.1 Predicate Encryption

Predicate encryption is a public key encryption paradigm where secret keys correspond to predicates P (i.e. boolean functions) and can be used to decrypt a ciphertext associated with an attribute ind , provided that the predicate value for this attribute is true ($P(ind)=1$) [17]. Predicate encryption enables the sender of the message to define specific policies and determine who will be able to decrypt the data, thus allowing for fine-grained access control. The Predicate Encryption scheme for a class of predicates P over a set of attributes Σ is formally described as a tuple of four algorithms, namely (Setup, Keygen, Enc, Dec) [17]:

- Setup. Takes as input a security parameter λ and outputs a master public/secret key pair (PP, msk) ,
- Keygen. Takes as input the master secret key msk and a (description of a predicate) $p \in P$ and outputs a secret key sk_f corresponding to f ,
- Enc. Takes as input the public key PP , an attribute $ind \in \Sigma$ and a message m and outputs a ciphertext c , this can be written as $c \leftarrow Enc_{PP}(ind, m)$
- Dec. Takes as input sk_f and a ciphertext c and outputs either a message m or the distinguished symbol \perp

We require that for all λ , all (PP, msk) generated by $Setup(1^\lambda)$, all $f \in F$, any key sk_f and all $ind \in \Sigma$:

- If $p(ind) = 1$ then $Dec(sk_f, c) = m$
- If $p(ind) = 0$ then $Dec(sk_f, c) = \perp$

This general notation for predicate encryption has been used as the base for the construction of systems offering advanced access control functionalities. Following are some of the most interesting variations and extensions of the standard predicate encryption notion.

2.2.1.1.1 Identity Based Encryption (IBE)

In IBE a unique, publicly available string with information about the user (i.e. identity) is used as the public encryption key. This string is also used by the trusted authority for the generation of the secret decryption key [18]. This means that both the ciphertext and the private key are associated with strings corresponding to identities, and the decryption key can reveal the message m only if the two strings are equal. Any string could serve as an identity, as for example an email address, an IP address, a location and more.

IBE can be viewed as functional encryption for the class of equality tests. The standard IBE notion ensures only payload hiding, although anonymous IBE can provide also the stronger notion of attribute hiding [19]. The message m is encrypted to the 'identity' string, which now serves as the index ($c \leftarrow Enc_{PP}(identity, m)$). The user with identity id who wants to decrypt ciphertext c , obtains a secret key sk_{fid} from the trusted authority. The predicate $P(ind, m)$ is true if and only if $ind=id$, and in this case the user can learn the message m . Otherwise the predicate p is false and the user learns nothing about messages encrypted for other identities [20]. A drawback of the standard IBE is that the policy index is returned in plaintext as part of the empty functionality. Anonymous IBE tries to overcome this leak of possibly sensitive information, as the string of the index remains hidden and the user can only infer whether she has a key corresponding to the ciphertext without learning anything about the used identity string. The only difference of anonymous IBE from standard IBE is that the function is defined as $f(\epsilon, (ind, m)) = len(m)$ revealing only the length of m .

2.2.1.1.2 Attribute Based Encryption (ABE)

ABE is another public-index predicate encryption notion that allows the definition of complex access policies [3]. The encryptor can specify a policy ϕ that consists of a specific combination of attributes and determine which data recipients will be able to decrypt the message, based on their attributes. These policies are formulated as Boolean formulas. The user has an attribute vector $u: \{0,1\}^*$, which places 1 in the position of every attribute that is

D2.3 GDPR-compliant and Privacy-Preserving Analytics for Healthcare Providers

true for the user, and in all other attribute places it remains a 0. From this point there are two approaches in ABE:

In Ciphertext-Policy ABE, also described in detail in [11][21][22] the policy ϕ is used as index in the functional encryption scheme ($c \leftarrow \text{Enc}_{\text{CPP}}(\phi, m)$). The user who wants to decrypt the ciphertext, obtains a secret key sk_f from the authority, based on her attribute vector u . The predicate function f returns the message m to the user if and only if the access policy is satisfied ($p_u(\phi, m) = m$ if $(\phi(u) = 1)$). Otherwise the user learns nothing about the data. In [11] a combination of Ciphertext-Policy ABE and Functional Encryption has been implemented to provide a system for analysis of medical data. In this case the user runs analytics over the ciphertext resulting from the ABE encryption. This model achieves payload privacy; however, user privacy depends on the selection of available keywords for the function definition.

In Key-Policy ABE [23], the roles of the policy and the user attributes vector are reversed. This means that the vector u is placed in the ciphertext ($c \leftarrow \text{Enc}_{\text{CPP}}(u, m)$) and policy ϕ is part of the secret key sk_f . The predicate function $p(u, m)$ will return the message m if the policy ϕ is true for u .

Both approaches have been proved safe in the weak selective security model.

2.2.1.1.3 Predicate-only Encryption and Multi-client Predicate-only Encryption

Predicate-only encryption is a variation of standard predicate encryption scheme, where the ciphertexts contain only the index attribute and the message m is omitted. In this case the result of the decryption is only the output of predicate p for the index [24].

[10] propose a predicate-only encryption scheme for testing the equality of two encrypted vectors from multiple clients. Multi-client functional encryption has been formally defined in [14] as a functional encryption scheme that would allow the computation of $f(x_1, x_2, \dots, x_n)$ from n ciphertexts of underlying inputs x_1, x_2, \dots, x_n . In their work, they view multi-input functional encryption as a way of performing multiparty computation, where the input ciphertexts are computed by the n different parties [10] focuses on a restricted functionality of predicates – namely conjunctive equality tests -, instead of trying to implement arbitrary functionalities, to achieve better efficiency. Their scheme is using pairings and achieves both attribute and predicate privacy.

2.2.1.2 Inner Product Functional Encryption

Extensive work has been conducted towards the construction of functional encryption schemes for the inner product functionality. In an inner product encryption scheme, secret keys sk_x are associated with vector $x \in Z$ and ciphertexts are associated with vector $y \in Z$. Given a secret key sk_x for x and ciphertext ct_y for y , the decryption outputs only the computation of $\langle x, y \rangle$, or in other words the inner product of the associated vectors x and y , without revealing any other information [25]. Although the inner products are less general than other functions (e.g. general circuits), they offer the required expressiveness to be used in various practical scenarios.

The extensive literature around inner product functional encryption can be considered promising for applying such schemes in practical scenarios that are of interest to ASCLEPIOS, therefore some schemes will be studied in more detail.

In their paper, Abdalla et al [1] gave the first construction of an IPE scheme, based on the Decisional Diffie-Hellman assumption. It exploits the homomorphic property of the ElGamal scheme.

Lemma. If $\mathcal{E}(\cdot)$ is the ElGamal encryption, we have

$$\begin{aligned}
 \mathcal{E}(m_1) \cdot \mathcal{E}(m_2) &= (g^{r_1}, m_1 \cdot h^{r_1})(g^{r_2}, m_2 \cdot h^{r_2}) \\
 &= (g^{r_1 + r_2}, (m_1 \cdot m_2) h^{r_1 + r_2}) \\
 &= \mathcal{E}(m_1 \cdot m_2)
 \end{aligned}$$

D2.3 GDPR-compliant and Privacy-Preserving Analytics for Healthcare Providers

If we choose $m_i = g^{x_i}$, it follows that $\prod_{i \in I} \mathcal{E}(g^{x_i})^{y_i} = \mathcal{E}(g^{x \cdot y})$. As such we can retrieve the inner product by computing the discrete logarithm. We cannot use this directly however, as decrypting the results requires knowledge of the secret key that can be used to obtain the plain texts as well. Instead, we use a different public key $h_i := g^{s_i}$ for every coefficient in the cyphertext while reusing the same encryption randomness g^r . This conserves the homomorphic property and the notion of a shared secret from ElGamal.

This limits our practical message space by the time complexity of the discrete logarithm, as shown in Table 4:

Table 4: Parameter size and operation time complexity analysis for [1]

Parameter	Size	Operation	Time complexity
mpk	l group elements	Setup()	$O(l)$
msk	l group elements	Encrypt()	$O(l)$
Ct	$l + 1$ group elements	KeyDer()	$O(l)$
Sk	l group elements	Decrypt()	$O(l + 2 \sqrt{M})$

The same paper also gives a way to construct a similar inner-product scheme for any PKE with certain properties, such as LWE (Learning with errors). The latter allows for a construction without the need to compute the discrete logarithm.

In [4] the authors build their inner product functional encryption scheme based on the plain Decisional Diffie Hellman assumption. Their scheme is proven secure in the selective security model, against unbounded - but polynomially related to the security parameter - secret key queries. The authors note that regardless the implementation, there are inherent security weaknesses in the inner product functionality, that cannot be overcome with any security construction. For example, an adversary that possesses the secret keys Sk_{y_i} for y_i that form the basis for the finite field Z , is able to uncover the secret vector x .

Preservation of function privacy, which entails that both the keys and the ciphertexts remain private, is a desired property for many applications. Bishop et.al [26] additionally consider the function-hiding property, where the secret key for (\cdot, \vec{y}) does not reveal \vec{y} . They build a function-hiding private key functional encryption scheme for inner product functionality, with asymmetric bilinear maps, based only on the Symmetric External Diffie-Hellman (SXDH) assumption to achieve unbounded indistinguishability-based security. They leverage the private key setting in order to prove function-hiding; a feature impossible in other public key schemes. They employ the idea for computing the inner product by placing two vectors in the exponents of opposite side of a bilinear group and computing the dot product via pairing, as proposed in [17]. In contrast to the initial scheme however, they introduce additional group elements in order to output the actual inner product result and not just a zero/non-zero value.

From the parameter size analysis (Table 5), we can see that while the size of the secret key grows exponentially, its generation mostly consists of sampling random integers. As such we can avoid storing and transferring the key itself and instead use a shared secret that is fed into a deterministic key derivation function.

D2.3 GDPR-compliant and Privacy-Preserving Analytics for Healthcare Providers

Table 5: Parameter size analysis

Parameter	Size
msk	$4l^2 \log p$
Ct	$(2l+6) \log p$
Sk	$(2l+6) \log p$

Regarding time complexity of the individual functional encryption steps, we see in Table 6 that the setup, encryption and decryption grow quadratically in the vector length. However, we point out that this is because of rather mundane operations (multiplication and pseudorandom sampling) that can be efficiently computed in parallel. In most practical applications, we expect the decryption operation to be the bottleneck.

Table 6: Operation time complexity analysis

Operation	Time complexity
Setup()	$O(l^2)$
sEncrypt()	$O(l^2)$
KeyDer()	$O(l^2)$
Decrypt()	$O(l + 2^{M/2})$

According to [25], the above presented security model is based on unrealistic admissibility constraint on the adversary's queries. [27][28][29] are three subsequent works that improved the original SXDH-based scheme of [26] in terms of parameter size.

The construction of [25] performs better than all these three schemes, by using n -dimensional vectors with just $n+1$ group elements, thus reducing even more the parameter sizes. They provide a construction that is slightly more efficient: instead of generating dual orthonormal vectors, they simply sample invertible matrices and publish its determinant:

$$\mathbf{B} \in \text{GL}_n(\mathbb{Z}_q), \mathbf{B}^* := \det(\mathbf{B})(\mathbf{B}^{-1})^T$$

While this only provides a slight improvement compared to the previous construction, it has a proof-of-concept implementation that allows us to test its performance. Additionally, they prove the security of their scheme on the stronger simulation-based notion in the generic group model.

In order to limit the information leakage which is inherent to the inner product functionality, the authors of [16] introduce an additional player in the traditional functional encryption scheme, that will add another security safeguard before revealing the computation result to the client. The addition is a helper responsible to apply restriction policies to the number of queries a user is eligible to make. It is required for the helper to be oblivious, as it is assumed to be untrusted and should not learn anything on the actual plaintext or the queries. The authors also apply another restriction to information provided to the user: instead of providing openly the result through a decryption step, their construction only

D2.3 GDPR-compliant and Privacy-Preserving Analytics for Healthcare Providers

returns whether the inner product result is within a set of values the user has provided previously and this is actually the output of the decryption test step.

The authors of [30] study the feasibility of inner product functional encryption schemes achieving “in-between” indistinguishability and simulation-based security, in the sense of providing guarantees for a restricted but wider than indistinguishability-based class of attack scenarios that are known to be feasible. They propose a setting where the encryption key is not publicly known and at the same time it is not identical with the master secret key for the generation of the user secret keys. Their construction is a private-key, function-hiding inner product functional encryption scheme for n length attributes and predicates, based on prime order bilinear groups. They prove this scheme to be to be fully simulation secure in the generic group model under the Decision Linear (DLIN) assumption. The authors claim that their security framework provides a strong tool with real-world potential, as their hybrid public/symmetric key setting allows for user roles with distinct permission rights regarding key generation for other users and creation of encrypted content.

Another desired attribute for the application of functional encryption in real-life scenarios is the support of computations over encrypted data coming from different and unrelated sources. **Multi-input Functional Encryption (MIFE)**, as explained before, tries to address this issue. [1] identifies the weak points of previous approaches, namely the reliance on unstable assumptions of some schemes for general functionalities. Secondly, it points out the inefficiency of the discrete-log-based solution in terms of supported messages size and decryption computation requirements of the pairing-based solution presented in [31]. Furthermore, it proposes a secret-key MIFE scheme for the inner product functionality based on [31], but this time without the use of bilinear maps. Their scheme is basically a transformation of any single-input inner product FE scheme to multi-input, dropping the bilinear groups requirement thus allowing for efficient calculations even on super-polynomial size messages. Their construction can be instantiated from any known single input FE scheme and can use various assumptions (e.g. plain DDH, composite residuosity, and LWE) to obtain MIFE. To showcase the validity of their generic construction they instantiate it with three different inner product FE schemes from the literature [15], which are based on the MDDH assumption, the LWE assumption and the Paillier cryptosystem respectively. They also provide a MIFE scheme with the function-hiding property, using a double-layered encryption approach as in [32]. Unlike their general construction, in this case they rely on pairing groups in order to ensure function-hiding.

An interesting cryptographic tool for enhanced FE inner product functionality is featured in [33]. The authors develop a solution that combines CCA2 or randomisable RCCA secure public-key encryption with garbled circuits and build a scheme that achieves fine-grained access control, by requesting a fresh key for every function or ciphertext evaluation request as an extra security layer. Controlled FE (C-FE) has advantages over FE, overcoming the simulation impossibility results of traditional FE when no limitation to the number of requests applies while using well-known cryptographic assumptions, and allowing the construction of efficient schemes for arbitrary functions. The authors constructed an efficient scheme, secure against malicious clients and honest-but-curious authorities in the non-function hiding security model. To evaluate their solution, they implemented what they named as “Superfast Inner-Product Construction” in Java, using the JCDE RSA-OAEP implementation, and demonstrated its efficiency in the decryption stage.

2.2.1.3 Functional Encryption for Element-wise Operations

[8] proposes a framework to support the training of a neural network over encrypted data using an underlying combination of their functional encryption construction for basic arithmetic operations and inner-product functional encryption for secure matrix computations as designed in [4]. Functional encryption was chosen over HE, in order to overcome the limitations of the latter in enabling evaluation of labelling during the back-propagation time, due to the confidentiality of computed results.

D2.3 GDPR-compliant and Privacy-Preserving Analytics for Healthcare Providers

Their construction for element-wise operations with functional encryption is derived from ElGamal encryption [34]. Regarding security, this scheme can resist an unauthorised access without the function key, but cannot prevent the direct inference of encrypted message m from the function result of these basic operations and their own y from the output of $f(x\Delta y)$, where x is the message, and Δ is one of the supported functions (+, -, *, /). To address this issue, they propose a preprocessing step, such as random mapping, to avoid direct inference for label. After describing the functional encryption schemes for secure computations, they showcase how their framework CryptoNN can be used for the training of a classic convolutional neural network for multiple classification (LeNet-5) with encrypted data. They implement their prototype using the Charm library, a crypto toolkit in Python, and Numpy for the implementation of the neural network. As the functional encryption schemes they utilise do not support floating point number operations, they made a trade-off in precision for time efficiency. Their model performed similarly to the original model in terms of accuracy, while it required longer training time due to the cryptographic computations.

2.2.1.4 Functional Encryption for Quadratic Polynomials

In [35] the authors go one step further than previous approaches that went up to linear functionalities. They build two functional encryption schemes that allow the calculation of quadratic functions over linear-size ciphertexts, using bilinear maps over integers. Their first constructed scheme is actually a family of public-key functional encryption schemes supporting the bilinear map functionality, built using a private-key, single ciphertext functional encryption scheme as a building block. This scheme is proven selectively secure under standard assumptions (MDDH and 3-pddh). Their second public-key functional encryption scheme is a relatively simpler and more efficient construction, proven secure in the generic group model for indistinguishability-based security. [9] introduce their own functional encryption scheme for quadratic functions and use it to train an image classification model with encrypted images. They highlight a common practical issue among functional encryption schemes; the need for discrete logarithm computations, and tackle it by constraining the model weights to get smaller outputs, while employing the Baby-Step Giant-Step Algorithm which transfers the main volume of computations in the preprocessing time. This functional encryption scheme relies also on bilinear groups and is proven secure in the generic group model. They implement their functional encryption scheme with the Pairing-based cryptography library of the Charm framework, and train a polynomial network classifier in TensorFlow. Performance comparison with the two quadratic functional encryption schemes in [35] show that this model is slightly more efficient. Another implementation of the quadratic scheme by [9] was realised in the context of Fentec¹, an EU project developing functional encryption technologies. Their paper [36] reports on their implementation and proposed use cases. Additionally, the two implementations (in C and Go languages) are available through a Github repository².

A recent scheme for quadratic functional encryption, used to build privacy preserving neural networks with application in image classification problems can be found in [37]. The proposed scheme is based on bilinear pairings and proven secure under the indistinguishability security notion, against adaptive adversaries in the generic group model, as in [35]. However, it outperforms the previously presented quadratic functional encryption schemes, in terms of overall complexity and ciphertext and decryption key size. Another point of distinction from the previous constructions is the introduction of counter-measures to collateral learning (i.e. the phenomenon of learning unexpected features from leaked information). This is accomplished with a two-fold strategy: first the limitation of the number of the outputs only to the necessary. Secondly, with the use of an adversarial training procedure in the first layers of the neural network, which enables the neural network to adapt against collateral training. The generalisation of the model against diverse adversaries is a

¹ <http://fentec.eu/>

² <https://github.com/fentec-project/neural-network-on-encrypted-data>

D2.3 GDPR-compliant and Privacy-Preserving Analytics for Healthcare Providers

challenge and although the used model was proved sufficient against a wide range of neural networks, the study was still not exhaustive.

2.2.1.5 Functional Encryption for General Polynomials

Research has been conducted on the capability of applying functional encryption in the computation of arbitrary polynomial-time functions. Impossibility results have shown fundamental limitations on the realisation of simulation-based security for functional encryption supporting arbitrary functions [3][5]. Furthermore, as per [38] the construction of functional encryption schemes for arbitrary functions with unbounded collusion resistance is still an open issue, so they rely on homomorphic encryption and tamperproof hardware tokens to overcome these impossibility results. However, according to [33] this solution suffers from the vulnerabilities of such hardware to reveal information, while at the same time these deterministic oracles are not based on concrete primitives [39]. The employment of garbled circuits for secure computations in the context of functional encryption is another alternative. An IND-CPA secure construction is provided in [40], where the authors introduce the notion of ‘Worry free Encryption’ (a scheme closely related to functional encryption and conditional disclosure of secrets) and implement it with the help of Yao’s garbled circuits. Their implementation however shows some disadvantages due to efficiency issues of reusable garbled circuits that limited them to computing only single functions [41] in addition to the encryption of each garbled wire with a different key, which grows significantly the ciphertext size, rendering this approach inefficient for many practical applications [33]. In [42] the authors tried to address the garbled circuit reusability issue and proposed a scheme for the evaluation of multiple functions based on the work by [40]. This attempt showed that even for a limited number of functions, the computations would require hundreds of years.

The work by [33] investigated the construction of controlled functional encryption schemes for the computation of inner-product and arbitrary polynomial-time functions. As ‘Controlled Functional Encryption’ (C-FE) they define a cryptographic tool, that similar to Functional Encryption, allows the user to learn only the result of a function over encrypted data, with the difference that the authority generates one-time keys. As a consequence, a new key request must be submitted by the user each time she wants to make a computation over a new ciphertext, in this controlled setting. Their general function implementation also adopts Yao’s garbled circuits [43] for the secure computations. The definition of C-FE includes two additional algorithms in comparison to the standard 4-algorithm Functional Encryption definition, namely: ‘KeyReq’ algorithm, where a new request is submitted to the authority for the issue of a new key, and the ‘Extract’ algorithm for the extraction of the policy parameter of the ciphertext in order to decide whether the user’s request is eligible.

The authors proceeded with the description of two schemes under this definition; an extremely efficient scheme for inner-product functionality (more details in Section 2.2.1.2) and a general construction for arbitrary polynomial-time function families. Their solution for the evaluation of a general function f over the encryption of data x with policy λ consists of the following steps: The authority generates a public-key encryption key pair (PP, msk) . Using the master public key, the owner encrypts the data pair (x, λ) and creates two ciphertexts (α, σ) . In order for the requestor to compute $f(x)$, he must first acquire a key from the authority, by sending a key request with parameters the function f and the policy ciphertext σ . After recovering λ from the decryption of σ , the authority can decide whether to honour the request. If the decision is positive, the authority generates the decryption key, by computing a function F' with two-party secure computation. F' takes as input the encrypted message α from the user side, the function f and the msk from the authority side, and outputs the $F(\text{Decryption}(\alpha), f)$. The user can use this output to obtain the value of $f(x)$. To encounter malicious attacks, the authors propose the use of a CCA2 secure public-key encryption scheme, or the use of Rerandomisable RCCA to additionally achieve pattern hiding. In order to overcome computation efficiency issues, they ensure that any secure

D2.3 GDPR-compliant and Privacy-Preserving Analytics for Healthcare Providers

computations with garbled circuits happen only for f and not f' , which involves decryption operations and would make the model extremely inefficient.

[12][44] approach the challenge of general polynomial computations with functional encryption from the point of constructing a scheme supporting all polynomial-size circuits. They employ the notion of indistinguishability obfuscation suggested by [45]. An indistinguishability Obfuscator iO for a class of circuits C guarantees that given two equivalent circuits $C1$ and $C2$ from the class, the two distribution of obfuscations $iO(C1)$ and $iO(C2)$ should be computationally indistinguishable. [12] propose a candidate indistinguishability obfuscator iO for all polynomial-size log-depth circuits (NC^1), using a simplified variant of multilinear maps (i.e. Multilinear Jigsaw Puzzles). Afterwards they combine their iO for NC^1 with injective one-way functions, public-key encryption and zero knowledge proofs, to obtain functional encryption for all circuits with indistinguishability security.

2.2.1.6 Functional Encryption for Randomised Functionalities

Going beyond functional encryption for all deterministic polynomial-time functions, research has considered the case of secure randomised functionalities calculation with functional encryption. Randomised functionalities not only set a challenge in the field of cryptography, but are also of great significance in real-life scenarios. In randomised functional encryption the requestor can only learn about the result of a randomised function over the encrypted data with private randomness, without inferring any more information about the data. The correctness criterion of a randomised functional encryption scheme requires the computational indistinguishability of the distribution of results $f(x)$ from the distribution of the randomised function directly computed over the encrypted data x on fresh randomness [46].

A problem in the context of functional encryption, is imposed by randomness, and more specifically by the need to remain non-inferable to the participating parties and especially dishonest encryptors who could tamper with the random coins. To overcome such problems, in [44] the authors construct a randomised functional encryption scheme with simulation-based security. In their construction they use various cryptographic primitives, namely indistinguishability obfuscation [45], puncturable pseudorandom functions, non-interactive witness indistinguishable proof systems. Simulation-based security however could not be realised for unbounded messages. Thus, they provide additional indistinguishability-based definitions and prove selective security for unbounded number of messages.

[46] introduces a randomised functional encryption scheme based on indistinguishability security. As the main interest of this paper is to explore the feasibility of constructing fully homomorphic encryption schemes from sufficiently strong randomised functional encryption, they focus on constructing schemes that imply FHE, instead of exploring the wide range of random functionalities that could be supported by functional encryption. They build a generic randomised functional encryption scheme for any n -ary randomised function RF_n , from an entropically secure FE supporting a specific functionality F_n of the same arity.

2.2.2 Hardware-enabled Functional Encryption

Advances in secure processor technologies have opened new paths for the implementation of privacy-preserving data analytics and computation frameworks. Processor extensions, such as Intel SGX³ and ARM TrustZone⁴, offer hardware and software-enabled security guards to enable the creation of secure execution environments (i.e. 'enclaves'), that provide a trusted memory area and use hardware based encryption to enable private and secure execution of code [47]. The enclaves offer three main functionalities, Isolation (seclusion of data and code inside the enclave memory area, while processes external to the enclave are not able to read or tamper them), Sealing (data from the enclave to be transferred and/or stored externally, is encrypted with a hardware-resident key and can later be retrieved by the enclave for the continuation of computations) and Attestation (a special key attests that the

³ <https://www.intel.com/content/www/us/en/architecture-and-technology/software-guard-extensions.html>

⁴ <https://developer.arm.com/ip-products/security-ip/trustzone>

D2.3 GDPR-compliant and Privacy-Preserving Analytics for Healthcare Providers

specific data, code, metadata and computed outputs are secluded in the enclave) [48]. TEEs however have certain drawbacks that impose limitations on the range of possible applications. An SGX CPU has 128 MB processor reserved memory available, causing performance issues [49]. Additionally, the SGX is susceptible to side-channel attacks, creating the requirement for programs intended for enclave execution, to be data-oblivious [48].

Various papers have proposed secure systems with the use of trusted execution environments. [50] designed a system to enable the distributed computation of MapReduce in cloud infrastructures, while ensuring with the integration of SGX enclaves the confidentiality of code and data. Another example is [51], a framework for deep neural networks that leverages TEEs for non-linear operations which are hard to secure, while outsourcing linear operations to faster co-located untrusted processors.

The idea of using trusted execution environments and more specifically SGX, for the emulation of a functional encryption setting has been suggested in the literature. In [48] the system comprises a single trusted authority and an arbitrary number of dynamically added decryption nodes, both powered by SGX. The standard functional encryption 4-algorithm scheme is now enabled by the functionalities of the enclaves. The trusted authority generates the master public and secret key. It provides the secret key to the decryption enclave(s) after performing remote attestation. Any encryption takes place using the generated public key. When a client application wants to perform a function over encrypted data, it requests a signature from the trusted authority, that will authorise the requested function. Afterwards the client sends the encrypted data, the function and the signature to the decryption enclave, which in turn checks the signature and provided it is valid, proceeds with the decryption of the data (using the master secret key), runs the requested function and returns only the result to the client. When this step is complete, all invalid signatures are aborted from the enclave. The IRON system was evaluated with three functional encryption constructions: Identity based encryption from pairings, order revealing encryption and 3 input DNF. To achieve side-channel attack resilience, they propose oblivious implementations of the functions requested by the client applications, which are the only, apart from decryption operations, that interact with the plaintext data.

2.3 Applications of Functional Encryption

Some of the most interesting use cases illustrating the practical value of the various proposed functional encryption implementations are presented in this section. The authors provide guidelines on the transition from simple functional encryption enabled calculations, such as inner products, matrix calculations, quadratic polynomials etc., to real-life scenarios. The possible applications span over a wide area, from privacy-preserving handling of biometric data for biometric-based authentication systems, to the facilitation of software obfuscation for critical operations, and more.

As the main focus of ASCLEPIOS is the implementation of privacy-preserving analytics on medical data with the assistance of functional encryption, a separate subsection is dedicated to health-related applications.

2.3.1 General applications

Fine-grained access control and sharing based on policies is an obvious use of attribute-based and identity-based encryption schemes [2][20][52], that can be applied in a multitude of domains (e.g. healthcare, insurance, government institutions, universities and more). In this use case, a predicate expresses an access policy and authorisation is granted if the requestor's key satisfies the predicate function.

Searching over ciphertexts to retrieve only data that satisfy a condition, is another application of functional encryption. [14] describe as an example that a user could retrieve all items that are less than a specific value by executing a binary comparison function with this specific value as threshold. With their flexible scheme, such searches could be realised for any query and even for data from different users as sources. The inner product functionality is considered as a potential enabler for a multitude of practical applications. [25] approaches

D2.3 GDPR-compliant and Privacy-Preserving Analytics for Healthcare Providers

the problem of **retrieving similar documents** as a **nearest-neighbour search** utilising the inner product functionality. More specifically, they use the inner product to compute the Euclidean distance between vectors corresponding to documents, in order to allow users find similar documents. In this setting, an authorised user acquires a master secret key from the authority. The user must project the document query to the vector space and afterwards encrypt this vector using the secret key. The authentication server computes the Euclidean distance between the query vector and the stored vectors corresponding to the available documents, and returns the set of documents with the smallest distance.

[4] showcases how inner products can be used in **descriptive statistics**. The weighted mean is a tool to describe the features of an information collection. In their work they present how a simple inner product DDH-based functional encryption scheme can be designed to compute the final course grades from individual tests with different weights. The function-hiding inner product functionality of [25] is also envisaged as the basis for **biometric authentication**. In this case, the inner product provides an efficient way to compute the Hamming distances of secret vectors. In their biometric authentication setup, the biometric scanners encrypt the user's biometric using the master secret key, provided by the authentication server. Subsequently, the scanners send the encrypted biometric to the authentication server, which in turn computes the Hamming distance between the secret vector of the biometric and the (also secret) user's stored credential. If the Hamming distance is adequately small, indicating satisfying similarity, the authentication succeeds.

Spam filtering has been mentioned as a task for encrypted operations in [20]. The authors of [37] proposed a use case scenario for **advanced automated email filtering operations**, in a recent work where they examine with more details how their quadratic functional encryption scheme could be used for such operations. They propose functional encryption enabled **classification** for this task, using public labels for non-sensitive information (e.g. the 'spam' label) and private labels for flags revealing personal information (ex. the hobbies of the recipient). The email recipient has a master public key which can be used by the sender to encrypt the email. The email server can perform classification over the encrypted email using the secret key and label the incoming email as spam or not. Furthermore, the server can infer if the mail falls under some of the other categories and decide whether to generate an 'urgency' alert or not. An adversary however is not able to infer anything about the private categories, only by learning the public outcomes of the mail filtering. In their experimentation, they use image recognition as an analogy to the envisioned use case and utilise the MNIST dataset⁵ to create a synthetic dataset, where the digit serves as the public classification feature and the font as the private feature.

[10] build their special purpose multi-client predicate-only functional encryption construction for equality checks, with the domain of **critical infrastructure monitoring** in mind. They are based on the fact that certain combinations of events should generate an alert and express this statement as predicates that take status messages from the individual operators and become true when the problematic combination appears. Their use cases involve firstly the detection of large scale cyberattacks through the sudden failure of multiple systems from different critical infrastructure (CI) operators, and secondly the prediction of demand shifts among CI operators due to disruptions, in order to take proactive supply measures. Both cases are facilitated by the sharing of the CI operators' status messages with a central authority. These messages are centrally collected and are encrypted due to their sensitive nature, as they could also include information such as the 'cybersecurity level' among others. The central authority can then evaluate the predicate function over these ciphertexts and if it becomes true, it raises an alert. As they would expect the monitoring authority possibly to be a third party, a requirement for this application is to ensure plaintext and predicate privacy.

Restricted-use software is another field, which has been explored by [12] as an application for their indistinguishability obfuscation functional encryption construction for general circuits. On certain cases, the developers may want to make available only partial functionality of

⁵ <http://yann.lecun.com/exdb/mnist/>

D2.3 GDPR-compliant and Privacy-Preserving Analytics for Healthcare Providers

their software rather than the full version. This can happen for demonstration or pricing reasons. The authors propose that indistinguishability obfuscation can facilitate such functionality by enabling the release of one software version restricted up to interface level and one that is obfuscated. They go on with more use cases, such as the release of obfuscated security patches for strategic purposes (not reveal vulnerabilities to attackers) and the protection of intellectual property rights by obfuscating new algorithms to prevent reverse-engineering.

A real-life scenario incorporating randomised functions, is illustrated in [44]. In their example, a bank transactions auditing procedure requires the **generation of a random sample of database entries**. This use case has two main requirements: firstly, the auditor must be sure that she will catch improper transactions with reasonable probability and secondly that a malicious auditor would not have access to 'too much' information, meaning that the auditor only has access to the random sample and not to arbitrary information about a client. [44] construct a functional encryption scheme supporting arbitrary randomised polynomial-size circuits for such applications.

[7] focuses on some crypto-oriented applications of functional encryption. These include the construction of a protocol for **verifiable computation** from the transformation of an ABE for general circuits, which enables a computationally weak client to hand over a complex computation to an untrusted server and be assured that the returning result cannot be incorrect. They also derive a **scalable signature** scheme from IBE, which can bridge the gap between securing large datasets without compromising performance. Lastly, they mention the use of functional encryption schemes for the construction of fully homomorphic encryption, a theme that has been featured extensively in [46].

2.3.2 Health-related applications

Functional encryption is a promising concept for a domain such as healthcare, where the vast majority of data are by definition personal and sensitive, thus imposing great limitations on related scientific research, personalisation and improvement of healthcare services. Following are some of the possible applications, as presented in literature.

Personalisation of medicine using the patient's genome, is a revolutionary concept. [33] provide a flow based on functional encryption for the execution of a disease susceptibility tests. The digitised genome is encrypted by a genotyping agency with the Controlled functional encryption public key issued by the authority. The ciphertext is then publicly available and can be used by a medical unit for the conduction of the disease susceptibility test, using the one-time function key (also issued by the authority) for this test. **Measuring similarity between genomes** enabling better treatment based on the past reactions of other patients with similar symptoms and genetic build-up, has also been studied in the context of functional encryption. Various application-specific schemes have been proposed, however [33] claim that in contrast to the other approaches, their C-FE scheme addresses this challenge in an efficient and highly scalable way – important feature for use in large populations. [36] demonstrate functional encryption for **disease predictions over encrypted data** and more specifically, they evaluate the risk of cardiovascular diseases. They utilise various multivariate algorithms that have been developed based on the Framingham heart study [53]. To implement this functionality they use the inner product functional encryption scheme on Pailler cryptosystem of [15], on a typical functional encryption infrastructure comprising of a central authority for key generation, an analysis component that performs functions on the encrypted data, and a client that encrypts her data with the public encryption key and sends the ciphertext to the analysis component. **Encrypted healthcare data processing for diagnostic purposes** is mentioned by [8] as a typical use case for machine learning over encrypted data. They address the challenge of training a model over encrypted data with their CryptoNN framework which is built on functional encryption for basic element operations and secure matrix computations. **Medical monitoring** through statistical computations on medical data coming from sensors and the patient's profile is investigated in [54], where a methodology for multiparty statistics with the

D2.3 GDPR-compliant and Privacy-Preserving Analytics for Healthcare Providers

use of multi-input functional encryption is presented, with the adoption of a tree topology to improve efficiency. **Enhanced security and privacy in IoT healthcare infrastructures** is another field of potential application for functional encryption. FE finds a two-fold application in the framework proposed in [11]: ciphertext policy ABE imposes enhanced access control to centralised data, while the integration of traditional FE allows the execution of secure analytics on encrypted data. Finally, in [55], a list of potential scenarios for private computations on medical data has been compiled from existing work in literature. Indicatively these scenarios include: The support of **privacy-preserving Electronic Health Record system**, where the patient is in control of her data and can define who has access even to selected areas of the record. **Medical research** can benefit from the availability of encrypted medical data upon which machine learning can be performed in a secure way. **Verifiability of outsourced computations** is another issue; especially now as healthcare providers choose more and more often cloud solutions.

2.4 Functional Encryption Considerations

2.4.1 Considerations on Indistinguishability and Simulation Security

A recurring consideration expressed in the literature concerns the weaknesses of the standard security notion of functional encryption, namely the indistinguishability-based security. Although indistinguishability ensures in theory that the messages are secure even for unbounded number of collusions, [3] and [56] showed that indistinguishability is weak, as trivially insecure schemes can be proved IND-secure. Given the inadequacy of this notion, extensive work has been conducted in numerous papers towards constructing schemes that are secure under the stronger notion of simulation security. The impossibility results for simulation security in all settings involving a simulated adversary with access only to the results of the functionality provided by the FE scheme [3] led to the formation of schemes that attempt to “bootstrap” IND-secure FE to SIM-security, compromising with weaker simulation security – but still enhancing it in comparison to indistinguishability security. Some indicative cases are those of sim-security in the generic group model [25][30], selective security for bounded number of messages [44], non-function hiding security models in a setting of malicious clients and honest-but-curious authorities [33].

2.4.2 Information Leakage due to Provided Functionality

The considerations discussed in this section do not rely on weaknesses in the implementation, but are an inevitable consequence of the functionality provided through the functional encryption scheme. By definition, functional encryption reveals information about the plaintext to those with the right keys. This becomes a fundamental problem when the knowledge of multiple function outputs, gives more information than the union of the individuals, as for example in the case of inner products where one can combine multiple inner products to learn more about the underlying data.

2.4.2.1 Matrix inversion

If an attacker can obtain at least n different functional keys for the same dataset, it constructs a matrix of the associated inner products. Multiplying the inverted matrix with the associated coefficients yields the entire dataset:

$$\vec{x} = [x_1 \quad \cdots \quad x_n] = [\langle \vec{x}, \vec{y}_1 \rangle \quad \cdots \quad \langle \vec{x}, \vec{y}_n \rangle] \times \begin{bmatrix} y_{1,1} & \cdots & y_{1,n} \\ \vdots & \ddots & \vdots \\ y_{n,1} & \cdots & y_{n,n} \end{bmatrix}^{-1}$$

D2.3 GDPR-compliant and Privacy-Preserving Analytics for Healthcare Providers

Where y_i, j is the i -th coefficient of vector y . It is required that the coefficient matrix is invertible, but this will almost always be the case when the vectors y_j are chosen independently from each other.

2.4.2.2 Singular vectors

Intuitively, the inner product with a (nearly) singular vector (e.g. where all but one of the components is negligible) leaks one of the components of the other vector: For vector v that is (nearly) singular in v_i , we get $v \cdot u = v_i u_i + \dots$. To prevent the deduction of individual values from the data, we must ensure that no keys are issued for (nearly) singular vectors.

It must be noted that multiple vectors can be combined to yield a new, different vector. Given $(\vec{x}, \vec{y}), (\vec{x}, \vec{y}')$, one can obviously compute $(\vec{x}, a\vec{y} + b\vec{y}') = a(\vec{x}, \vec{y}) + b(\vec{x}, \vec{y}')$ for any a, b . Therefore, it must also be ensured that no linear combination of vectors $a_1\vec{v}_1 + a_2\vec{v}_2 + \dots + a_n\vec{v}_n$ is (nearly) singular.

This quickly becomes a problem when we add or remove values from the dataset, as this results in vectors that differ only in one (or a few) dimensions. Let $(\vec{v}', \vec{u}') := (v v_i, u u_i)$, where (\vec{v}, \vec{u}) is the concatenation of the vectors in their dimensions. We then can easily retrieve $v_i u_i = \langle \vec{v}', \vec{u}' \rangle - \langle \vec{v}, \vec{u} \rangle$

The most obvious mitigation is to limit the functional keys that a party has. An oblivious helper that is required for each operation can help with this, such as with the Oblivious Helper.

2.4.2.3 Mean and variance

Suppose we have a sample of n values, of which the sample mean μ and variance σ^2 . Even without knowing anything about the underlying distribution, we can give an upper and lower bound on the sample maximum and minimum, respectively.

For a given σ^2, μ, n , we will find (without loss of generality) the maximum possible sample value b . This (multi)set is of the form $X = \{a, \dots, a, b\}$, where

$$\begin{aligned}\sigma^2 &= E[(X - \mu)^2] = \sum [X^2]/n - \mu^2 \\ &= \frac{a^2(n-1) + b^2}{n} - \mu^2, \\ \mu &= \frac{a(n-1) + b}{n}\end{aligned}$$

$$b = \mu \pm \sqrt{\sigma^2 * (n-1)}$$

Solving for b gives

Lemma. *In a set of n values with mean μ and variance σ^2 , all values are in the range $[\mu - \sqrt{\sigma^2 * (n-1)}, \mu + \sqrt{\sigma^2 * (n-1)}]$.*

This knowledge can then be used in re-identification attacks.

2.4.2.4 Application-specific attacks

The weaknesses described are far from theoretical. In the absence of a stronger security definition, it becomes easy to imagine a scenario where the security requirements are violated. For example, Ligier et al. [57] demonstrates how Principal Component Analysis and other machine learning techniques can be used to get information on the input of a linear classifier (or any other single-input IPE scheme) when it has a training set that is similar to

D2.3 GDPR-compliant and Privacy-Preserving Analytics for Healthcare Providers

the input. This approach yields highly accurate predictions on the input data when an attacker obtains sufficient functional keys.

2.4.3 Performance Issues

The trade-off between the supported functionalities and performance issues is also discussed. Some approaches attempt to construct general-purpose FE schemes [12], while others step on the inefficiency of these schemes in terms of key and ciphertext sizes, as well as computation time, which make them impractical for real-life applications and focus their efforts in supporting limited functionalities, such as the inner-product functionality [25].

2.4.4 Hardware-enabled approaches

Regarding hardware-enabled functional encryption, two points shall be taken into consideration. The first concerns the susceptibility of SGX to side-channel attacks [48], be it either physical attacks by an adversary with physical CPU access, or software attacks on the host of the CPU, as for example through compromised OS. Several software attacks have exploited the privileged interfaces to undermine the system's security as in [58][59]. Secondly, the memory size limitations of SGX [48] (only 128MB of memory area are available to enclaves, and from those the applications can use approximately 90MB) impose on their turn limitations to the complexity of supported functionalities and to the size of datasets that could be imported to the enclave to be used for the computations. Such limitations can be overcome with an appropriate system design, however as this may entail the exchange of data from the secured space with the memory of the system outside the enclaves, further encryption and security requirements may occur.

As all security solutions are only designed to protect against only a subset of potential attacks, the focus of security design requirements shall not be on making a system impossible to bypass, but rather on making the success of an attack as "costly" - in terms of time and money - as possible for the attacker⁶.

⁶ <http://infocenter.arm.com/help/index.jsp?topic=/com.arm.doc.prd29-genc-009492c/ch01s01s02.html> (last accessed 13/05/2020)

3 ASCLEPIOS Functional Encryption System

Section 2 presented a comprehensive state of the art review of functional encryption schemes and applications and provided insights into the advantages and the limitations of this cryptographic paradigm. ASCLEPIOS aspires to leverage functional encryption to offer healthcare providers inherently secure ways to perform analytics over sensitive data without disclosing more information than required and whilst keeping the underlying data encrypted. It should be noted that although IBE, ABE and Predicate Encryption were briefly discussed, the ASCLEPIOS FE analytics focus on schemes that can provide computations over the underlying data. For such operations, inner product schemes, as explained, are the most promising since they go beyond access control and can also be used to implement specific computation functions, as opposed to other proposed generic constructions. The landscape analysis demonstrates a hype in inner product functional encryption work in the academic field, still in many cases the perspective is more theoretical, and limitations enforced on practical applications are not considered relevant. Specific applications have been recently proposed, some of them exploring also more advanced analytics functionalities, including neural networks. Yet, the underlying assumptions may not be appropriate for usage in real-world scenarios in the healthcare domain, e.g. very small neural networks are considered due to FE limitations in computing the relevant functions. Some more promising applications are emerging, but the current maturity level of FE schemes and protocols is limiting their generalisation capacity.

The ASCLEPIOS FE-enabled analytics solution, as part of the project's framework, aims to provide services that leverage functional encryption across a wide range of commonly used functions on healthcare data. The focus is therefore in providing mechanisms that will allow healthcare providers to perform common computations over available data to support real-world needs, e.g. computations over the medical examinations of a specific patient over time and computation of aggregate information regarding patients that share a condition or some characteristics. Therefore, FE is leveraged to perform statistical computations adaptable to the data and operations of healthcare providers.

In this respect, ASCLEPIOS follows a dual approach to provide flexibility and allow healthcare providers to adopt the solution that best addresses their needs, e.g. based on time performance, security aspects and technical complexity etc. The first approach is based on a multi-input symmetric functional encryption scheme and the second is based on the asymmetric key setting. For the latter, ASCLEPIOS adopts state of the art schemes identified during the literature review, whereas the symmetric scheme is developed in the context of the project and is described in detail in Section 3.1.

It should be noted that both approaches are based on purely cryptographic FE techniques, i.e. the encryption is performed in the software level and is not hardware assisted (as some of the examples presented in Section 2.2.2 based on Intel SGX). However, the ASCLEPIOS framework foresees leveraging Trusted Execution Environments to further safeguard sensitive operations. TEEs are secure, integrity-protected environments, with processing memory and storage capabilities, isolated from an untrusted, Rich Execution Environment that compromises the OS and installed applications. More Information on using TEE's for cloud-based environments can be found in [63][64][65]. In this context, the ASCLEPIOS FE analytics services use a TEE for the key generation processes, as will be described.

3.1 Symmetric MIFE Scheme

This section presents the work performed to design a multi-client MIFE in the symmetric key setting, which was used as a basis for building some of the ASCLEPIOS statistical computation functions. The work is heavily influenced by the symmetric key MIFE scheme for inner products proposed in [1]. Based on this work, we constructed a symmetric key MIFE scheme for the ℓ_1 norm of an arbitrary vector space and showed that our construction can also support the multi-client model while preserving exactly the same security properties as the MIFE for inner-product in [1].

3.1.1 Preliminaries

Notation: If \mathcal{Y} is a set, we use $y \leftarrow^{\$} \mathcal{Y}$ if y is chosen uniformly at random from \mathcal{Y} . The cardinality of \mathcal{Y} is denoted by $|\mathcal{Y}|$. Vectors are denoted in bold as $\mathbf{x} = [x_1, \dots, x_n]$. The set of users is $\mathcal{U} = \{u_1, \dots, u_\ell\}$. A probabilistic polynomial time (PPT) adversary ADV is a randomised algorithm for which there exists a polynomial $p(z)$ such that for all input z , the running time of $\text{ADV}(z)$ is bounded by $p(|z|)$.

Definition 1 (Negligible Function). A function $\text{negl}(\cdot)$ is called negligible if and only if

$$\forall c \in \mathbb{N}, \exists \epsilon_0 \in \mathbb{N} \text{ such that } \forall \epsilon \geq \epsilon_0: \text{negl}(\epsilon) < \epsilon^{-c}.$$

Definition 2. (Inner Product). The inner product (or dot product) of \mathbb{Z}^n is a function $\langle \cdot, \cdot \rangle$ defined by:

$$f(x, y) = \langle \mathbf{x}, \mathbf{y} \rangle = x_1 y_1 + \dots + x_n y_n, \mathbf{x} = [x_1, \dots, x_n], \mathbf{y} = [y_1, \dots, y_n] \in \mathbb{Z}^n$$

Definition 2. (ℓ_1 norm). The ℓ_1 norm of \mathbb{Z}^n is a function $\|\cdot\|_1$ defined by:

$$f(x, y) = \|\mathbf{x}\|_1 = x_1 + \dots + x_n, \mathbf{x} = [x_1, \dots, x_n] \in \mathbb{Z}^n$$

From definition 1 and 2 it follows directly that if $\mathbf{x} = [x_1, \dots, x_n] \in \mathbb{Z}^n$ and $\mathbf{y} = [1, \dots, 1] \in \mathbb{Z}^n$, then $\langle \mathbf{x}, \mathbf{y} \rangle = \sum_{i=1}^n x_i = x_1 \cdot 1 + \dots + x_n \cdot 1 = \|\mathbf{x}\|_1$.

We now define MIFE in the symmetric key setting. Note that while this definition suits the single-client model, it is inadequate for a multi-client setup.

Definition 3. (Multi-Input Functional Encryption in the Symmetric Key Setting). Let $\mathcal{F} = \{f_1, \dots, f_n\}$ be a family of n -ary functions where each $f_i: \mathbb{Z}^n \rightarrow \mathbb{Z}$. A multi-input functional encryption scheme for \mathcal{F} consists of the following algorithms:

- **Setup(1^λ):** Takes as Input a security parameter λ and outputs a secret key $\mathbf{K} = [k_1, \dots, k_n] \in \mathbb{Z}^n$.
- **Enc(\mathbf{K}, i, x_i):** Takes as input \mathbf{K} , an index $i \in [n]$ and a message $x_i \in \mathcal{X}$ and outputs a ciphertext ct_i .
- **KeyGen(\mathbf{K}):** Takes as input \mathbf{K} and outputs a functional decryption key FK .
- **Dec(FK, ct_1, \dots, ct_n):** Takes as input a decryption key FK for a function f_i and n ciphertexts and outputs a value $y \in \mathcal{Y}$.

For the needs of our work, we borrow the one-adaptive (one-AD) and one-selective (one-SEL) security definitions from [1] that were first formalised in [66]. Informally, in the one-AD-IND security game, the adversary ADV receives the encryption key of the MIFE scheme and then adaptively queries the corresponding oracle for functional decryption keys of her choice. Furthermore, ADV outputs two messages x_0 and x_1 to the encryption oracle who flips a random coin and outputs an encryption of $x_\beta, \beta \in \{0, 1\}$. If the functional keys are associated the functions that do not distinguish between the messages (i.e. $f(x_0) = f(x_1)$) then ADV should *not* be able to distinguish between the encryption of x_0 and x_1 . In the case of the one-SEL-IND security, the game is identical to the one-AD-IND case, with the only difference being that ADV needs to decide on the x_0 and x_1 messages before seeing the encryption key. The “one” in both security games determines that the encryption oracle can only be queried once for each slot i (i.e. the adversary is not allowed to issue multiple queries to the encryption oracle for the same x_i).



D2.3 GDPR-compliant and Privacy-Preserving Analytics for Healthcare Providers

Definition 4. (one-AD-IND secure MIFE). For every MIFE scheme for \mathcal{F} , every PPT adversary ADV , and every security parameter $\lambda \in \mathbb{N}$ we define the following experiment for $\beta \in \{0, 1\}$:

one-AD-IND $_{\beta}^{MIFE}(1^{\lambda}, ADV)$:
 $K \leftarrow \text{Setup}(1^{\lambda})$;
 $\alpha \leftarrow ADV^{KeyGen(K), Enc(\cdot, \cdot)}$;
Output α

Where $Enc(\cdot, \cdot)$ is an oracle that on input (i, x_i^0, x_i^1) flips a random coin β and outputs $Enc(K, i, x_i^{\beta})$, $\beta \in \{0, 1\}$. Moreover, ADV is restricted to only make queries to the $KeyGen$ oracle satisfying $f(x_1^0, \dots, x_n^0) = f(x_1^1, \dots, x_n^1)$. A MIFE scheme is said to be one-AD-IND secure if for all PPT adversaries ADV , their advantage is negligible in λ where the advantage is defined as:

$$\begin{aligned} adv^{one-AD-IND}(\lambda, ADV) &= |\Pr[\text{one-AD-IND}_0^{MIFE}(1^{\lambda}, ADV) = 1] \\ &\quad - \Pr[\text{one-AD-IND}_1^{MIFE}(1^{\lambda}, ADV) = 1]| \end{aligned}$$

Definition 5. (one-SEL-IND secure MIFE). For every MIFE scheme for \mathcal{F} , every PPT adversary ADV , and every security parameter $\lambda \in \mathbb{N}$ we define the following experiment for $\beta \in \{0, 1\}$:

one-SEL-IND $_{\beta}^{MIFE}(1^{\lambda}, ADV)$:
 $\{x_i^b\}_{i \in \mathbb{N}, b \in \{0, 1\}} \leftarrow ADV(1^{\lambda}, f_i)$;
 $K \leftarrow \text{Setup}(1^{\lambda})$;
 $\alpha \leftarrow ADV^{KeyGen(K)}(\{ct_i\})$;
Output α

ADV is restricted to only make queries to the $KeyGen$ oracle satisfying $f(x_1^0, \dots, x_n^0) = f(x_1^1, \dots, x_n^1)$. A MIFE scheme is said to be one-AD-IND secure if for all PPT adversaries ADV , their advantage is negligible in λ where the advantage is defined as:

$$\begin{aligned} adv^{one-SEL-IND}(\lambda, ADV) &= |\Pr[\text{one-SEL-IND}_0^{MIFE}(1^{\lambda}, ADV) = 1] \\ &\quad - \Pr[\text{one-SEL-IND}_1^{MIFE}(1^{\lambda}, ADV) = 1]| \end{aligned}$$

3.1.2 Multi-Input Functional Encryption for the ℓ_1 Norm

In this section, we first present a MIFE scheme for the ℓ_1 norm in the single-client model and then we transform our initial construction to the multi-client model. In particular, we show how the one-AD-IND-secure MIFE scheme for inner-products from [1], can be transformed to a one-AD-IND-secure MIFE scheme for the ℓ_1 norm ($MIFE_{\ell_1}$), while preserving exactly the same security properties. Then, we show how we can transform our construction from the single-client model to the multi-client one. For purposes of completeness, we briefly recall the one-AD-IND-secure MIFE scheme for inner-products in Figure 1. The security of both MIFE schemes (inner products and ℓ_1 norm), is derived from the fact that they behave as the functional encryption equivalent of the one-time-pad. Note that, just like in the case of the one-time-pad, to achieve perfect secrecy, we require that $|k_i| \geq |x_i|$, where k_i is the encryption key and x_i , the message to be encrypted.

$\text{Setup}(1^\lambda) :$ $\forall i \in [n], k_i \xleftarrow{\$} \mathbb{Z}$ $\text{Return } \mathbf{K} = \{k_1, \dots, k_n\} \in \mathbb{Z}^n$ $\text{Enc}(\mathbf{K}, i, x_i) :$ $\text{Return } ct_i = x_i + k_i$	$\text{KeyGen}(\mathbf{K}, y_1 \dots y_n) :$ $\text{Return } \text{FK} = \sum_{i \in [n]} \langle k_i, y_i \rangle$ $\text{Dec}(\text{FK}, ct_1, \dots, ct_n) :$ $\text{Return } \sum_{i=1}^n \langle ct_i, y_i \rangle - \text{FK}$
--	---

Figure 2: one-AD-IND secure MIFE for Inner Products

In the scheme of , y fixing \mathbf{y} to be $\mathbf{y} = [1, \dots, 1]$ we compute $\langle \mathbf{x}, \mathbf{1} \rangle = \|\mathbf{x}\|_1$, for $\mathbf{x} \in \mathbb{Z}$. By doing do, we manage to transform the original inner products MIFE to a new construct that successfully computes the ℓ_1 norm. Our construction is illustrated in Figure 3.

$\text{Setup}(1^\lambda) :$ $\forall i \in [n], k_i \xleftarrow{\$} \mathbb{Z}$ $\text{Return } \mathbf{K} = [k_1, \dots, k_n] \in \mathbb{Z}^n$ $\text{Enc}(\mathbf{K}, i, x_i) :$ $\text{Return } ct_i = x_i + k_i$	$\text{KeyGen}(\mathbf{K}) :$ $\text{Return } \text{FK} = \ \mathbf{K}\ _1 = \sum_i^n k_i$ $\text{Dec}(\text{FK}, ct_1, \dots, ct_n) :$ $\text{Return } \sum_{i=1}^n ct_i - \text{sk}_f$
--	---

Figure 3: one-AD-IND secure MIFE for the ℓ_1 norm

Theorem 1. *The MIFE scheme for the ℓ_1 norm (described in Figure 2) is one-AD-IND secure. That is, for all PPT adversaries ADV:*

$$\text{adv}_{\text{ADV}}^{\text{one-AD-IND}}(\lambda) = 0$$

Proof. *The proof consists of two parts. First we construct a selective distinguisher B whose advantage for the one-SEL-IND experiment is an upper bound for the advantage of any adaptive distinguisher ADV. Then, using the fact that the MIFE_{ℓ_1} behaves like the one-time-pad, we prove that the advantage of B is zero.*

For the first part of the proof we will use a complexity argument. In particular, let B be an adversary that guesses the challenge $\{x_i^b\}$ and then simulates the one-AD-IND experiment of ADV. If B successfully guesses ADV's challenge, then she can simulate ADV's view. Otherwise it outputs \perp . Hence, ADV's advantage maximises when B guesses correctly the challenge. If the input space is \mathcal{X} , then B can guess successfully with probability $|\mathcal{X}|^{-1}$. Hence:

$$\text{adv}_{\text{ADV}}^{\text{one-AD-IND}} \leq |\mathcal{X}|^{-1} \text{adv}_B^{\text{one-SEL-IND}}$$

It can be seen that if the input space $|\mathcal{X}|$ is very large. The advantage of ADV tends to zero independently of the value of $\text{adv}_B^{\text{one-SEL-IND}}$ (i.e. $|\mathcal{X}| \rightarrow \infty \Rightarrow \text{adv}_{\text{ADV}}^{\text{one-AD-IND}} \rightarrow 0$). However, we still show that no matter the cardinality of \mathcal{X} , $\text{adv}_{\text{ADV}}^{\text{one-AD-IND}} = 0$. To do so, we will prove that $\text{adv}_B^{\text{one-SEL-IND}} = 0$. This will directly imply that $\text{adv}_{\text{ADV}}^{\text{one-AD-IND}} = 0$ since

D2.3 GDPR-compliant and Privacy-Preserving Analytics for Healthcare Providers

$adv_{ADV}^{one-AD-IND} \leq adv_B^{one-SEL-IND}$. In Figure 3 we present a hybrid game that is identical to the one-SEL-IND security game. This is derived from the fact that if $u \xrightarrow{\$} \mathbb{Z}$, then $\{u_i\}$ and $\{u_i - x_i^\beta\}$ are identical distributions. Finally, it is easy to see that only information leaking about β is $\|r - x_i^\beta\|$, which is independent of β according to the definition of the security game and the restrictions of the adversary.

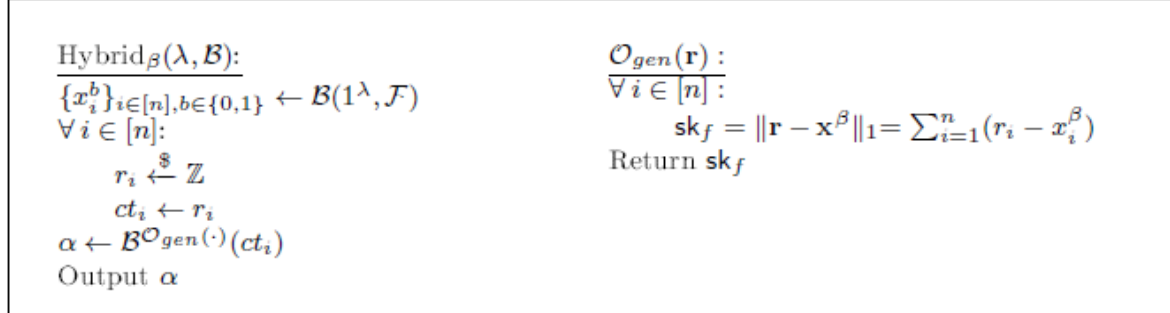


Figure 4: Hybrid Games for the proof of Theorem 1

3.1.3 From Single-Client to Multi-Client MIFE

We are now ready to describe how we can transform our single-user $MIFE_{\ell_1}$ to the multi-user MIFE for the ℓ_1 norm ($MIMIFE_{\ell_1}$). The idea is the following. Each user generates a symmetric key $k_i \in \mathbb{Z}$ which uses it to encrypt a plaintext x_i as $ct_i = k_i + x_i$. All the generated symmetric keys form a vector $\mathbf{K} = [k_1, \dots, k_n] \in \mathbb{Z}^n$, where n is the number of users. The functional decryption key FK is then $\|\mathbf{K}\|_1$ and decryption works as follows:

$$\sum_{i=1}^n ct_i - FK = \sum_{i=1}^n (k_i + x_i) - \sum_{i=1}^n k_i = \sum_{i=1}^n x_i = \|\mathbf{x}\|_1$$

A third party decryptor who would get access to the FK should only learn $\|\mathbf{x}\|_1$ and not each individual x_i . In addition to that, the users should never reveal their symmetric keys. To achieve this, we assume the existence of a trusted authority that will allow users to perform an MPC in order to jointly compute a masked version of FK without revealing each distinct k_i . Before we proceed to the actual description of our construction (Figure 5) we present a high-level overview of our system model that consists of a trusted authority (TA) and an evaluator (EV) that evaluates the value of a function f on a set of given ciphertexts

Trusted Authority (TA): TA is running in an enclave and is responsible for generating and distributing a unique random number s_i to each user u_i . The users will use the received random values to mask their symmetric keys. By doing so, and considering the fact that TA is running in an enclave and thus it is trusted, they will be able to jointly compute a masked version of the functional decryption key FK which will be used by the evaluator to calculate FK.

Evaluator (EV): EV is responsible for collecting all users' ciphertexts $\{ct_1, \dots, ct_n\}$, generating the functional decryption key FK based on the masked value that will receive from the users, and finally, calculate $f(x_1, \dots, x_n)$ without getting any valuable information about the individual values x_i

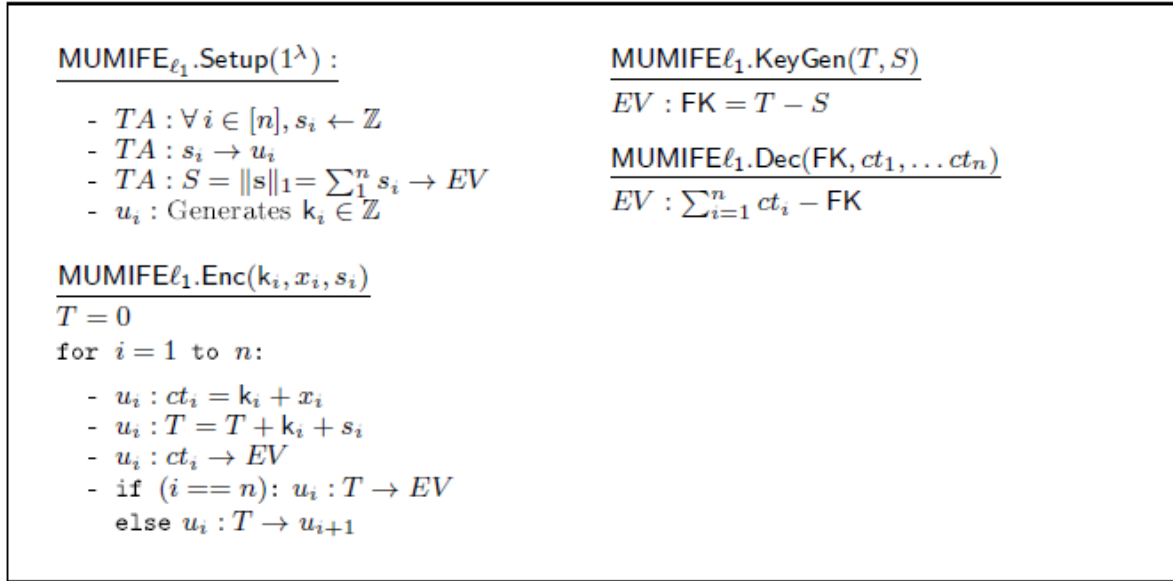


Figure 5: Multi Input MIFE for the ℓ_1 norm

Correctness: The correctness of the $MUMIFE_{\ell_1}$ scheme presented in Figure 4 follows directly since:

$$\begin{aligned}
 \sum_{i=1}^n ct_i - FK &= \sum_{i=1}^n ct_i - T + S = \sum_{i=1}^n k_i + \sum_{i=1}^n x_i - \left(\sum_{i=1}^n (k_i + s_i) \right) + \sum_{i=1}^n s_i \\
 &= \sum_{i=1}^n x_i \\
 &= \|x\|_1
 \end{aligned}$$

Theorem 2. *The Multi-User Multi-Input Functional Encryption scheme for the ℓ_1 norm (described in Figure 5) is one-AD-IND-secure. That is, for all PPT adversaries ADV:*

$$adv_{ADV}^{one-AD-IND}(\lambda) = 0$$

Proof. *The proof is omitted since it is a direct result from Theorem 1. This can be seen by the fact that the Encryption and KeyGen Oracles are identical to the ones described in Figure 3. The only difference is that in the case of $MUMIFE_{\ell_1}$, the **Setup** algorithm is executed by multiple users instead of one, since each user generates a distinct unique key. Without loss of generality we can assume that this is exactly the same procedure since in the case of $MIFE_{\ell_1}$, one user sample n random numbers from \mathbb{Z} resulting to a vector $\mathbf{K} = [k_1, \dots, k_n]$, and in the case of $MUMIFE_{\ell_1}$, n users sample one random number from \mathbb{Z} resulting to a vector $\mathbf{K}' = [k'_1, \dots, k'_n]$. However, the distributions $\{k_i\}$ and $\{k'_i\}$ are identical and thus we conclude that we can use exactly the same Hybrid game as the one in Figure 4.*

3.2 FE Services Implementation & Workflows

As explained, the scope of the ASCLEPIOS FE Analytics is to provide a set of services that can be used by healthcare providers to perform statistical computations over numeric data. The two provided approaches, i.e. the symmetric and asymmetric encryption, have inherent differences which are reflected in the way the services of the two types are used and will be shown in the sequence diagrams provided in subsequent sections. However, the main entities that are involved in the application of the FE services are the same, as follows:

D2.3 GDPR-compliant and Privacy-Preserving Analytics for Healthcare Providers

- **Trusted Authority:** The Trusted Authority runs inside a Trusted Execution Environment and is responsible the generation and retrieval of the keys.
- **Evaluator:** The evaluator is responsible for calculating the function result using the ciphertext(s) and the functional encryption key.
- **API Server:** The API server is responsible for handling the HTTP requests through which the analytics services are invoked. It is also responsible for handling requests for the encryption of the data when these are invoked by the users a priori, i.e. when users initiate the encryption of their data to make them available to functional encryption services.
- **User:** The Users are the stakeholders that provide the data. Depending on the nature of the data, where they reside and who performs their encryption (and upload), the user may be a healthcare organisation, a physician, a patient or any other ASCLEPIOS user. The authorisation and authentication processes are handled on the framework level and therefore these aspects are not considered here.
- **Analyst:** The analyst is, similar to the User entity, an authorised and authenticated ASCLEPIOS user. In the FE Analytics workflow, the Analyst is the one invoking a particular service, i.e. the one that needs to obtain a computation on the data provided by the users.

3.2.1 Symmetric MIFE services

These services are based on the scheme presented in Section 3.1 and allow the Analyst to compute some statistics over the data coming from multiple users, including sum, average and weighted average (where the weights can be considered part of the applied function). Figure 6 shows an indicative workflow of an FE service that computes the sum of two values provided by two different users.

The workflow is initiated by the Analyst who invokes the service that applies the sum function of the symmetric MIFE service for two users, each of which provides a specific value. However, this is only an abstraction to facilitate the presentation of the workflow. As described in D2.1, data from healthcare providers are in JSON format and therefore there are numerous and diverse values potentially available in the computations, especially when considering that a user as explained may be a healthcare provider. The invoked function therefore handles the required data selection process, e.g. a different function will be called to compute the number of days that patients with a certain condition were hospitalised as a whole (wherein each input in the FE scheme corresponds to a patient and the value represents number of days) vs to compute the amount of antibiotics prescribed to a patient within a year (wherein each input corresponds to a patient's visit to a physician and the value represents the number of prescribed antibiotics in this visit).

The encryption, decryption and key generation functionalities are implemented in C++. Flask⁷, a lightweight WSGI web application framework, is used to implement the RESTful API. The TEE used for the Trusted Authority is Intel SGX.

⁷ <https://flask.palletsprojects.com/en/1.1.x/>

Asclepios Symmetric FE : Sum of values from 2 users

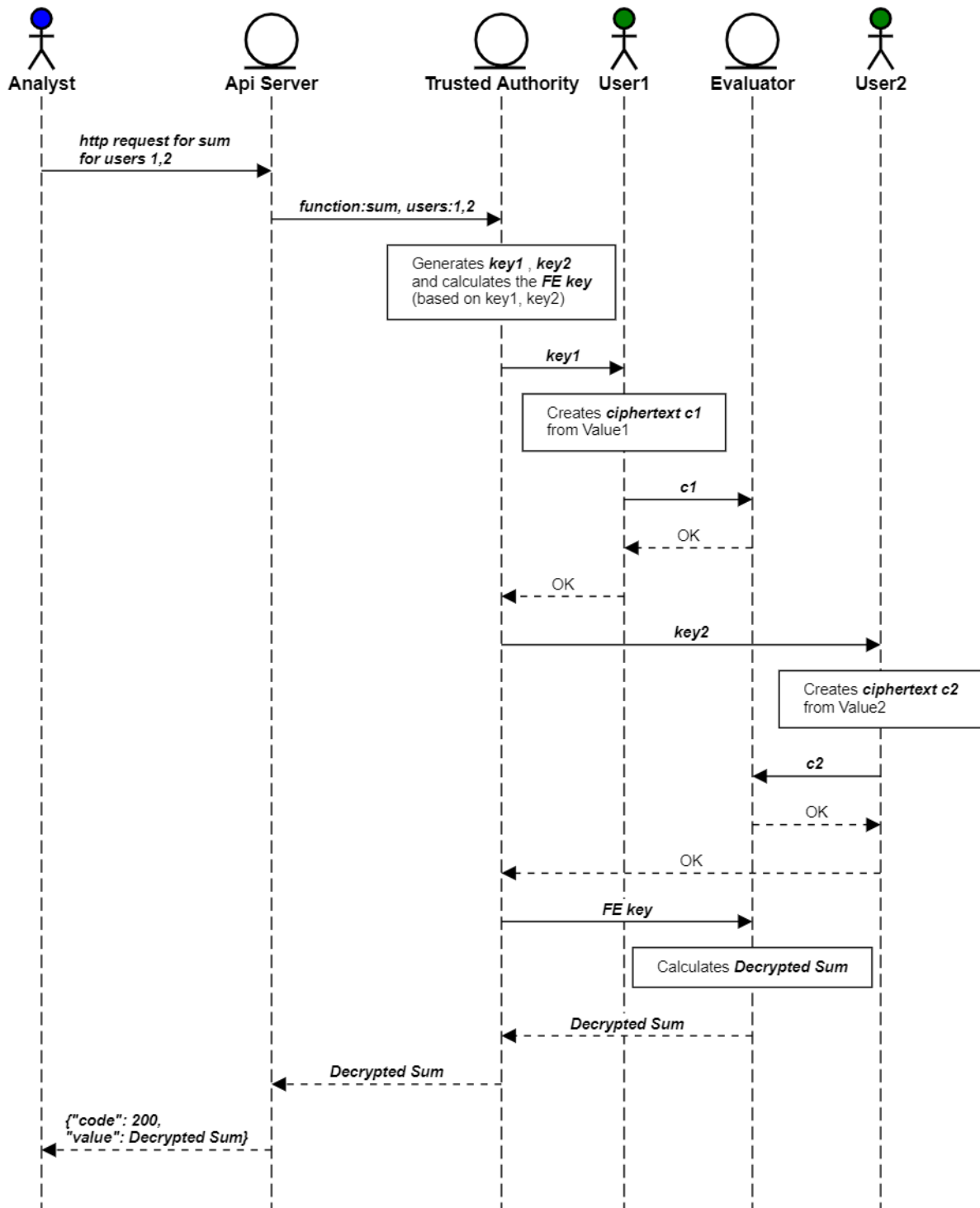


Figure 6: Symmetric MIFE example

3.2.2 Asymmetric IPFE services

The asymmetric inner product FE services are based on schemes found during the literature review, specifically the ones provided in [4] and [15]. The implementation of the key generation, encryption and decryption algorithms is based on the open source CiFEr library⁸.

⁸ <https://github.com/fentec-project/CiFEr>

D2.3 GDPR-compliant and Privacy-Preserving Analytics for Healthcare Providers

As in the case of the symmetric MIFE services, C++ and Flask are used to implement the functionalities provided by these services as well. The TEE used for the Trusted Authority is again Intel SGX.

Services in this category also provide basic statistical computations, the main differences in scope being that they target cases where all required input data come from the same user (i.e. data provider) and data can be diverse, e.g. an indicative (simplistic) usage could be to compute the probability of a heart attack by combining the age, weight and cholesterol levels of the patient based on a linear regression formula (the formula is considered part of the function). Two slightly different workflows are foreseen for these services.

The first one is similar to the one for the symmetric example and is initiated by an Analyst invoking a function to get the results of a statistical computation (denoted as func1), as shown in Figure 7. Assuming func1 refers to the heart attack risk calculation example mentioned above, vector Y that is shown in the diagram corresponds to the regression factors for each of the input variables that are provided through vector X.

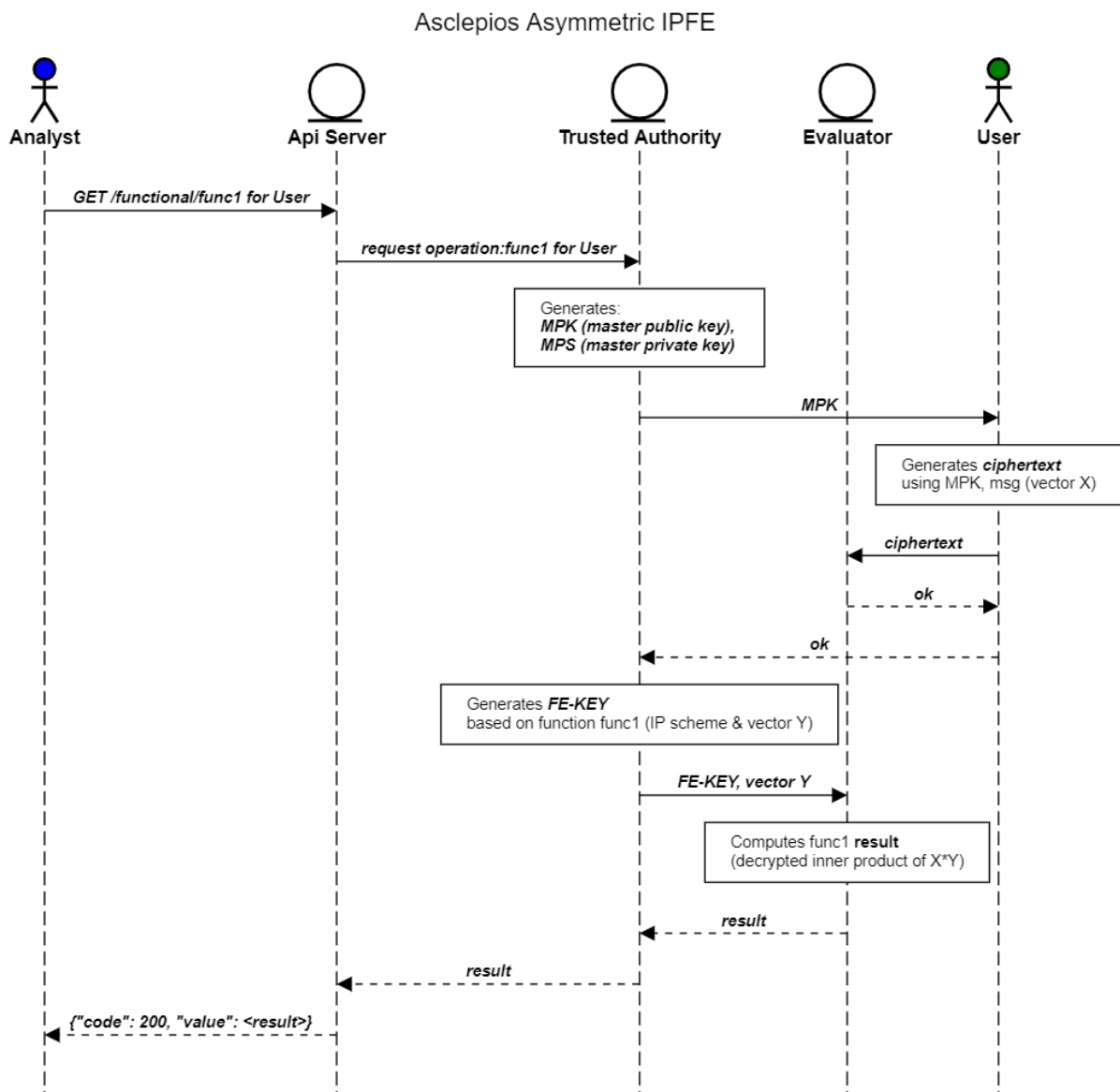


Figure 7: Asymmetric IPFE example

The presented example in Figure 7 assumes that the Trusted Authority generates the master public and master secret keys as part of the workflow. However, if the specific User has

D2.3 GDPR-compliant and Privacy-Preserving Analytics for Healthcare Providers

previously acquired a master public key for this scheme, then the same key can be used again to encrypt the new message (vector X).

Furthermore, it is possible that the encryption and decryption processes are, to an extent, detached, in the sense that it is possible for a User to choose to make their data available for statistical computations through FE services, prior to an Analyst requesting to use them in a computation. In this case, the evaluator, which runs on the cloud, retrieves the encrypted data every time they are needed for a computation. This process is shown in Figure 8.

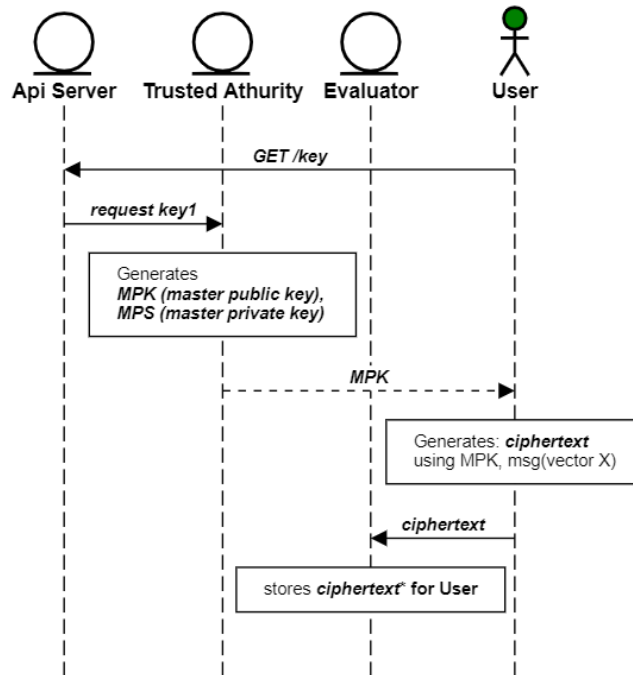


Figure 8: Asymmetric IPFE – Encrypt Only

The data are then available in the cloud in an encrypted form and can be used whenever the FE service is invoked, i.e. by any Analyst and for any function implemented using this scheme. The process is shown in Figure 9.

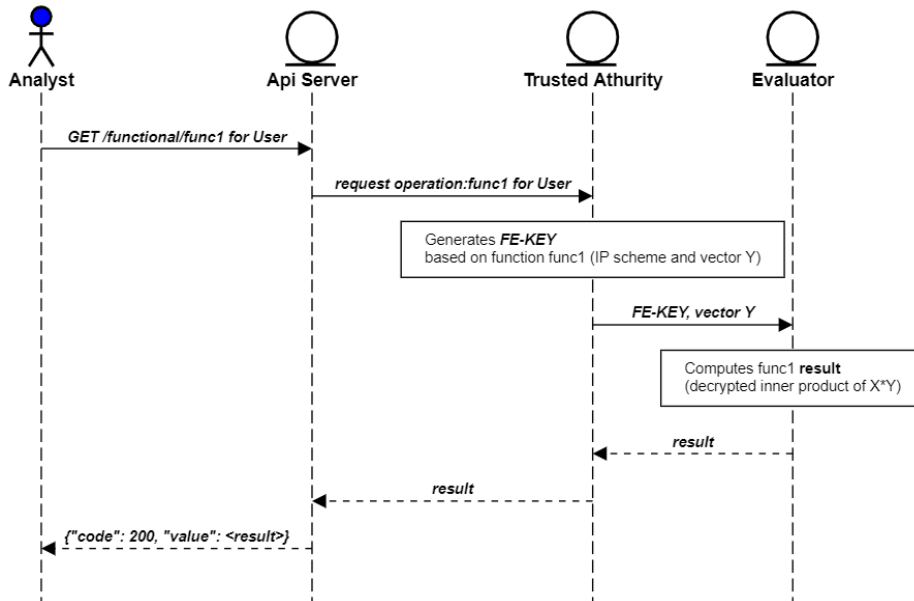


Figure 9: Asymmetric IPFE – Invoke Function Only

3.3 FE Analytics Usage in ASLEPIOS

Section 3.2 described the functionalities of the FE Analytics services, the schemes that are used in the background and the workflows that are supported. As shown also in the provided sequence diagrams, the services can be invoked through a simple RESTful API, which provides endpoints for the various supported schemes and functions.

In the case of the asymmetric MIFE scheme, depicted in Figure 6, invoking a function through the API will handle in the background all required interactions and processes in order to acquire the values from all users and return the result to the requestor (Analyst). In the case of the symmetric IPFE functions, there are two different ways of invoking the services: The first works similar to the symmetric one and exposes endpoints that when invoked will handle all processes and interactions, including the communication with the user to acquire the encrypted data (Figure 7). The second way assumes encryption of the data happens before the service being invoked by an Analyst and therefore endpoints are exposed for requesting to encrypt data and for requesting to apply a function on data that are expected to be already available (although this can be easily combined with the first approach in order to get the data that are already available and also request those that are not), as shown in Figure 8 and Figure 9.

In order to provide a more complete solution, the FE Analytics are expected to be used also combined with other ASCLEPIOS tools and services, therefore during design and implementation potential workflows in this context were also considered. As an example, although in the stand alone version the service invocation specifies on which users the computation should be applied, FE Analytics can be combined with the SSE scheme to enable the computation of statistics not on explicitly provided users, but on users and data assets that match the selected search criteria. Thus, the SSE service can feed the FE services with the data on which the computation should be applied, either directly or indirectly by providing the index in the database so as to retrieve them. Furthermore, the encryption process could be unified from a user perspective (in the background the data will be encrypted separately for each service), so that the user could select which data should be uploaded and for which service through the same process. Combining the ASCLEPIOS SSE with FE services will also enable pipelined computations applied on data that meet user-defined criteria, e.g. assuming a function F1 that instantiates the ASCLEPIOS FE service for the evaluation of a regression formula that computes a patient's risk for a certain condition based on specific attributes (fields) from the patient's EHR and a function F2 that corresponds to an FE service that computes the average, it will be possible for an Analyst to define a query that calculates the "average risk score to develop condition C for patients that meet characteristics A and B", where characteristics could be for example age thresholds.

3.4 GDPR-considerations

The present section examines GDPR compliance perspectives of the ASCLEPIOS functional encryption services. In order to provide a legal analysis in this direction, the scope of analytics operations that should be considered needs to be first defined. It should be noted that although the implemented services provide statistical computations, the analysis is extended to reflect on machine learning models as well for completeness, since the implemented inner product functionality could be adapted to implement regression functions. Within the context of ASCLEPIOS, statistical functions and machine learning methods are applied specifically to encrypted data and leveraging isolated environments. Therefore, only encrypted data and operations to retrieve these statistics or train machine learning models are considered. How the resulting statistics or models are later used, such as to make predictions for specific patients using auxiliary data, is considered to be outside of the scope. The second aspect that is investigated is the extent to which the data that are used in the analytics operations should be classified as personal data. It seems clear that the underlying

D2.3 GDPR-compliant and Privacy-Preserving Analytics for Healthcare Providers

healthcare data that are used in the analytics are indeed personal data under the definition of the GDPR, because they relate to identifiable natural persons. Whether that is also the case for the data in its processed (encrypted) form and the resulting statistics and models is harder to answer and depends on specific circumstances. The question of whether a natural person can be reasonably identified (in)directly through these data needs to be answered to provide some insights in this respect.

Recital 26 of the GDPR goes into detail on how this question should be answered: "To determine whether a natural person is identifiable, account should be taken of all the means reasonably likely to be used, such as singling out, either by the controller or by another person to identify the natural person directly or indirectly. To ascertain whether means are reasonably likely to be used to identify the natural person, account should be taken of all objective factors, such as the costs of and the amount of time required for identification, taking into consideration the available technology at the time of the processing and technological developments." This first sentence is more or less unchanged compared to the Privacy Directive (Directive 95/46/EC). In *Breyer v. Germany* (ECLI:EU:C:2016:779), the CJEU ruled that information should be classified as personal data even when data from third parties is required to identify data subjects. Specifically, it deemed the assistance of a competent authority in identifying data subjects a reasonable means, unless this is prohibited by law or requires a disproportionate effort. From this we can conclude that the healthcare data in its encrypted form likely still qualifies as personal data when a third party possesses the means to decrypt or otherwise identify natural persons from this data. With respect to the resulting statistics and models, it is probably necessary to apply the mentioned reasonability test based on a case-by-case basis. [60] argues that models may be subject to model inversion and member inference attacks, which leak information about the training data of a model. In these cases, models might indeed be personal data. More generally, [61] point out that it is necessary to consider not only the information alone, but also its environment. Indeed, the availability of (linkable) auxiliary data, the governance of the data (who has access and under what conditions) and possible motivations for re-identification are all factors in whether re-identification is reasonable.

The third aspect that is examined refers to whether the exercise of data subject rights should be facilitated and if so, in what way. In the absence of specific exceptions, such as under a) or b) below, the general answer is affirmative. How exactly these should be met differs by each right, which we will discuss separately.

- a) Does the exception of article 11(2) apply? The data subject rights do not apply when the controller is not in a position to identify the data subject. However, data subjects may still provide additional information to enable their identification, in which case the controller is still obliged to act on the request.
- b) Do the optional derogations in article 89(2) and (3) apply? European Union and Member State law may give derogations from some of the data subject rights for data that are processed for scientific or historical research purposes or statistical purposes or archiving purposes in the public interest, when those rights are contrary to those purposes. There does not exist a universal answer, because it depends on the specific jurisdiction of the processing.
 - i. Does the processing by ASCLEPIOS fall under the purposes mentioned in article 89? It is likely that some of the processing qualifies as being done for scientific research purposes. Recital 159 states: "For the purposes of this Regulation, the processing of personal data for scientific research purposes should be interpreted in a broad manner including for example technological development and demonstration, fundamental research, applied research and privately funded research. [...] Scientific research purposes should also include studies conducted in the public interest in the area of public health." However, this depends on the specific case; When the processing is done outside of an academic context, for example in the treatment of a specific patient, this article likely does not apply.

D2.3 GDPR-compliant and Privacy-Preserving Analytics for Healthcare Providers

- c) How should the controller act on erasure requests? We note that the right to obtain erasure ("right to be forgotten") is by no means absolute. We will discuss a number of grounds on which such request could reasonably be rejected within our application. If erasure is warranted, this may take a significant effort depending on the technical details. According to [60], when a machine learning model itself qualifies as personal data as previously discussed, either the model must be retrained on the amended data (which often carries a great computational cost) or the model must be amended after training (something that is still being explored and currently cannot be done on existing systems).
- i. Does the subject have a valid grounds to demand erasure, as in 17(1)? This is highly case-specific and depends on lawfulness and legitimate grounds for the processing. We point out that in practice, processing is often based on consent (article 6(1)) even when this would not be the only applicable legal ground. In those cases, the controller is not obliged to erase the personal data when another legal ground exists.
 - ii. Does the controller have a valid overriding exception as in 17(3)? Paragraph 3 provides that the right to erasure does not apply when processing is necessary for the purposes mentioned in article 89. As we mentioned in b-i), some processing within the ASCLEPIOS framework may indeed qualify as being done for those purposes. It remains to show that processing of the given personal data is in fact **necessary** for those purposes. It can be argued that in some cases the erasure of data does not disproportionately affect those purposes, especially when the amount of data to be erased is insignificant compared to the full data set and there are no strong requirements on the completeness of the data. However, this is merely an interpretation.
- d) How should rectification be performed? The accuracy of personal data is one of the main pillars of the right to data protection. Article 16 specifies that data subjects have the absolute right to obtain rectification of inaccurate personal data. While intuitively one would say that personal health information is fact-based and therefore does not contain inaccuracies, this is not always true. Indeed, there are various reasons why data that are entered into the system might be wrong or changes due to new developments. This is supported by [62], who analysed health record amendment requests in the United States. It goes without saying that the accuracy of the data is often essential for the purposes for which they are processed. Rectification is therefore in the controller's best interests, even in the absence of a specific legal obligation. Rectification may take significant effort, similar to erasure in (c). In cases where the personal data are incomplete, it may be sufficient to add a supplementary statement.
- e) How should restriction on processing be performed? See the considerations under c) and d)

The final aspect that needs to be examined is whether the analytics operations implement appropriate security measures, as described in art. 25 and art. 32. This is a highly complex question that can only be answered after a full assessment of the environment/implementation and cannot be therefore examined in the scope of the functional encryption mechanisms presented in this deliverable. We point out that the ASCLEPIOS project contributes technical measures to ensure that only authorised people have access to personal data and that analytics users only learn the result of their analytics operation and nothing else. Therefore, it indeed implements certain data-protection principles. That is not to say that the ASCLEPIOS framework by itself presents an appropriate level of security, as this would require an assessment of the risk associated with the specific processing compared to the feasibility of additional measures. In addition, we point out that some fundamental principles, such as data minimisation at time of collection and retention period, are outside of the scope of ASCLEPIOS as they are heavily dependent on the application.

4 Cybersecurity Landscape in Healthcare

The second axis defining the work presented in the current deliverable, as explained in Section 1.1, is related to Task T2.4 activities, and evolves around identifying a set of analytics used by healthcare providers in their methods and efforts to secure their infrastructures and the data and information owned or exchanged, particularly in the context of leveraging CSP operations. The need for a comprehensive cybersecurity strategy spanning from detection and mitigation to prevention of security threats becomes even more evident when considering the great number of assets at stake and the risks from a data breach of patient information or the interruption of normal operations.

The present section provides background information regarding the threats against which healthcare providers need to secure their infrastructures and the challenges they need to address in this effort. Furthermore, methodological frameworks and technical solutions that help towards this direction are briefly presented, highlighting the particular needs and complexities that arise in the detection of abnormal behaviours which could indicate malicious activities but also benign yet inappropriate usage that exposes the system to cyber threats. Finally, data analytics and visualisation methods that can be employed to provide actionable insights in the identification and mitigation of suspicious (data) access behaviours are presented.

4.1 Cyber Threats in Healthcare

A cybersecurity threat is any circumstance or event with the potential to adversely impact an asset through unauthorised access, destruction, disclosure, modification of data and/or denial of service⁹. ENISA reports present the vast cyberthreat landscape, which includes Malware, Web Based Attacks, Web Application Attacks, Phishing, Denial of Service, Spam, Botnets, Data Breaches, Insider Threats, Physical manipulation/ damage/ theft /loss, Information leakage, Identity Theft, Cryptojacking, Ransomware and Cyber Espionage [67]. The 2018 report highlighted healthcare as the leading sector in the number of incidents (27%), demonstrating incidents of vast information leakage, such as the accidental exposure of 3,5 million records due to a misconfiguration of the Amazon Simple Storage Server cloud server, which was reported by the 211 LA County.

A cybersecurity incident exposes the healthcare organisation to various serious risks, such as the loss or corruption of organisation information and patient data and health records, the breach of personal information, the disruption of healthcare services, the access and control of medical infrastructure by malicious actors, damage of the organisation's reputation and financial costs [68]. The detrimental impact of such incidents combined with the fundamental vulnerabilities healthcare organisations present due to the nature of the domain -requiring the provision of services under emergency situations-, the inadequate preparedness of infrastructure, and the lack of security awareness, make them a preferable target for malicious actors. However, cybersecurity threats in the healthcare are not always a result of malicious actions, but also emerge from human-errors, system and third-party failures, extending the scope of security measures and practices that shall be considered by the security departments of the organisations. According to [69] and [70] the five most common cybersecurity threats affecting healthcare organisations are the following:

- **E-mail Phishing Attack:** Phishing is a social engineering technique, using appropriately crafted messages to lure the recipients into opening a malicious attachment, click on an unsafe URL or hand over their credentials. E-mail phishing exploits emails as the medium for the execution of the attack. Attackers take advantage of leaked and hacked personal data to compose convincing, targeted messages and increase the success of their campaigns. One of the most notable e-mail phishing attacks targeting healthcare in

⁹ Definition by ENISA: <https://www.enisa.europa.eu/topics/threat-risk-management/risk-management/current-risk/risk-management-inventory/glossary> (last access 18/05/2020)

D2.3 GDPR-compliant and Privacy-Preserving Analytics for Healthcare Providers

2019, was conducted against Montpellier university medical centre¹⁰, where an infected email opened by an employee infected more than 600 computers, but was suspended from spreading to all of its 6.000 machines by the internal network infrastructure of the hospital.

- **Ransomware Attack:** In a ransomware attack, the attacker gains ownership of the files and/or devices and deprives the legitimate owner of access, demanding a ransom to restore back ownership. This type of attack is usually conducted with the use of malicious software (malware) that encrypts the data of the affected system, and the real owner can only get hold of the decryption key by paying the ransom usually in a cryptocurrency. Phishing emails are used as a conduit for the installation of the malware at the targeted system, and take advantage of software vulnerabilities, such as unpatched software. The WannaCry¹¹ cyberattack against UK's National Health Service in 2017 is one of the most eminent ransomware attacks, exploiting a Windows vulnerability to infect 200.000 computers and led to the cancellation of approximately 20.000 appointments, costing the NHS almost £92m for the restoration, cleanup and upgrade of its IT systems.
- **Loss or theft of equipment or data:** An everyday phenomenon, such as the loss or theft of paperwork or of a device with unencrypted data, can lead to data breaches and malicious actors getting hold of sensitive data. Additionally, the loss of equipment could impose disruption of operations until it is restored, while the recovery of lost data is not always feasible if an appropriate backup policy were not applied until the time of loss.
- **Insider Threat:** The accidental or intentional abuse of access to the organisation's assets by current or former employees, partners or contractors is a viable security threat for all organisations. Three are the most common types of insider threats: a malicious insider who intentionally discloses information for personal benefit or to inflict harm, a negligent insider who exposes data by mistake or by not following the security policies and instructions and the compromised insider who acts unintentionally as a medium for an attacker. An example of a data breach due to human error is the case of Independence Blue Cross¹² in 2018, where an employee accidentally posted publicly a file containing personal and medical information of almost 17.000 patients. The accident went undetected for two months, giving the opportunity to unauthorised users to view its contents.
- **Attacks against connected medical devices:** The hijacking of IT-based medical equipment can lead to leakage of sensitive patient's data but even to imposing a direct threat for the patient's life through the manipulation of the compromised device's operation (e.g. power off, continuous reboot etc.). A recent vulnerability discovered in two models of hospital anaesthesia machines manufactured by General Electric, and exposed by CyberMDX¹³ is potentially life-threatening for patients, as an attacker could gain remote control and tamper with settings such as the concentration of oxygen, silence device alarms and modify logs.

Other cyber threats faced by healthcare organisations, and more specifically smart hospitals, are included in the wider threat taxonomy by ENISA [71] which enlists additional malicious actions, such as the Denial of Service attacks (ex. the Boston Children Hospital attack¹⁴ in 2019), system failures (for example the unavailability of data due to overload), supply chain failures due to unavailability of a cloud service or network provider and natural phenomena (Figure 10).

¹⁰ <https://www.stormshield.com/news/top-5-cyberattacks-against-the-health-care-industry/>

¹¹ <https://www.telegraph.co.uk/technology/2018/10/11/wannacry-cyber-attack-cost-nhs-92m-19000-appointments-cancelled/>

¹² <https://www.fiercehealthcare.com/payer/independence-blue-cross-data-breach-cybersecurity-privacy>

¹³ <https://www.zdnet.com/article/vulnerabilities-found-in-ge-anaesthesia-machines/>

¹⁴ <https://thethreatreport.com/story-behind-the-ddos-attack-vs-boston-children-hospital/>



Figure 10: Threat taxonomy for smart hospitals by ENISA [71]

Particular attention should be given to a special type of attacks that have emerged during the last years; the new age attacks [72]. These attacks differentiate from traditional attacks in the attitude of the attackers after they gain the desired control: whereas in traditional attacks the attackers would reveal their accomplishment in order to claim a short-term reward (like ransom), in new-age attacks the intention is to remain undetected for as long as possible and gather data. Usual targets of this kind of attacks are IT systems and organisations of high complexity with critical data; thus, hospitals are among the most critical infrastructures to be affected by a new age attack.

Following is a classification of the new threats as suggested in [72]:

- **Advanced Persistent Threats (APT):** Advanced persistent threats are performed in multiple stages (reconnaissance, foothold establishment, lateral movement, exfiltration/impediment, post-exfiltration/post-impediment [73]). They are sophisticated attacks, focusing at gaining access to the system and remaining undetected for a long period in order to steal high-value data.
- **Polymorphic threats:** Polymorphic threats, notably called the enemy of signature-based cyber defence systems, are attacks such as worms, Trojans and virus, that constantly change attributes (e.g. filename, compression), making their detection extremely difficult.
- **Zero-day threats:** These threats exploit previously publicly unknown vulnerabilities of software and systems. These vulnerabilities are not always unknown to the vendors, yet they are not disseminated prior to patching them, for security purposes. Zero-day attacks can remain undetected for a long period and the exploited system vulnerabilities can be a challenge to fix.
- **Composite threats:** This type of attacks combines two attack approaches. On the one hand are the syntactic attacks that exploit technical software/hardware vulnerabilities, while the semantic attacks utilise social vulnerabilities. The composite attacks come as a combination of these two.

D2.3 GDPR-compliant and Privacy-Preserving Analytics for Healthcare Providers

The Healthcare Information and Management Systems Society [74] has composed a list with the initial points of compromise of the most significant security incidents identified by 166 qualified information security leaders from a number of healthcare organisations that participated in their Cybersecurity Survey for 2019. The respondents identified the e-mail as the most common point of compromise with 59%, with human error in the second place with 25%. This result showcases the efficiency of email phishing campaigns, which makes them an extremely popular means for inexpensive and efficient cyberattacks. A second observation is the key role of the human factor, which is underlined as an important point of compromise, either by being the victim of a phishing attack or by openly exposing information by mistake or on purpose. Other identified entry-points for attackers are the compromised vendors, consultants, clients or other parties (Figure 11).

Initial Point of Compromise	2019					Total	2018 Total
	Hospital	Non-Acute	Vendor	Other	Total		
E-mail (e.g., phishing e-mail)	69%	56%	35%	68%	59%	70%	
Human error	30%	25%	21%	16%	25%	-	
Compromise of vendor, consultant, client, or other party	20%	3%	0%	8%	10%	3%	
Hardware or software infected with malware “off the shelf” (e.g., pre-loaded malicious software)	7%	8%	9%	8%	8%	3%	
Compromise of mobile device (e.g., malware infection or otherwise)	11%	0%	12%	0%	7%	3%	
Compromise of our organization’s website and/or web server (e.g., SQL injection, XSS, etc.)	6%	3%	9%	4%	5%	4%	
Compromise of remote access server (e.g., RDP, VNC, remote access gateway, etc.)	3%	8%	9%	4%	5%	-	
Compromise of medical device (e.g., malware infection or otherwise)	10%	0%	6%	0%	5%	3%	
Compromise of third party website (not a vendor, consultant, or third party partner)	7%	0%	3%	8%	5%	2%	
Compromise of our cloud provider/service	3%	3%	0%	0%	2%	2%	
Other	1%	3%	3%	4%	2%	10%	

Figure 11: Initial Points of Compromise for Security Incidents [74]

4.2 Cybersecurity Frameworks and Tools

Practices and guidelines have been published by standardisation bodies and cybersecurity organisations providing a framework for organisations to organise an effective cybersecurity plan. An interesting point is that specific guides have been delineated, targeting the healthcare domain and the protection of patients, due to the challenging nature of the domain as well as the criticality of the data and assets at stake. These practices attempt to cover all aspects of an organisation’s operation and infrastructure. However, their multitude and their dependence from the adoption by the involved human actors, make it difficult for them to be fully applied in practice.

The National Institute of Standards and Technology (NIST) has constructed a five-function cybersecurity structure to cover the whole cybersecurity lifecycle (Identify, Protect, Detect, Respond, Recover) describing the intended outcome of activities as a group of cybersecurity outcomes (i.e. “Categories”) which are closely knit to particular activities [71]. Examples of categories are the “Data Security” and “Protective Technology” of the “Protect” function, “Anomalies and Events”, “Security Continuous Monitoring” and “Detection Processes” for the

D2.3 GDPR-compliant and Privacy-Preserving Analytics for Healthcare Providers

“Detect” function and more. The complete NIST Cybersecurity Framework is available in Figure 12. This framework is applicable generally to organisations relying on technology, whether focusing on information technology, industrial control systems, cyber-physical systems, or connected devices and IoT.

From the enlisted categories it can be generally deduced that some of them are business-oriented, addressing the assessment of risks and targets, timely planning, responsibility identification and policy-making (ex. “Business Environment”, “Risk Management Strategy”, “Detection Processes”, “Response Planning”, “Awareness and Training” etc.), while others are associated with the application of privacy-preserving technologies, the adoption of software-enabled cybersecurity solutions and the construction of secure infrastructures (ex. “Identity Management, Authentication and Access Control”, “Data Security”, “Protective technology” etc.).

Function Unique Identifier	Function	Category Unique Identifier	Category
ID	Identify	ID.AM	Asset Management
		ID.BE	Business Environment
		ID.GV	Governance
		ID.RA	Risk Assessment
		ID.RM	Risk Management Strategy
		ID.SC	Supply Chain Risk Management
PR	Protect	PR.AC	Identity Management and Access Control
		PR.AT	Awareness and Training
		PR.DS	Data Security
		PR.IP	Information Protection Processes and Procedures
		PR.MA	Maintenance
		PR.PT	Protective Technology
DE	Detect	DE.AE	Anomalies and Events
		DE.CM	Security Continuous Monitoring
		DE.DP	Detection Processes
RS	Respond	RS.RP	Response Planning
		RS.CO	Communications
		RS.AN	Analysis
		RS.MI	Mitigation
		RS.IM	Improvements
RC	Recover	RC.RP	Recovery Planning
		RC.IM	Improvements
		RC.CO	Communications

Figure 12: NIST Cybersecurity Framework [75]

Based on the five identified prevailing cybersecurity threats (e-mail phishing attack, ransomware attack, loss/theft of equipment or data, accidental or intentional data loss, attacks against connected medical devices/ medjacking), the 405(d) Task Group singled out the following ten most effective mitigation practices, which are in the core of the holistic framework for cybersecurity in the health industry published in 2015 by NIST[69]:

1. Email Protection Systems

D2.3 GDPR-compliant and Privacy-Preserving Analytics for Healthcare Providers

2. Endpoint Protection Systems
3. Access Management
4. Data Protection and Loss Prevention
5. Asset Management
6. Network Management
7. Vulnerability Management
8. Incident Response
9. Medical Device Security
10. Cybersecurity Policies

Each practice outlines a series of steps to protect against the potential cyber threats affecting the particular system (for example, the “Email Protection Systems” practice provides steps to protect against ransomware, phishing and data leakage). Each of the ten main practices has been further broken into various sub-practices corresponding to the five main cybersecurity functions, and tailored to the needs and resources of the organisation based on its size (small healthcare organisations [77], medium and large healthcare organisations [78]). As the framework contains in total 88 practices, their implementation is an extremely demanding task, thus NIST has also prescribed a simple model to help cybersecurity experts of any healthcare organisation with the enumeration and prioritisation of practices. The model comprises the following steps:

- Step 1: Enumerate and Prioritise Threats
- Step 2: Review Practices Tailored to Mitigate Threats
- Step 3: Determine Gaps Compared to practices
- Step 4: Identify Improvement Opportunity and Implement
- Step 5: Repeat for Next Threats

In the context of the current analysis, special attention is placed on cybersecurity categories and practices involving the employment of privacy-preserving technologies and software-enabled protection technologies, and how these can be realised in the cybersecurity programs of organisations, as these are the ones more relevant to the ASCLEPIOS scope. The EU-funded Coordination and Support Action project focusing on privacy-preserving big data technologies, e-Sides¹⁵, has conducted research on the effectiveness of mechanisms used to deploy privacy-preserving technologies [79]. They have identified three main directions, encompassing the individual classes of technologies, as follows:

- **Technologies that Change Data:** Anonymisation, sanitisation and encryption are the three classes under the “Technologies that Change Data” category. In anonymisation, the personally identifiable information of a dataset is encrypted or totally removed, while sanitisation involves the encryption or removal of sensitive information. Anonymisation could be considered as a type of sanitisation. Some indicative popular methods of anonymisation and sanitisation are k-anonymity, l-diversity and differential privacy. In encryption, the data are encoded and only eligible users can acquire total or partial access to them, with the use of a decryption key. Apart from traditional cryptographic primitives, such as Public-key cryptography, Symmetric-key cryptography and digital signatures, new schemes and primitives providing functionalities for extended or refined applications have emerged and are subject of extensive research, such as attribute-based encryption, functional encryption and homomorphic encryption. Multi-party computation is another cryptographic field combining distributed processing with encryption to enable secure computations.

¹⁵ <https://e-sides.eu/e-sides-project>

D2.3 GDPR-compliant and Privacy-Preserving Analytics for Healthcare Providers

- **Technologies for Handling Data:** Access control is the restriction of users and applications from data and resources if certain conditions are not met. Attribute-based access control is a technique going beyond simple access control, by allowing the use of certain attributes for the decision on granting or denying access, such as the user role, time, IP and more. Policy enforcement applies a set of defined rules for the allowed handling of data. Accountability is the principle that guarantees the enforcement of data protection policies and is implemented through auditing and monitoring processes.
- **Technologies Empowering the User:** Transparency is the core data protection principle of explicitly informing the data subject regarding the way her data is collected and processed, while data provenance is related to being able to track the lineage of information. Both pose a challenge, especially in the era of big data and black-box machine learning, where the user can only see the results of computations over data from heterogeneous sources. Access and portability technologies enable users view their stored data and move them between service providers without losing any information. Candidate technology enablers for these purposes are consent mechanisms, privacy preferences and personal stores of data.

The integration of protective software in organisations is another measure proposed by cybersecurity frameworks. Some of the most widely used information security technologies are IDSs/IPs, UTM, SIEMs and LMSs. These are briefly presented below:

- **Intrusion Detection System / Intrusion Prevention System (IDS / IPS):** These systems monitor network traffic and application activity to detect unusual patterns and potentially malevolent behaviours and create alarms whenever such an activity is detected. The differentiating point between IDSs and IPSs is that the former are passive, in the sense of being limited to alerting upon identification of unusual traffic, while the latter have a more active role, integrating preventive mechanisms that prevent or block the further dispersion of the suspicious data and flows. They can be installed at crucial network points to inspect traffic on the network (network-based systems) or on a single device to monitor port and applications activity (host-based systems). IDS and IPS are mainly signature-based systems, meaning that they rely largely on signatures from known attacks to detect similar patterns, and may be ineffective against new threats for which the related signatures are not available.
- **Unified Threat Management (UTM):** These are single protection units in the form of a device or platform that integrates various mechanisms such as antivirus, IDS/IPS and firewalls. The concentration of threat management in a single point can be efficient in terms of monitoring and training, however it comes with the disadvantage of creating a single point of failure. Additionally, the incorporated security solutions may not fit the needs of the individual network components.
- **Security Information and Event Management (SIEM):** SIEMs are systems collecting information and identified events from various sources such as IDS/IPS, firewalls, antivirus and combining them with network log data, to generate insightful correlations concerning the whole system. They are mainly rule-based, however some modern SIEMs can establish a normal activity baseline from gathered data, and use this for anomaly detection. The deployment and configuration of a SIEM is a very demanding task, requiring the exhaustive analysis of the organisation's infrastructure and engineering specifications, the allocation of adequate hardware resources, as well as the customisation of provided dashboards and visualised analytics to fit the organisation's needs. Additionally, a SIEM creates the requirement for dedicated staff that will monitor and review the emerging alert data, making it a non-viable solution for smaller organisations with limited resources.
- **Log Management System (LMS):** These are software systems that aggregate and store logs from multiple network endpoints and systems into a single point. LMSs

D2.3 GDPR-compliant and Privacy-Preserving Analytics for Healthcare Providers

allow the administrators to view and correlate log data from disparate systems. They have not been designed solely with cybersecurity in mind, as they are also a useful tool for IT management, DevOps & operations and auditing. In comparison to SIEMs, they provide a more lightweight, easy to deploy solution for proactive log monitoring. New age LMSs, such as XpoLog¹⁶ provide, among others, features for AI-enabled error detection – a task closely related to identification of malicious activity through anomaly detection or pattern identification -, real time alerts, smart log analysis and more.

4.3 Current Status and Challenges

The previous section presented methodological and technical approaches that can help healthcare providers towards putting in place a preventive and proactive strategy to secure their infrastructures and the data and information owned or exchanged. Yet, outlining an effective and realistic cybersecurity strategy with the aid of the appropriate software can be a challenge, even with the best practices and guidelines available.

Even when targeted tools are used, administrators struggle with the overwhelming volume and diversity of information that needs to be monitored in order to identify which incidents need to be reviewed and with what priority [76]. Research and development activities to enhance the functionalities of cybersecurity software are constantly in motion to keep up with the evolving threat landscape and the accelerating needs of the current connected world, a reality affecting also healthcare providers.

In a policy-definition level, a cybersecurity expert shall consider the gap between “knowing what to do” and actually “doing what needs to be done”. As highlighted in [74] it is common for outlined security policies to not reflect needs and limitations deriving from day-to-day organisation operations, thus resulting in employees perceiving these policies as a barrier to actual work and finally ignoring them.

Some truly significant barriers every organisation faces in the remediation and mitigation of cybersecurity incidents, are enlisted in [74] and include among others the inherent difficulty of addressing the vast number of emerging and new threats, the lack of personnel with the appropriate background in cybersecurity, the inadequacy of financing for cybersecurity, the management and monitoring of endpoints in infrastructures with numerous interfacing devices, the complexity of network infrastructure and more. An interesting observation from the same survey is the confidence of security experts against these barriers. This can be interpreted in two ways: either as grounded confidence reflecting the strong understanding and knowledge of the respondents, or as over-confidence that could lead to superficial handling and underestimation of threats.

The human factor is adding perplexity towards achieving a satisfactory degree of cybersecurity in the healthcare domain due to lack of awareness. In [80] it is highlighted as a significant hindrance towards fighting two of the most prevailing threats in the domain - phishing and theft of data/equipment. The reason for that lays on multiple factors: the lack of technical knowledge among personnel in the healthcare, the stressful working environment, working in shifts and the communication gap between IT and clinicians can be blamed for the increased vulnerability towards these two threat types. Surprisingly, in [74] a high number of respondents (18%) reported that their healthcare organisations did not conduct phishing tests. This is a highly concerning result, given that the e-mail is the prevailing means of communication and data exchange, and is widely exploited by malicious actors for social engineering attacks, thus leaving the healthcare organisations exposed to attacks.

Usage of privacy-preserving technologies brings new challenges, both in terms of theoretical considerations and technology limitations, as pointed out in a comprehensive study by the e-Sides project [79]. Indicatively, they mention the trade-off between privacy and usefulness of data in anonymisation approaches, as well as the irreversibility of anonymisation over data.

¹⁶ <https://www.xplg.com/>

D2.3 GDPR-compliant and Privacy-Preserving Analytics for Healthcare Providers

Particularly in the anonymisation of medical data, there are additional technical difficulties to be considered, caused by the data format (e.g. free text, indirect descriptions). Techniques such as encryption can protect the actual message of a data flow, but may leave the flow metadata exposed, such as the size of data packets and network protocol [76]. Another factor that should be considered, is the computational cost added by encryption, both for the actual codification of data but also for the inspection of encrypted traffic. Finally, the authors of [79] state that some of the presented privacy-preserving technologies, namely policy enforcement, accountability and transparency and user control technologies have not yet reached the required maturity to be widely adopted and integrated in other systems, and are still subject of research, while noting that the individual techniques are inadequate on their own and shall be combined in holistic solutions to provide the desired privacy.

4.4 Data Analytics for Cyber Threat Detection

It becomes evident that the cybersecurity landscape in healthcare is changing; new technologies are leveraged to help healthcare organisations raise defences against cyber threats, yet new challenges emerge in this process. Apart from the policy-level considerations discussed in the previous sections, technical capabilities and limitations should not be overlooked. The tools presented in Section 4.2 are in principle domain agnostic and need to be properly adapted to the specific needs of a healthcare infrastructure, more so in the aforementioned evolving landscape. The current section presents data analytics algorithms and methods for the detection of threats and abnormal/malicious behaviour that can be valuable in assisting healthcare providers safeguard their assets.

Generally, the detection of cyber threats and attacks is a task that can be approached from two points of view [81][82][83].

The first approach aims at detecting suspicious activities based on previously known attacks [81]. It is called **misuse detection** and is often used interchangeably with the term 'signature-based detection'. The reason is that misuse detection relies heavily on attack profiles (i.e. signatures) that are created for future reference, whenever a new attack is identified in a system. These signatures can be either generated internally or come from external sources and other organisations. Misuse detection can be implemented through knowledge-based or ML-based techniques. In one of the most popular knowledge-based methods, the system uses a set of user- and predefined rules to perform signature matching between the packets under inspection and the signatures in the catalogue, and decides on allowing or dropping the inspected packet. Finite state machines and expert systems can also be employed for knowledge-based detection, using state transitions and pattern matching respectively. ML-based misuse detection is enabled by supervised machine learning techniques, mainly classification and regression, where data from known attacks are used for model training. Some indicative algorithms for misuse detection found in literature are: Back Propagation Artificial Neural Networks (BP-ANN), Decision Trees (DT), Support Vector Machines (SVM). The advantage of misuse-based systems is that when it comes to known attacks, they are very accurate and demonstrate a low rate of false-positives. However, they are incapable of identifying new attacks, while their dependence from signature catalogues creates the demanding requirement for regular knowledge base updates and maintenance. Another limitation is set by the fact that due to fear of exposing system vulnerabilities or other organisation-significant information, organisations that have been under a new attack may not report instantly if at all. Some of the most widespread commercial IDSs and SIEMs are mainly rule-based systems.

The second attack detection approach lays on the assumption that a malicious actor demonstrates different behaviour from a normal user [MH16], and tries to identify deviations from a normal activity baseline. This is called **anomaly-based detection** and relies mainly on behavioural patterns that have been identified as normal. This approach can be implemented with three main techniques: statistical-based, knowledge-based and ml-based [84],[85] and [86]. In statistical-based anomaly detection a reference profile under normal

D2.3 GDPR-compliant and Privacy-Preserving Analytics for Healthcare Providers

conditions is created. This profile is used for the assessment of the deviation of the real captured traffic from normality, and the result of the comparison is reflected in an anomaly score. When this anomaly score surpasses the defined threshold, an alert is created. Univariate, multivariate and time-series are mentioned for the modelling purposes in detection systems. In knowledge-based anomaly detection, previous states of the system are used for behavioural modelling or the extraction of rules that will be employed for the classification of packets. This technique requires a great amount of information and human effort to function properly. The third technique towards anomaly detection takes advantage of the analogy between unsupervised learning that does not require previous knowledge and detecting unknown attacks for which we have no evidence. However, some authors have proposed the mixture of supervised and unsupervised algorithms in order to increase the accuracy of their models. Some indicative machine learning (ML) algorithms that have been mentioned for anomaly detection are: Decision Trees, Random Forests, k-NN, Support Vector Machines, SOM Neural Networks, k-means clustering, Isolation Forests etc. Anomaly-based systems perform with high accuracy in detecting unknown attacks in systems demonstrating static behavioural patterns [84], but do not achieve the same results in other cases.

Table 7: A taxonomy of the most common algorithmic families used for anomaly detection.

	Supervised	Unsupervised	Semi-Supervised
Statistics		ARMA, VARMA, ARIMA, CUSUM, Exp. Smoothing, MCD, EDM, etc.	
Knowledge-based/ Computational models	Rule-based expert systems		FSA, Markov, HMM
Machine Learning	Decision Trees, Random Forest, SVM, ANN, Bayesian Networks	k-means, SOM, EM, LOF, k-NN, Isolation Forests	One-class SVM
Deep Learning	RNN, LSTM, GAN, GRU		RNN, LSTM, GAN, GRU

A common taxonomy of the analytics methods that cope with anomaly or misuse detection in a time series contain three broad categories: supervised, unsupervised and semi-supervised.

4.4.1 Supervised Methods

Supervised learning techniques assume the availability of a dataset where each input pattern has been labelled either as "normal" or "abnormal". This dataset is used to train a classifier, which is then able to determine the right class for any unseen data instance.

Unfortunately, supervised-based techniques, suffer from two major drawbacks when it comes to anomaly detection. First, they rely heavily on experts' domain knowledge regarding the characteristics of an anomaly, which should be provided in the form of labels in the dataset. This is extremely difficult to happen when the data available are vast, and it gets even harder to keep this dataset up-to-date. Second, the anomalous instances are usually far fewer than the normal ones in the training data and this imbalance raises several issues [87].

D2.3 GDPR-compliant and Privacy-Preserving Analytics for Healthcare Providers

In the context of the ASCLEPIOS needs for anomaly detection over data access patterns, employing supervised learning techniques is not considered a feasible option, therefore such methods are not extensively discussed. The most common algorithms of this category are: Support Vector Machines, Decision Trees, Random Forest, Artificial Neural Networks, Rule-based classifiers, and Bayesian networks [88]. Apparently, for this type of techniques the distinction between anomaly and misuse detection makes little sense.

4.4.2 Unsupervised Methods

Unsupervised learning techniques are capable of detecting outliers in an unlabelled dataset under the assumption that the majority of the instances are normal. So, they are looking for instances that seem to fit least to the remainder of the dataset. As such, they are mostly suitable for anomaly detection tasks. Most of these algorithms employ statistics to detect divergence in mean or clustering techniques based on similarity measures.

On the field of statistics, anomaly detection in time-series analysis is based on regression models like those of the family of moving average, such as the autoregressive moving average (ARMA), the autoregressive integrated moving average (ARIMA), the vector autoregression (VARMA), the Seasonal ARIMA (SARIMA) and others. Additional statistical methods include the CUMulative SUM Statistics (CUSUM), the exponential smoothing, the Minimum Covariance Determinant (MCD), the Gibbs Test, the Extreme Studentised Deviate (ECD), etc.[89][90][91]

A more recent approach is the E-Divisive with Medians algorithm (EDM) [92], also known as Twitter's Breakout detection method since it is applied on the Twitter social network to monitor user experience. This approach estimates the statistical significance of a breakout through a permutation test and returns a list of breakout points.

On the other hand, **clustering approaches** group similar data instances based on some similarity measure. Most common approaches rely on distance as a measure, such as the k-means, the k-Medoids or the Expectation-Minimisation (EM) Clustering algorithm. There are also a number of popular density-based or neighbour-based clustering methods, like k-NN, Local Outlier Factor (LOF) and Isolation Forest [93].

4.4.3 Semi-supervised Methods

Semi-supervised techniques are considered to be the most efficient approach when it comes to anomaly or misuse detection in the big data era. They construct a model representing normal behaviour from a given normal training set, and then test the likelihood of a test instance to have been generated by the learnt model. If the probability of having been generated by the given model is very low, then the sequence is marked as anomalous.

This category includes **computational models** like the Finite State Automata (FSA), Markov models and Hidden Markov Models (HMMs) [94][95], as well as Machine Learning binary classifiers like the One-Class SVM.

In more recent studies, **deep learning solutions** such as the Generative Adversarial Networks (GANs), Long ShortTerm Memory (LSTM) and Gated (GRU) recurrent networks have emerged as possible competitors in this category.

The use of autoencoders and GANs is demonstrated in the work of Akcay et al.[96] which is based on an adversarial training architecture in the computer vision domain. A simple LSTM approach is proposed in [97] for anomaly detection in ECG time signals, whereas GRUs can address the problem of missing values in time series analysis according to Che et al [98].

This family of techniques is also the best candidate to deal with multivariate time series. There are some cases, where the goal to detect one entity's anomalies goes through the evaluation of multiple time-dependent variables and the examination of how one affects the other. In practice, this is a very challenging and demanding task since the underlying correlation between the variables can be very complex and difficult to capture. Nevertheless, it is more intuitive, effective and efficient to detect anomalies at the entity-level rather than the metric-level (i.e. per each single variable).

D2.3 GDPR-compliant and Privacy-Preserving Analytics for Healthcare Providers

In this direction, several publications have shown the potential of semi-supervised deep learning methods before to address this challenging task with state-of-the-art performance [99].

To name a few, Malhotra et al. [100][101] proposed a stacked LSTM approach trained on normal data where the resulting prediction errors are modelled as a multivariate Gaussian distribution. This distribution is then employed to assess the likelihood of anomalous behaviour. When multiple input sequences are present, then the joint distribution of the prediction errors is used. Su et al. [102] builds on top of GRU RNNs by utilising key techniques such as stochastic variable connection and planar normalising flow to reconstruct input data by representations, and use the reconstruction probabilities to determine anomalies.

On a side-note, several key players in data analytics, like Twitter, Yahoo or Facebook have developed their own techniques and frameworks for time-series analysis and anomaly detection. For instance, Yahoo has introduced an open-source framework, called EGADS, that consists of three main components: a time-series modelling module, an anomaly detection module and an alerting service [103]. Its architecture promises scalable, accurate and automated anomaly detection. From an algorithmic point of view, it combines several algorithms from statistics and machine learning in various levels in order to reach a satisfactory performance. Facebook's data science team, on the other hand, developed also an open-source solution for time-series forecasting called Prophet¹⁷. Prophet, due to its simplicity, has found many supporters in research community who utilise it for outlier detection purposes with very promising results [104]. Twitter's Breakout detection method has been already described in 4.4.2

4.4.4 Ensemble learning Methods

A recent trend in machine learning is the application of various algorithms, each with its own strengths and weaknesses, onto the same problem, thus constructing a set of models instead of one best model. These models, then, decide with some combinatorial approach (e.g. majority voting) what the best outcome will be. Apparently, the algorithms provide diverse predictions and this is exactly what is exploited by this method, which is called Ensemble learning.

According to [105], single model algorithms may suffer from three different drawbacks:

- statistical problem - appears when the space of hypothesis is too large for the amount of available training data. This results in several algorithms with similar accuracy, followed by the risk of choosing one that will not predict future data points well;
- computational problem - many of the known algorithms are not guaranteed to find the global optimum;
- representation problem - results from the hypothesis space not containing any model that is a good approximation of the true distribution.

These issues may also appear in anomaly or misuse detection in time series and can be addressed by ensemble learning approaches. Examples of this methodology include supervised and semi-supervised methods from both statistics and machine learning families [106][107][108][109].

4.5 Data Visualisation for Cyber Threat Detection

Collection and analysis of data coming from network logs, cybersecurity tools such as IDS/IPS, firewalls and antiviruses, can assist an organisation in gaining knowledge over regular traffic and activities, understanding user behaviours and timely identifying any suspicious actions. The experts are able to monitor data flows, alerts triggered in cybersecurity systems and other types of data, and by following intellectual processes based on their expertise and experience, convey patterns and signals of abuse.

¹⁷ <https://github.com/facebook/prophet>

D2.3 GDPR-compliant and Privacy-Preserving Analytics for Healthcare Providers

However, the vast volume and dynamicality of available data make it extremely demanding and time-consuming for a single person or a group to effectively monitor incoming data and single out any attack indication. It could be stated that employing data analytics methods like the ones described in the previous section can facilitate this process, yet the complexity of the underlying problem prohibits automation through data analytics methods. The output of such systems still needs in most cases to be reviewed and handled by the administrator, which makes it even more important to reduce the number of false positives generated by the employed methods.

[110] focuses on the challenging task of monitoring large-scale networks to detect suspicious behaviours and patterns and note that it is hard for experts to keep up. The authors propose the utilisation of visualisation techniques to assist administrators in the identification of interesting events as well as to limit the attributes -and consequently the amount of data- they need to review only to the actually meaningful. Among others, they study the effectiveness and appropriateness of various visualisation views (e.g. area charts, Gantt charts, Treemaps, network graphs) in the setting of network management and anomaly detection. [76] also highlight the value of visualisations in cyber threat monitoring systems, for increased awareness and modelling purposes in critical infrastructures, zooming into healthcare organisations. They claim that the lack of situational awareness on the side of cybersecurity experts within healthcare organisations is a weak spot in cyber-attack prevention and design a system that allows the operator to manipulate visualisations and configure the parameters in order to get a better view of the organisation's infrastructure, connected devices and more.

Some attack types, like APTs, are based on slowly gaining access to systems and remaining unattended for big periods of time, rather than conducting sudden invasions that would cause disruptions easy to be captured by the human eye. With the help of visual analytics, cybersecurity experts and administrators can use this increasing availability of information to their advantage. They can generate graphical representations that provide an intuitive view of network communication data and finally make sense out of this information storm. Visualisation in cybersecurity finds two main uses: firstly, efficient monitoring - either of the high-level network structure through aggregated data or real-time monitoring through streaming data-, and secondly, visualised anomaly detection as deviation from usual user behaviour.

In [111] the authors conducted an extensive survey on methods and systems of visualisation-enabled anomaly detection in four domains where it has significant value: transactions – fraud detection -, travelling – irregular travelling direction, hotspots, patterns detection-, social interactions – criminals, fraudsters detection- and network communications - cybersecurity. They noted that user behaviours can derive from direct or indirect user actions, as is the case of cyber-attacks conducted by network nodes but orchestrated by a malevolent user. Afterwards they provided an interesting categorisation of anomalous behaviours into egocentric and collaborative, inspired by point and collective anomalies respectively [89]. Typical egocentric anomalous behaviours in cybersecurity include hijacking network traces, a port scan, and unusually high traffic volume on a machine. As collective behaviours, we can see any attack involving more than one information exchange between machines, such as botnets and periodic attacks.

Glyph visualisations are distinguished as the prevailing type for egocentric anomalous behaviours, sequence visualisations as the most appropriate for the collaborative anomalies, while graphs are designated for both types of anomalies [111]. Sequence visualisations can be used to explore network flows and alarms evolution in time and identify periodic patterns and abnormal trends. Regarding graph representations, matrices have been used in egocentric anomalous behaviour detection, where outliers can be indicated with a different node colour [112]. An interactive decision tree in [113] allows the analyst to deduce unlabelled anomalies from areas with sparse training data which include high-density clusters. Collective communications have been visualised as node-link diagrams in [114]. Other examples of collective anomalous behaviours detection graphs include bundle

D2.3 GDPR-compliant and Privacy-Preserving Analytics for Healthcare Providers

diagrams of port activity for scanning attack detection, ring-based representations of alerts for identification of critical attacks [115] [].

Some visualisations offer interaction features facilitating better tracking and monitoring of network communications, while allowing the analyst to explore data at the preferred granularity level, or group events in a handy way to easily come back to them later. In [116] [] the user can store, view together and easily correlate flagged events. Exploration of visual elements in different levels of granularity is also a favourable asset for a visualisation tool. This includes zooming on the graphic area [115] [], clicking on nodes to extend the graph and study the effects of other nodes on the suspicious one, aggregate IPs according to variables such as prefix and country [117] – a useful functionality for big and complicated organisations and networks. Selection of visualised information through available filters [112] for various network traffic parameters (e.g. IP address, ports, protocols, display type) can also assist the analyst. Finally, the ability to export visualisations can also be considered as a helpful interaction.

4.6 Privacy Considerations

Logs are a veritable source of information for cybersecurity systems. They are used for the timely identification of security incidents, policy violations and malevolent activities, but also for auditing and forensic analysis [118]. Certain decisions shall be made regarding the information of data that should be logged, both for practical reasons regarding storage resources and management, but also for safeguarding the privacy of involved persons. Logs are likely to include user IDs, IPs, location and other information that constitute personally identifiable information; thus, they shall be handled accordingly. Unfortunately, the complete removal or anonymisation of these data would render logs useless for monitoring and forensics. The General Data Protection Regulation (GDPR) can provide guidelines for privacy-aware handling of log data and proposes taking certain precautions, in order to ensure transparency and security¹⁸. Some good practices towards this direction are the following:

- The development and application of policies regarding the duration for which these data shall remain at the disposal of the organisation: The cybersecurity department shall have a clear understanding of the information contained in the logs, and in cooperation with other departments assess the period of time this information is truly valuable for cybersecurity purposes. Additionally, appropriate practices shall be designed and activated for the disposal of these data after the defined period.
- Encryption of data and access policies enforcement: Encrypting log data that will be persisted, is one of the most popular ways for ensuring security. The employment of access policies that will grant access only to eligible personnel adds a second layer of security against any abusive activity.

¹⁸ <https://logdna.com/best-practices-for-gdpr-logging/>

5 ASCLEPIOS Cybersecurity, Encryption and Access Analytics for Healthcare Providers

5.1 Motivation and Scope

The previous section provided a landscape review of current cybersecurity challenges in healthcare and presented frameworks, methods and tools that can guide and assist healthcare providers in their efforts to secure their infrastructures and the data and information owned or exchanged. Some of the discussed aspects concern well known cyber threats and attacks which are handled, either in a defensive or in a proactive way, by available cybersecurity tools. Their characterisation as well-known does not imply that they are always easy to detect, but merely that established workflows are foreseen on how to approach them. As highlighted, an effective and holistic cybersecurity strategy further entails detection of abnormal patterns across the voluminous and heterogeneous data generated daily by an organisation's systems in order to timely identify and handle malicious, or otherwise insecure behaviours.

ASCLEPIOS implements advanced data access control mechanisms and leverages powerful cryptographic schemes and trusted hardware capabilities to enhance the security level of healthcare services and enable healthcare providers take advantage of cloud computing. The ASCLEPIOS framework can be used to instantiate innovative and inherently secure workflows for sharing data and insights over data. Details over the inner workings of involved tools and services are provided in the corresponding technical deliverables.

In the context of Task T2.4 it is important to explain that ASCLEPIOS brings into the everyday operations of healthcare providers a set of services and user interactions that out of the box cybersecurity solutions cannot fully address, since there are no established baselines and rulesets to provide insights about whether operation is smooth or anomalies are experienced in terms of user and system behaviour and interactions. ASCLEPIOS services by design enhance the cybersecurity level of the organisation and allow healthcare providers to leverage CSP capabilities. In this context, it is believed that developing targeted analysis processes to complement these new services with monitoring mechanisms will significantly facilitate their adoption by the healthcare providers and will offer valuable insights about the way data access operations take place through them and allow healthcare providers to timely detect and handle abnormal behaviours.

This section presents the ASCLEPIOS Cybersecurity, Encryption and Access Analytics for Healthcare Providers (CEAA) component which is responsible for delivering insights about encryption and decryption activities, data access patterns, normal and abnormal behaviours, cyber threats and security incidents. It should be stressed that the focus of the CEAA component is not on implementing a general-purpose intrusion detection system, as many such commercial solutions are available. Instead, CEAA focuses on providing targeted insights regarding data access patterns that emerge from the usage of CSP operations by healthcare providers who leverage new data encryption, decryption and fine-grained access control mechanisms, as the ones offered by the ASCLEPIOS framework. Insights targeted to such operations and pattern identification shall help healthcare providers that utilise the ASCLEPIOS framework gain a deeper understanding of their system and the way data access services are used and ultimately build threat preventive mechanisms around their infrastructure and data.

5.2 Design Methodology

5.2.1 Definition of Target Usage

The envisioned usage of CEAA within the ASCLEPIOS framework spans across the following three cross-cutting high-level axes:

Axis I: System Monitoring

D2.3 GDPR-compliant and Privacy-Preserving Analytics for Healthcare Providers

CEAA will help healthcare providers understand the way their data handling processes are structured, how they operate and how they are utilised. Towards this goal, CEAA will provide information regarding the data interactions performed daily and the usual data search, access and processing needs, how these differ among stakeholders and among different organisation departments, how they evolve over time etc.

This will primarily address the user's need to have an overall view of the system. Through this monitoring, CEAA will provide to the user metrics that capture the overall system status and show whether the underlying services are used as expected, but will also highlight cases that should be investigated more closely and guide the user to inspect possible deviations from normal operations – not necessarily linked to malicious actions.

Axis II: Behavioural Analysis and Incident Detection and Investigation

CEAA will help healthcare providers identify abnormal usage behaviour, either malicious or otherwise insecure, by providing targeted information, metrics and analysis results. As explained in previous sections, this is not a straightforward task and it becomes apparent even in the attempt to define what constitutes abnormal and/or insecure behaviour.

As a first note, dependence on the underlying organisation's interactions and infrastructure should be considered. It may be the case that cross-department data searches are unusual and therefore indicative of a data breach attempt or could be part of the normal workflow. Successive failed attempts to retrieve a certain data asset could indicate a brute force attack to acquire information but could also be caused by users who forget their credentials. At the same time, a malicious attempt to acquire data may have different impact on the organisation depending on the one hand on the specific data asset and on the other hand on the way it is protected, e.g. acquiring a ciphertext that cannot be further used may not be considered by the administrator equally important as gaining access to unencrypted (plaintext) information.

CEAA will provide the means both to easily locate abnormal behaviours and to further explore their root cause. Abnormal behaviour in this respect could be of two distinct types:

1. The behaviour of a malicious actor attempting an attack against the system, e.g. aiming to compromise the availability of resources or to extract private information
2. The behaviour of a benign user who is having difficulties in using the services in an appropriate way or is, knowingly or not, interacting with a system in an unconventional way which could make it (or the underlying data) vulnerable to attacks

Axis III: Enhancement of the provided services

The ASCLEPIOS services form an innovative framework for data safeguarding and secure exchange of data and knowledge that leverages cloud computing. As such, during the experimentation with the framework, valuable insights will emerge regarding the efficiency of the underlying services and their technical limitations. Having relevant metrics, e.g. time needed to perform an operation on encrypted data, mostly used data workflows, number of registered users and key generation requests etc. shall be very useful towards improving the system design and thus the user experience. These insights are mostly related to scalability issues and performance and will help identify bottlenecks and other issues in the data processing pipelines. Relevant metrics will be also be valuable from the CSP perspective, as they could indicate the need to scale up/down or migrate the deployed services, thus helping in resource planning.

5.2.2 Identification of Input Sources and Data

Having set the scope of CEAA, the next step is to examine the information that is/ will be available to achieve the defined goals in the ASCLEPIOS context. CEAA will provide insights pertaining to the way the ASCLEPIOS data handling services are used and will therefore act as a custom log analytics tool for the information coming from other components of the ASCLEPIOS architecture documented in [21]. This includes the following high-level services:

D2.3 GDPR-compliant and Privacy-Preserving Analytics for Healthcare Providers

- Searching for and retrieving data through the application of the Symmetric Searchable Encryption (SSE) scheme (described in D2.1 [119])
- Searching for and retrieving data through the application of combined SSE and Attribute-Based Encryption (ABE) schemes (described in D2.2 [120]). This can actually be considered as an enhancement over the search functionality which utilises only the SSE scheme.
- Applying functions on encrypted data without revealing the original information through the application of the Functional Encryption (FE) schemes (described in Section 3 of the current deliverable)
- Applying multi party computations based on the privacy preserving analytics mechanisms (the exact mechanisms of which will be reported in D2.4)
- Combining ABAC Attribute Based Access Control (ABAC) and ABE to provide two different layers of authorisation control, i.e. the ABAC layer to permit or deny access and/or editing rights to (encrypted) EHRs; and the ABE layer which handles the way sensitive data should be decrypted (described in D3.1 [121] and refined in the subsequent WP3 deliverables, i.e. D3.2 [122] and D3.3 [123])

CEAA hence needs to monitor all interactions between the different architecture layers and components but also within layers in certain cases, e.g. when search (through searchable encryption) is used in order to identify the inputs that need to be fed to an analytics function (offered through functional encryption). Although the above services are mostly seen from a consumption perspective, they also entail the encryption operations that are required in order for the information to be made available for direct retrieval, search and function application. Hence, CEAA should be able to receive and process all logs generated by the aforementioned services and all foreseen interacting entities. The challenges stemming from this requirement in the current context, i.e. the evolving ASCLEPIOS framework, the privacy and security considerations, and the lack of established logging processes, will be further discussed in Section 5.3.

5.2.3 Definition of Metrics

Having identified the log data sources, i.e. the broad data content that will be made available, and the high-level goals that CEAA should achieve, the next step towards designing the CEAA solution is to outline the insights that should be provided. Towards this goal, a set of metrics that the security analyst, i.e. the CEAA user, would use to assess the system status along the three axes described in Section 5.2.1, were drafted. It should be stressed that even though certain characteristics of the ASCLEPIOS services affecting the information that is available to CEAA are known, it was decided that the metrics formulation process should not be limited to them in order to provide more generalisable insights. The metrics were compiled to help flesh out the way the three core CEAA targets can be accomplished and therefore it was preferred to initially approach the underlying analysis problem in a more holistic and extensible way, even if the implementation will always need to follow the design decisions of the framework as a whole.

Even though as explained the CEAA goals are grouped under three axes, the axes are cross-cutting hence most of the identified metrics are relevant across all of them and the difference lies in the way the information is interpreted or visualised and combined with other data to address a different need.

As a last note before providing these metrics that served as initial design guidelines, it is considered logical that the user would want to define the timeframe of the provided metrics and analytics in a flexible way, e.g. the current month, last month, last week, from [specific datetime] to [specific datetime], etc. Therefore, most of the metrics refer to a specific selected timeframe.

ASCLEPIOS agnostic metrics

This set of metrics is relevant in any cybersecurity system and thus also to CEAA

- | |
|---|
| <ol style="list-style-type: none">1. Number of requests the system received in the corresponding timeframe2. Requestors' OS distribution |
|---|

D2.3 GDPR-compliant and Privacy-Preserving Analytics for Healthcare Providers

3. Distribution of HTTP response codes
4. Distribution of requesting agents (browsers/applications)
5. Requests' distribution per day of week
6. Requests' distribution per hour of the day
7. Number of requests made within the organisation's normal working hours
8. Distribution of requestors' origin country(based on IP)
9. Distribution of used protocols
10. Expected incoming and outgoing traffic
11. Number of encountered (unique) IPs
12. Mostly used ports (ports distribution)
13. Most used system services (traffic distribution across services)
14. Expected sequences of requests (distribution of request sequences)
15. Number of (un)authenticated requests made to the system
16. Average number of unique users interacting with the system daily
17. Number of unique applications interacting with the system services
18. Number and distribution of successful/failed requests

For some of the above metrics, derivative information is also considered, but not listed here exhaustively. As an example, it is possible to extract from metric 11 referring to number of unique IPs, the most commonly encountered IPs or the IPs that have made more than X requests in a given timeframe and this can lead to the definition of new metrics. Furthermore, it should be highlighted that some of the aforementioned metrics are the result of an analysis (e.g. expected traffic), going beyond descriptive statistics. Additionally, depending on the actual usage of the CEAA component, some metrics may be "translated" to more appropriate ones for the target user. As an example, the distribution of ports in the requests is interesting for the security analyst and also from the CSP perspective, but a chart presenting this information from a more business-oriented perspective (e.g. distribution of requests per ASCLEPIOS services that run in known ports and requests targeting non-exposed ports/services) could be more intuitive in some cases.

ASCLEPIOS specific metrics

This set of metrics targets functionalities provided by the ASCLEPIOS framework.

In order to avoid repetition, the 18 metrics defined above are also relevant here but referring either to a specific service (e.g. Number of requests made to the SSE service) or to the comparison among services (e.g. Number of requests each of the available ASCLEPIOS services receives / Distribution of requests between the SSE and FE services). It should be stressed though that even if the metric in these cases is almost the same, the extracted insights may differ significantly due to the inherent services' differences but also due to the underlying workflow, e.g. a failed ABAC request would correspond to a situation different than a failed ABE request (which would happen after a successful ABAC request).

There are also metrics that are service specific and were not included in the previous ones, as follows:

1. Number of "break-glass" requests (i.e. requests made with a special attribute denoting that there is a situation critical for a human life and therefore should be accepted)
2. Number of requests making use of attributes that the organisation has flagged as special (and thus need to be monitored more closely)
3. Distribution of the category of the policies that were violated in failed ABAC requests
4. Distribution of requests to the functional encryption service across the provided functions
5. Distribution of requests to the searchable encryption service across the provided search alternatives (e.g. range queries vs keyword queries vs complex queries)
6. Volume of the data that were encrypted
7. Volume of the data that were encrypted per service
8. Volume of the data that were retrieved
9. Distribution of ASCLEPIOS applications through which requests are made

D2.3 GDPR-compliant and Privacy-Preserving Analytics for Healthcare Providers

10. Role distribution among the users performing requests (e.g. physician, nurse, researcher, ...)
11. Distribution of other selected attributes (regarding the requestor, the context or the requested asset) among the performed requests
12. Distribution of departments from which requests originated
13. Number of requests that are cross-department (i.e. data encrypted by a specific department of the healthcare organisation are requested by another department)
14. Number of requests originated from entities external to the organisation
15. Distribution of search requests along the various data assets
16. Average execution time of a search/retrieval/function application request
17. Percentage of FE requests that came as a follow-up to an SSE request
18. Granularity distribution of the data that were the target of the request (e.g. aggregate data vs a specific EHR)
19. Increase in failed requests per service for a given timeframe
20. Average/maximum number of successive failed requests to a specific service
21. Number of requests performed from the same IP in a given timeframe
22. Maximum number of successive failed requests preceding a successful request
23. Average number of requests to a specific function of the functional encryption service performed daily
24. Average number of requests to a specific function of the functional encryption service from the same requestor performed daily

All defined metrics can be further refined based on contextual information not included in the initial definition, i.e. specific filters can be added to the above metrics leading to a different measurement that could be used to examine different aspects. As an example, examining whether the discussed requests were performed on a weekend would essentially lead to 42 (18 + 24) additional metrics. Obviously, an exhaustive list of metrics cannot be compiled and even if it were possible, it would be inefficient for the user to keep track of so many metrics at the same time.

The reason for generating these metrics was to brainstorm and provide some more concrete insights on the information that needs to be extracted from the services' logs (since many of the above metrics are essentially different computations performed on the same underlying data) and the way it should be processed in order to be valuable for the user.

5.2.4 Requirements and Considerations

Compiling an initial set of metrics that the CEAA should monitor helped identify the type of information that should be monitored and the interactions between the ASCLEPIOS tools that can provide useful insights regarding the security level of the system.

As explained, the process of defining the initial metrics was deliberately not guided by the status of the underlying services. The goal was to collect an initial set of requirements regarding the information that the CEAA user would want in order to have a better understanding of the data access patterns, including encryption and decryption activities but also general information about the network usage, which would allow the smooth integration of ASCLEPIOS functionalities into the healthcare organisation's operations.

Having completed this process, requirements were extracted regarding the data that need to be collected, the data processing methods foreseen and then examine whether any technical, security or privacy issues arise. This analysis highlighted certain aspects that need to be considered before proceeding with the concrete CEAA design and implementation:

1. The underlying services are subject to change: It should be stressed that the ASCLEPIOS services have a strong research orientation and the exact way in which they will be combined and adapted to perform specific functions within real-world use cases in the healthcare domain will be further explored as the project progresses, naturally causing them to evolve. This entails potential implementation changes which could also affect the CEAA functionalities either directly (e.g. a change in the

D2.3 GDPR-compliant and Privacy-Preserving Analytics for Healthcare Providers

- way information is logged) or indirectly (e.g. through extending the integration of two services in order to enable an additional workflow, thus creating the need to add new metrics)
2. The information that is logged and the way it is processed may pose security and privacy risks: As briefly discussed in Section 4.6, logs may reveal information that is either sensitive or can be used to expose system vulnerabilities or help reverse engineer some of its processes. Especially within the ASCLEPIOS context, it is important to ensure that an appropriate balance is found between the two types of security, i.e. to ensure that it is both possible to put in place monitoring and threat detection mechanisms and also to do so in a way that does not impose further vulnerabilities from a privacy or security perspective. Each ASCLEPIOS service-identified metric pair was therefore examined to understand the risk in exposing the corresponding information in contrast with the security risk that could go unnoticed if it were not available.
 3. There are technical limitations regarding the information that can be logged and provided by the underlying tools: Apart from security and privacy concerns stemming from revealing (through logging) information regarding the inner workings of some services, there are also technical limitations that may hinder the extraction of such information altogether. As an example, it may be the case that the way the policy evaluation process is implemented, no granular information can be provided regarding the reasons a request was denied. Since the ASCLEPIOS services are designed primarily to ensure privacy and security in data access and exchange, it may be inherently not possible to provide all information required by the security analyst for monitoring purposes.
 4. Prior knowledge of some information specific to each healthcare organisation is required: Detection of abnormal behaviour entails having a common ground of what constitutes normal behaviour, and this is inherently dependent on contextual and operational information of the underlying organisation. Indicatively, assessing whether receiving requests over patients data during non-working hours of the organisation is normal requires at least the following being known: (a) which are the working hours and (b) whether for the specific organisation such behaviour is expected (e.g. due to research work conducted by interns which is not restricted to specific time intervals) and whether this is something permanent or temporary. Contextualisation of information, which is organisation-specific, is also helpful in terms of presentation and visualisation, as it helps create a more intuitive interface for the security analyst.

The above considerations can serve as high-level requirements about the CEAA architecture: flexibility is required in terms of adding or slightly altering the input data, therefore the data ingestion mechanism should be extensible and also the core data processing methods should be performed on an appropriately normalised dataset and not hard-coded to work with specific predefined data sources (some basic formatting requirements should however be foreseen and agreed among ASCLEPIOS services). Different logging levels should also be foreseen, as especially during the initial experimentation with the ASCLEPIOS services, more concrete insights about the underlying information that could be revealed will be extracted, thus providing guidelines towards this aspect. Furthermore, adapting the provided functionalities to the organisation's context should be supported in a way that does not require significant changes in terms of analytics and visualisation mechanisms, i.e. through a straightforward configuration process. As a final note, although CEAA primarily aims to provide insights stemming from the ASCLEPIOS services, it could also be leveraged to monitor additional data handling services in this context (i.e. flexible data access over encrypted data), therefore providing flexibility in the implementation was also examined from this perspective.

5.3 CEAA Design

5.3.1 Architecture

Based on the defined scope of CEAA, and as also explained in the landscape analysis presented in Section 4, the main functionalities of the tool, which can be directly mapped to design requirements, are:

- To ingest and contextualise log data from numerous and possibly diverse sources
- To perform fast queries and computations over the collected data
- To detect, locate and highlight potential anomalies in the collected data
- To provide meaningful insights to the security analyst in an intuitive way
- To enable the security analyst to adapt data and analysis representation to the current needs

Further leveraging the insights that were extracted during the design methodology steps described in Section 5.2, i.e. (a) the definition of the input sources, (b) the outlined data processing required based on the defined metrics and (c) the extracted requirements regarding configuration flexibility, the CEAA architecture was designed, as depicted in Figure 13.

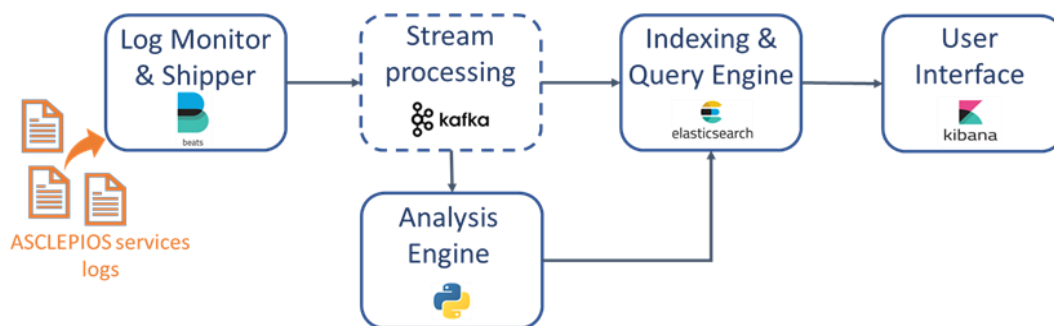


Figure 13: CEAA Architecture

CEAA comprises five main components, as follows:

- A module that constantly monitors the logs from the ASCLEPIOS services and ships them to the main CEAA engine for processing. Depending on the deployment details of the services and the location of the logs, i.e. whether all logs will be centrally placed in the same system path, one or more instances of the log monitor may be required. For each of the logs, a configuration is provided so that the corresponding monitoring module will know how to process the logs before forwarding them to the subsequent pipeline steps. This module uses Filebeat¹⁹ to deliver its functionalities.
- A stream processing module based on Kafka²⁰ that offers scalability and flexibility in the pipelined operations. This module is optional, in the sense that during the initial deployment of the services the expected traffic may not require such mechanisms, in which case the logs are directly sent to the indexing and analysis modules.
- The analysis engine is responsible for the unsupervised anomaly detection methods performed on the collected logs. The engine applies a python implementation²¹ of the open source Twitter breakout detection algorithm²², which was presented in Section 4.4.2, on the requests' timeseries and sends the results to the indexing and query engine.
- The indexing and query engine, powered by Elasticsearch²³, which facilitates the contextualisation and integration of the collected information and enables fast searches and data aggregations to be performed on the data

¹⁹ <https://www.elastic.co/beats/filebeat>

²⁰ <https://kafka.apache.org/>

²¹ <https://github.com/mysl/BreakoutDetection>

²² <https://github.com/twitter/BreakoutDetection>

²³ <https://www.elastic.co/elasticsearch/>

D2.3 GDPR-compliant and Privacy-Preserving Analytics for Healthcare Providers

- The user interface, which is responsible for the presentation of the collected information but also of the generated knowledge over the underlying data in an intuitive way. The user interface is an interactive dashboard implemented using Kibana²⁴.

CEAA leverages established open source tools and technologies to create a scalable and flexible data pipeline, from logs, to analysis, to indexing and querying and finally to interactive visualisations. This pipeline is the backbone of the CEAA workflow which will be presented in the next section.

5.3.2 Functionalities and Workflow

The CEAA workflow, enabled by the previously described architecture, spans across four steps, as shown in Figure 14. The workflow examines the actual usage of CEAA and does not refer to installation and deployment of the tool. For the depicted workflow to run, some configuration aspects on an administrative level (i.e. based on the complete ASCLEPIOS framework) should have been resolved. These are not considered part of the CEAA configuration, but since its functionalities are affected by them, they are described here for completeness:

- The log level of the monitored services should be defined. As previously explained, this depends on the available options of the corresponding tools, which in turn depends on technical limitations and on security and privacy concerns. These are examined both per service and will also be examined for the framework as a whole in the corresponding deliverables
- The specific functions and functionalities offered by the monitored services and used by the specific healthcare organisation should be defined, indicatively including the FE functions and search functionalities that are available.
- The configuration required to parse the log files coming from each of the services should be available. This is actually part of the CEAA implementation, but it is created on the ASCLEPIOS-level and not per user or organisation, as it only depends on the version of the service/tool being used, i.e. each time one of the monitored ASCLEPIOS components changes its functionalities in a way that affects the logging process, the CEAA configuration for this tool should be updated to reflect this change. These configurations mainly refer to the way the log monitoring module (depicted in the architecture in Figure 13) will ship the logs to the other modules for processing.

After the deployment of CEAA and its initial configuration across the aforementioned three aspects, the normal operation of the tool starts, spanning across four conceptual steps, two of which entail explicit user actions and interactions with the tool, whereas the other two involve background tool processes, as depicted in Figure 14. Each step is further detailed below.

²⁴ <https://www.elastic.co/kibana>

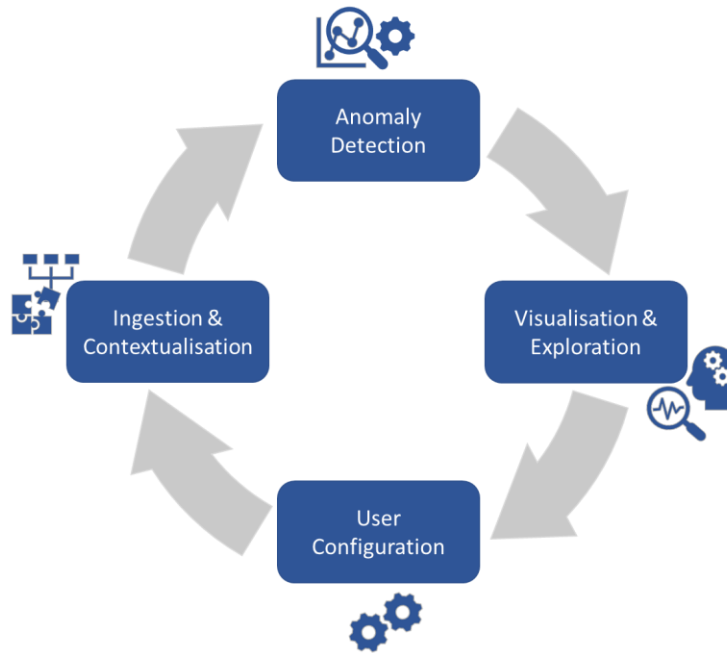


Figure 14: CEAA High-Level Workflow

1. **User configuration:** This step corresponds mainly to the initial configuration required to adapt the CEAA functionalities to the organisation’s needs and also to ensure the processes in the next steps run smoothly and efficiently for the security analyst. As previously explained, certain contextual information depends on the specifics of each healthcare organisation and providing it significantly helps both in presenting the underlying information in a more meaningful way and in improving the anomaly detection functionalities. This initial setup includes:

- The normal working hours of the organisation (conditional definition, e.g. per department or for specific time periods, is also possible)
- The staff roles, e.g. physicians, nurses, etc. (if such attributes are considered)
- The IP ranges of the organisation, which can also be provided per department, if available
- The IP ranges of any collaborating organisations
- Special policies that should be monitored, e.g. it may be the case that a “brake glass” policy is implemented in cases when data access is granted because the corresponding request was flagged as critical for saving a human life.
- Depending on the selected logging level, additional information about the policy categories may also be provided

Some of the aforementioned aspects may change over time and therefore configuration may be needed again to ensure that the way the information is presented remains up to date and well suited to the organisation’s and user’s needs.

2. **Data ingestion and contextualisation:** This step includes all processes from the collection of the input data until their indexing. Logs are first collected from all monitored ASCLEPIOS services which may be available as a single source or not, depending on whether all services store their logs collectively in a central location or if they are kept separately. It should be noted that the collection refers to an almost real time streaming process.

When new log entries are received, they are parsed to extract the required structured information and are transformed into the common underlying schema defined by CEAA. Data ingestion pipelines are used to define contextualisation rules for the

D2.3 GDPR-compliant and Privacy-Preserving Analytics for Healthcare Providers

extracted information, including enriching IPs with GeoIP information, such as country codes and coordinates. The configuration provided in the first step (e.g. regarding working hours) is used to annotate certain fields with additional contextual information. This process is performed in the background leveraging the Kibana “scripted fields” functionality, i.e. the ability to contextualise some fields and use them to enhance the provided visualisations without actually creating and storing additional fields. The same process is used to enrich some date fields with common information, e.g. extract the day of the week. The contextualisation process is very flexible, allowing for new contextual fields to be easily added based on the organisation’s needs even for previously ingested data without requiring re-indexing. After the data transformation is completed, the collected data are indexed and available for queries.

3. **Anomaly Detection:** In this step the Twitter breakout detection algorithm is applied on the requests’ timeseries, as these are extracted from the logs. The algorithm is actually a function based on the EDM (E-Divisive with Medians) algorithm and detects changes in the distribution of a given timeseries based on robust statistical metrics. The provided method is fast and non-parametric and can detect efficiently multiple breakouts in the given timeseries. The algorithm provides as output specific points (instances) of the timeseries, which are stored as annotations, i.e. special fields, in the normalised log information.

Section 4.4 provided insights into numerous anomaly detection methods, both supervised and unsupervised, as well as some semi-supervised approaches. Due to the fact that CEAA processes data from the ASCLEPIOS services there is an inherent difficulty in creating a realistic baseline of what constitutes a normal behaviour. This would require having some prior knowledge about the exact way in which the services will be deployed, integrated and used in the everyday operations of a healthcare organisation. Therefore, supervised and semi-supervised techniques are not considered, although the CEAA architecture and implemented mechanisms could be easily extended to include algorithms in this direction. In terms of unsupervised techniques, the selected algorithm, i.e. Twitter breakout detection, provides a scalable and robust way in detecting deviations from the normal behaviour, which is dynamically defined based on the incoming traffic. It should be noted that due to the fact that CEAA also handles relations among timeseries and sequences of events which depend not only on the individual ASCLEPIOS services but also on their interactions (i.e. on the framework’s behaviour as a whole), anomaly detection cannot be fully automated. The next step in the CEAA workflow explains how visual means are also used in this direction to guide the security analyst in identifying and further inspecting abnormal behaviour.

4. **Visualisation & Exploration:** The final step in this workflow is the one that involves most user interaction with the system, i.e. the visual exploration of the collected data and the intuitive presentation of the extracted insights. CEAA provides an interactive dashboard through which the user can monitor the system behaviour based on various metrics and visualisations, and also filter the presented information in a flexible way to further examine particular instances or groups of instances. The dashboard can be used across all three axes described in section 5.2.1, i.e. (a) for general system monitoring, (b) for behavioural analysis and incident detection and investigation and (c) to deduce potential improvement points for the deployment of the underlying ASCLEPIOS services. The type of information provided to the user as well as the way it is presented and the way the user can interact with it and adapt it is shown in more detail in the next section.

5.4 CEAA Interface and Usage

The present section provides information about the CEAA functionalities in a more graphical way by showing an indicative instantiation of the tool’s interface. As highlighted in Section

D2.3 GDPR-compliant and Privacy-Preserving Analytics for Healthcare Providers

4.5, an interactive interface offering intuitive visualisations can significantly assist security administrators in the identification of interesting events, guide them towards reviewing only meaningful and important in the given context information, thus reducing the time to detect and react to abnormal incidents.

In order to showcase the CEAA interface and since the ASCLEPIOS services it monitors are not yet integrated and deployed, the example is based on appropriately created synthetic data. Synthetic data generation and usage was part also of the CEAA design and experimentation processes, as real data from the deployed tools are not yet available.

In order to create the synthetic datasets, the following process was followed:

1. Open source timeseries datasets from Kaggle²⁵ (provided for anomaly detection algorithm development) were used to create some core data, i.e. the timestamps of requests in realistic patterns.
2. Different log profiles of the ASCLEPIOS tools were created based on discussions among consortium members regarding security and privacy concerns and technical limitations, as previously explained. Different log levels were discussed and are supported by CEAA, which is expected to also provide valuable insights regarding the most appropriate configuration in the normal operation of the tools/ services, i.e. when used by healthcare organisations. In the example shown in this section, the following assumptions are made in the generation of the particular synthetic dataset:
 - a. The staff role of the requestor is available in the logs. The following three roles have been assumed for this sample dataset: physician, nurse, intern. The role “other” appears when none of the previous three is true.
 - b. When requests are performed within the organisation, the source and target departments are logged. For the source department this may be either explicitly logged or deduced by the IP (the latter was the case for this example). For the destination department, since the data assets are uploaded in the cloud, this information is retrieved from the assets’ metadata, if available, i.e. each uploaded asset is linked to the department from which it originated. This could be generalised (e.g. to multiple collaborating care service providers) or completely removed if all assets are considered to be provided by the organisation as a whole. Three departments were assumed in this dataset.
 - c. A “break-glass” policy is implemented, allowing unrestricted access to the requested data under the notion that a human life is in danger. Usage of this policy is logged.
 - d. When requests fail due to policy violation, the high-level category of the policy that caused the failure is logged. This is one of the most challenging aspects in the log levels, as revealing policies may help a malicious user reverse-engineer the system or leak personal information, whereas revealing nothing may obstruct detection and understanding of abnormal behaviour. It should be noted that technical limitations related to enforcing strong security in the data access mechanisms, as previously explained, may not allow such information to be made available.
 - e. The monitored actions include data upload (i.e. encryption and upload, denoted as “insert”), data update and delete, data retrieval, search over data and applying a function on selected data. Three different functions are assumed available (f1, f2, f3) and two distinct search types, one based on a keyword and the other based on a provided range for numeric data.
 - f. There is a distinction in the requests made to obtain aggregate information and the ones made to retrieve individual EHR data. The distinction may be based on concrete logged information per request or on the nature of the service that was invoked (e.g. functional encryption services are by definition

²⁵ <https://www.kaggle.com/>

D2.3 GDPR-compliant and Privacy-Preserving Analytics for Healthcare Providers

targeting aggregate information, whereas search through searchable encryption will retrieve the actual EHR or parts of it).

3. A profile is created for the data distribution across the required variables, i.e. the percentage of requests that will be granted/ denied, the distribution of requests across the available actions and sub-actions, the role and department distribution etc.

CEAA is configurable in the sense that (a) contextual information can be provided to enhance and extend the provided visualisations and (b) the data are modelled in such a way that some actions related to changes in the underlying tools do not require reconfiguration of the CEAA dashboard, e.g. new functions that are added in the functional encryption service will automatically appear in the CEAA dashboard when logged. Although parameterisation and configuration of the interface when it is first used is foreseen, CEAA provides some basic common charts when first deployed. The example shown in this section presents these charts according to the defined log level that was previously described.

Before presenting the CEAA interface, it should be stressed that since the showcase is based on synthetic data, the visualisations depict the CEAA functionalities and not identified patterns in real world healthcare organisations. As an example, the distributions of the dataset variables were defined during the synthetic dataset generation process, therefore no comments are provided on the depicted patterns and frequencies. Furthermore, the values that appear are indicative, i.e. a healthcare organisation may have more than three departments as assumed here and more roles than the three selected here for presentation reasons (the ASCLEPIOS model defines many potential roles that can be used in this context). Finally, some more technical information, e.g. regarding the ports where the services run, was not included in the dataset as it would offer little value for the experimentation purposes that these synthetic data serve.

Figure 15 and Figure 16 depict the overall CEAA dashboard. More details will be provided in the subsequent figures that “zoom into” specific charts and metrics.



D2.3 GDPR-compliant and Privacy-Preserving Analytics for Healthcare Providers



Figure 15: CEAA Dashboard I



D2.3 GDPR-compliant and Privacy-Preserving Analytics for Healthcare Providers

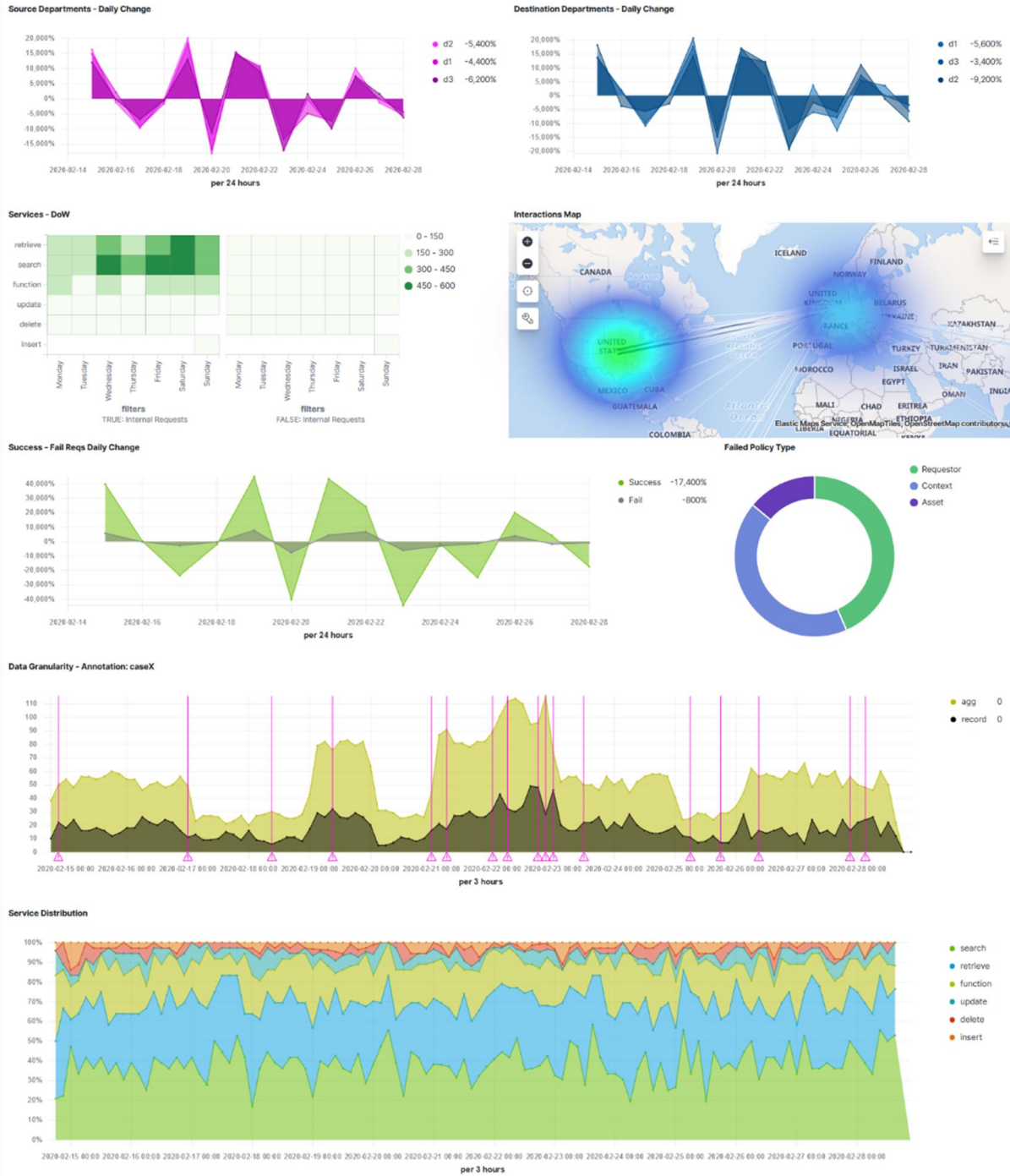


Figure 16: CEAA Dashboard II

The upper part of the dashboard, depicted in Figure 17 has the following information:

- Dropdown filters that can be used to adjust the information that is shown based on some preselected options (e.g. role of the requestor, invoked ASCLEPIOS service, etc). The ASCLEPIOS services are presented as actions to be more intuitive for the end user, e.g. the service that corresponds to SSE is shown as search. For completeness, Figure 18 shows some of the filters with selected values. It should be noted that these filters are meant to provide some quick commonly used filtering options, but all charts can be also used as filters, i.e. by clicking on either on the visualisation or its legend, all information on the dashboard is appropriately updated.

D2.3 GDPR-compliant and Privacy-Preserving Analytics for Healthcare Providers

- Some core metrics that provide quick insights into the current system status (in the time interval selected through the date picker on the upper right), as follows:
 - The number of “Break Glass” requests so that the security administrator can instantly check if usage of this special (due to elevated rights) policy is abnormal
 - The total number of requests and of failed requests and the number of retrieved assets (this could alternatively be changed to show the volume of the retrieved assets)
 - The number of cross department requests and the number of unique requestors based on the source IP of the request
 - Some more detailed statistics regarding failed requests throughout the selected time period and within the last 10 minutes. Specifically, the dashboard monitors the maximum number of successive failed requests to the system’s services, the maximum number of successive failed requests performed by the same IP (requestor) and the maximum number of successive failed requests performed on the same asset.

The aforementioned charts correspond to various metrics defined in Section 5.2.3. The number of “break glass” requests and the number of cross-department requests were explicitly provided in the list of the ASCLEPIOS specific metrics. The number of total and failed requests appears in the first and last ASCLEPIOS agnostic metrics (“Number of requests the system received in the corresponding timeframe” and “Number and distribution of successful/failed requests” respectively), but using the provided filters, the presented value can also express ASCLEPIOS specific metrics (e.g. failed requests per service). The last set of metrics shown in Figure 17 was not included in the originally defined metrics, but as explained the provided list was not (and cannot be) exhaustive. Instead, CEAA is designed to allow flexible metric computations configurable by the user.

Regarding the target usage axes presented in Section 5.2.1, these high level metrics can be used across all three. Indicatively, the number of requests is a general metric that depending on the value could also indicate normal or abnormal behaviour, which in turn may imply malicious activity (threat) or increased benign usage and potential need for more resources. Similarly the number of failed requests on its own may not be very informative, but combined with other shown information can be used to draw valuable conclusions.

As a general note, there is no one to one mapping between the information presented in the CEAA interface and the defined metrics and usage axes, as the interaction capabilities of the dashboard allow more flexibility in this respect.

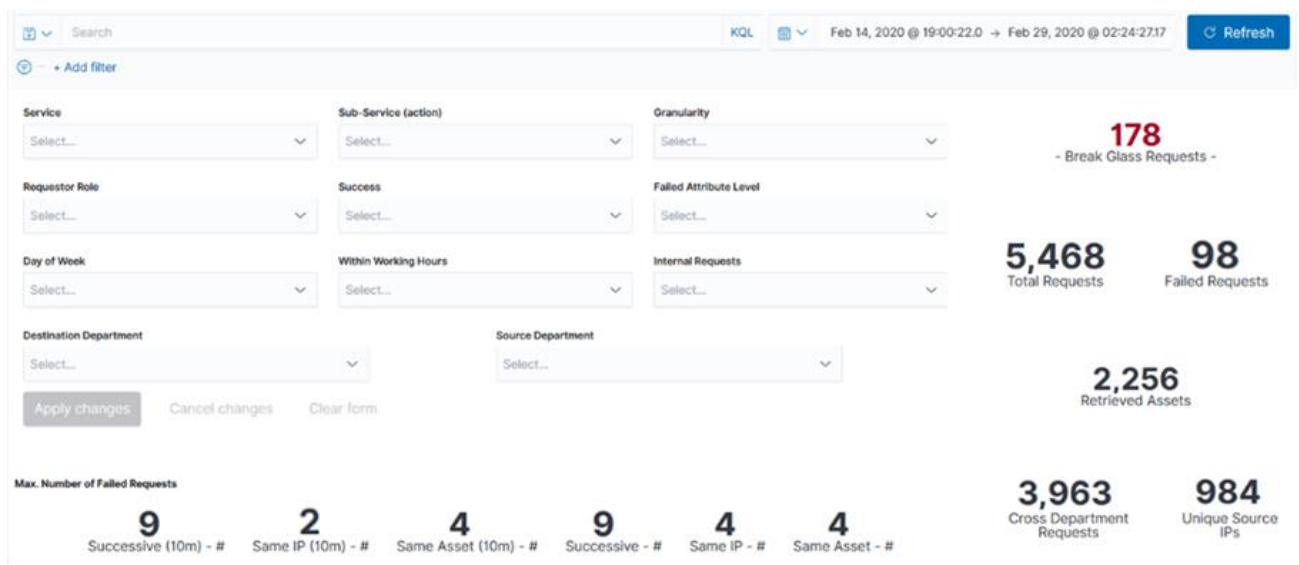


Figure 17: Dashboard Filters & Core Metrics

D2.3 GDPR-compliant and Privacy-Preserving Analytics for Healthcare Providers

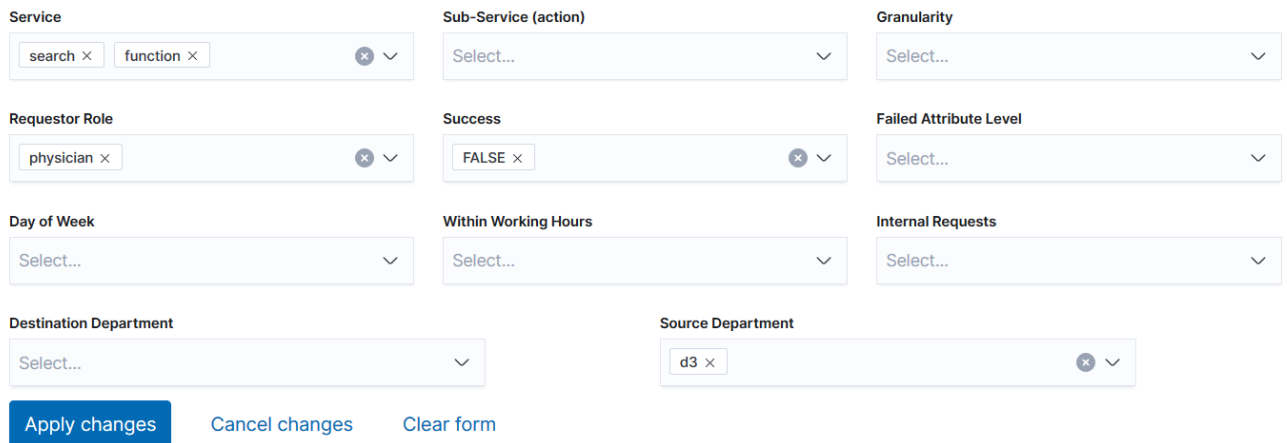


Figure 18: Dashboard Filters Completed

After these high-level statistics, numerous visualisations are provided to the user. The first row of charts shows basic security metrics, i.e. not specific to ASCLEPIOS: requests distribution throughout the day, the most commonly returned HTTP status codes and the most common tags found in the logs, if CEAA is configured to monitor all logs generated by the services (i.e. not only information about the performed requests).

Then some insights regarding the requests' distribution between working and non-working hours is provided, as shown in Figure 19. The distribution is shown in multi-level pie-charts and is examined also per service (on the left) and then also in respect to the requestor's role as staff member (depicted on the right).

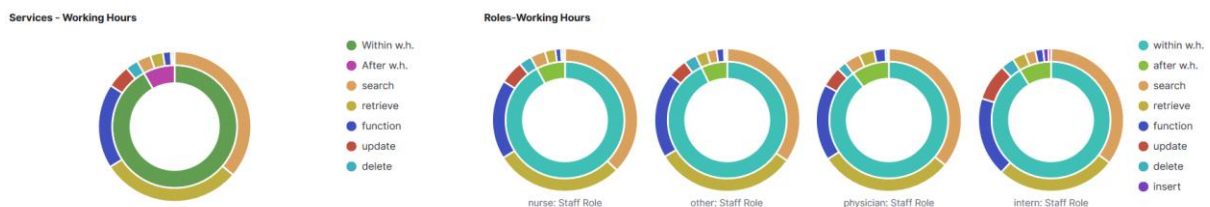


Figure 19: Requests distribution based on invoked services and requestor's role between working and non-working hours of the organisation

Then, as shown in Figure 20, the distribution of the requests over time across the monitored services is provided, where two different incident types are identified and annotated. The first one, shown in red colour, highlights the cases where the "Break Glass" policy was used. The second, shown in light blue, highlights abnormal behaviour in the underlying time series, as identified by the anomaly detection algorithm. Hovering over the annotations provides more information about the detected incident.

D2.3 GDPR-compliant and Privacy-Preserving Analytics for Healthcare Providers

Services over Time (Break Glass, Anomalies)

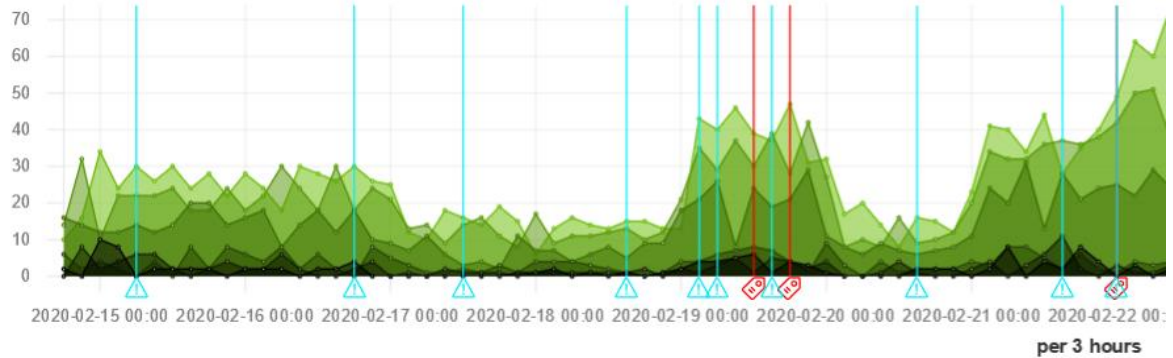


Figure 20: Annotated requests distribution over time

Two charts showing which functions and which search types are used depending on the day of the week are provided. Figure 15 shows both charts, one for the functions applied through functional encryption and one for the two supported search types. Figure 21 zooms into the first one. The reason for showing this information is that certain functions may be more common some days of the week, depending on the way the organisations’ operations are organised and scheduled, therefore deviations may be spotted in this way. If this is not the case and uniform distribution is expected, the chart could be used also in a weekly basis to check whether this is indeed the case or it can also be removed/ altered to be better adjusted to the user’s monitoring needs.

Functions - DoW

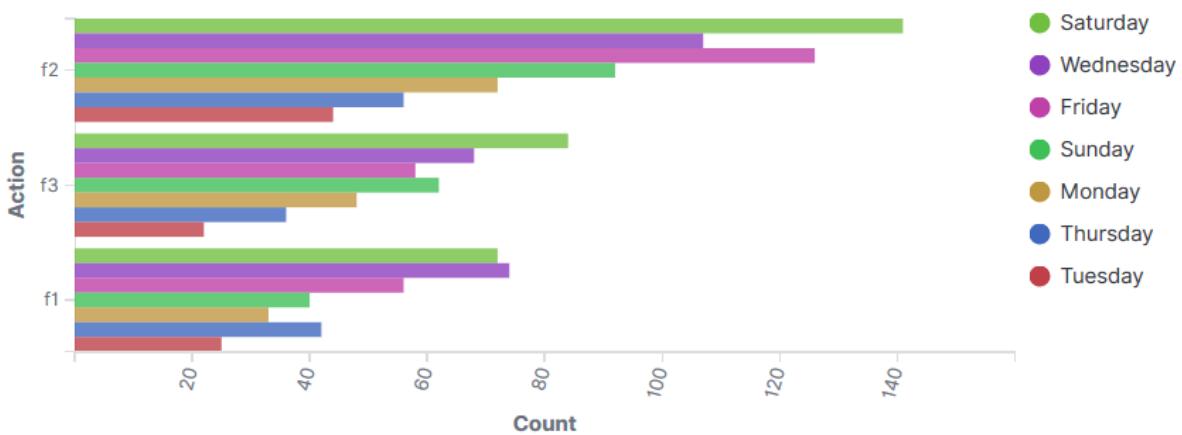


Figure 21: Functions usage across days of the week

The next row of charts shows the way requests are distributed among departments, but monitoring the daily change in the performed requests. This allows the user to have a basic understanding of the normal situation (which may be very different across departments) and easily spot deviations in the behaviour of the requests of a specific department or changes reflected in all departments, indicating a busy day or a potentially abnormal behaviour that should be checked. Figure 22 shows one of the two charts in this row, specifically the one referring to the requests originating from a department (as opposed to the second one that shows requests to access data that are provided by the departments).

D2.3 GDPR-compliant and Privacy-Preserving Analytics for Healthcare Providers

Source Departments - Daily Change

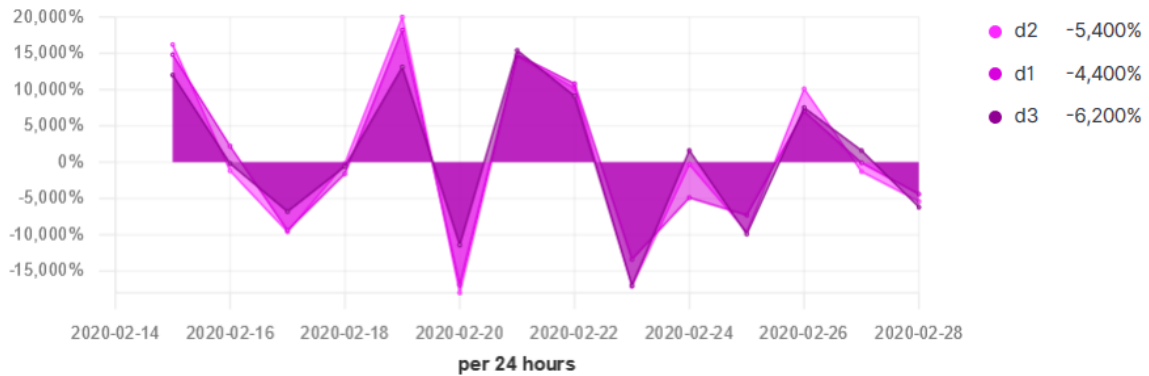


Figure 22: Daily change in number of requests performed per department

Then, a heatmap is provided to highlight the busiest days of the week in relation to the invoked service. A distinction is made based on whether it is an internal to the organisation request or request performed by an external entity, i.e. not from an IP known to belong to the organisation. In the example shown in Figure 23, internal requests are significantly more than the external ones, hence the second heatmap appears empty. Applying a filter to keep only external requests would allow the user to easily inspect these as well. A map is also provided (shown in Figure 16) which can be used both to monitor the distribution of requests across continents and specific countries.

Services - DoW

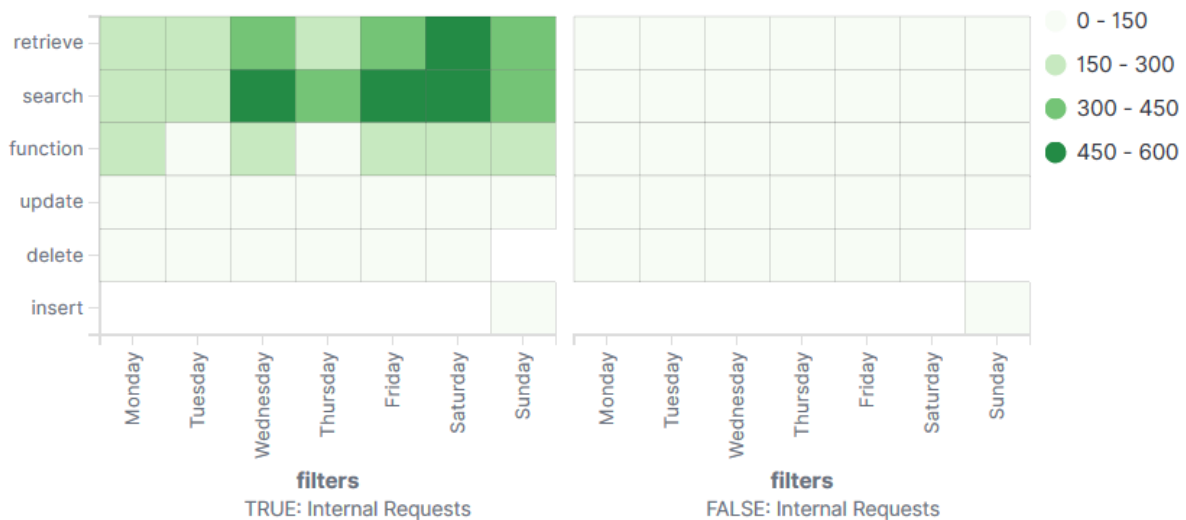


Figure 23: Service heatmap over days of week for internal and external requests

The next two charts, depicted in Figure 24 provide some insights into the distribution of failed requests, first over time in comparison to the total requests and based on the daily change and then (through the depicted pie chart) regarding the policy level that caused the request to fail, i.e. whether it was based on the requestor attributes, the asset attributes or the given context. . It should be noted that these insights depend also on the way this information will be logged, e.g. assuming that both the requestor and the contextual attributes violated the defined policy, it may be the case that both types are logged or only one of them (i.e. the one that was first evaluated and failed). The CEAA user will be aware of the underlying configuration in order to understand what is shown and how it can be leveraged.



D2.3 GDPR-compliant and Privacy-Preserving Analytics for Healthcare Providers

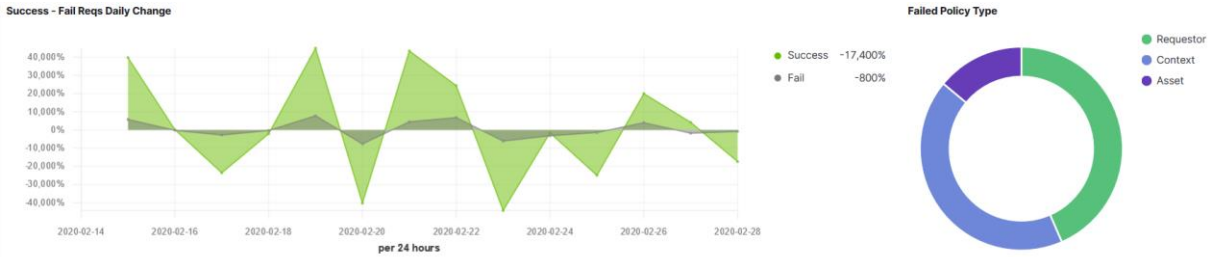


Figure 24: Failed requests distribution and policy insights

The next chart provides insights regarding the granularity level of the retrieved/ used assets, i.e. whether the performed requests refer to data aggregations or specific fine-grained information, i.e. parts of or complete electronic health records. This is shown in Figure 25, along with some annotations on the timeline. These annotations are created based on the initial user configuration if there are any specific cases, denoted by a combination of conditions-filters, that should be identified and reviewed. The pink annotations in the chart mark these cases for the user to easily spot them. It should be noted that more than one annotation types can be configured and also that it is possible to deliver this functionality across the dashboard as a whole, if preferable, i.e. by creating a custom filter to limit the information only to these cases when needed.

Data Granularity - Annotation: caseX

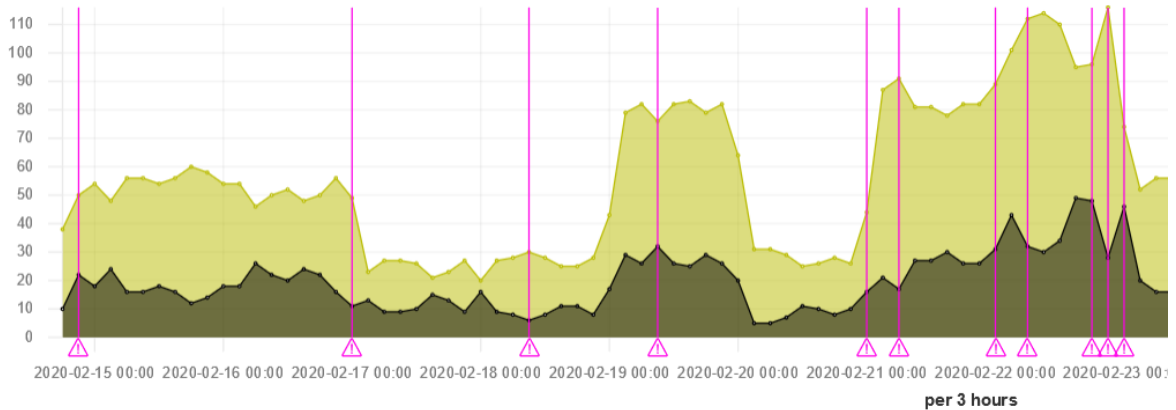


Figure 25: Annotated data granularity distribution of requests over time

The last of the preconfigured default CEAA charts, depicted in Figure 26 shows the requests distribution across the various services. This is similar to the information shown in Figure 20, but here the focus is not only on how these evolve over time but also to highlight their relevant frequency in a more clear way. It is also possible to add annotations for specific cases based on provided rules or for detected anomalies.

D2.3 GDPR-compliant and Privacy-Preserving Analytics for Healthcare Providers

Service Distribution

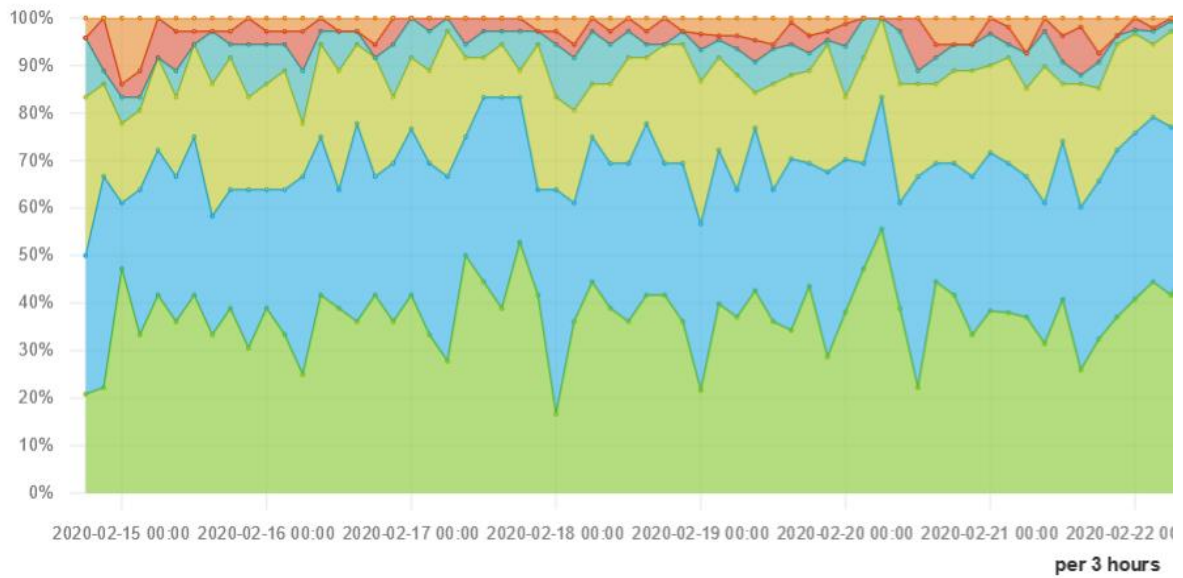


Figure 26: Service distribution (percentages)

As a final note, CEAA can also offer a detailed table of requests along with selected information per request to allow the user to investigate specific incidents and even perform free text search over the logs, if such fine grained information is made available to the user. The default CEAA dashboard provides several configuration and extension options, but only regarding aggregate data.

6 Conclusions

The purpose of the present deliverable was to report on the activities performed in the context of tasks T2.3 “GDPR-compliant and Functional Encryption-enabled Prescriptive Analytics for Healthcare Providers” and T2.4 “Cybersecurity, Encryption and Access Analytics for CSP operation to Healthcare Providers” and present their main outcomes.

In the scope of T2.3, the deliverable first provided an extensive state of the art review of the functional encryption (FE) cryptographic paradigm and presented insights from numerous academic papers in the field spanning across the following functional encryption subclasses: (a) predicate encryption, (b) inner product FE, (c) FE for element-wise operations, (d) FE for quadratic polynomials, (e) FE for general polynomials and (g) FE for randomised functionalities. Inner product FE emerged from the performed analysis as the most promising for helping FE transcend its theoretical boundaries to be used in the implementation of specific mathematical functions and was therefore studied in more detail. Applications of functional encryption were also explored, particularly in the healthcare domain. The deliverable also provided considerations on the applicability of the proposed schemes in respect to the adopted security notions, potential information leakage due to the provided functionality and performance issues. Apart from purely cryptographic approaches, hardware-enabled FE was also discussed. Overall, the potential as well as the current limitations of the functional encryption paradigm were documented.

The deliverable then presented the ASCLEPIOS functional encryption analytics solution that was designed and implemented to enable healthcare providers to perform statistical computations over encrypted data. The provided services leverage inner product FE schemes to offer directly usable mechanisms, available through RESTful services, for healthcare providers to apply commonly used statistics functions on their data. The ASCLEPIOS dual approach that offers flexibility to healthcare providers in selecting the most appropriate solutions for their needs was described. The first of the two FE service categories that were implemented is based on a symmetric multi-input functional encryption scheme, which supports multi-client settings and can be used to perform computations over data from different healthcare providers. This scheme was developed within the ASCLEPIOS context and was documented in detail. The second category of FE services implements two different schemes from the relevant literature based on the asymmetric key setting. Both schemes leverage TEEs and particularly Intel SGX for the implementation of the trusted authority which is responsible for the key generation process. For both approaches, sequence diagrams were provided to show how they can be used and which are the involved entities in the FE analytics workflow. Furthermore, potential combination with the SSE scheme in order to provide flexibility in defining the input to the FE statistical computations and a generally smoother user experience was briefly discussed. Finally, an analysis was performed and presented regarding GDPR compliance considerations that are applicable in the context of the FE-enabled analytics services.

In the scope of T2.4, the deliverable first presented a comprehensive analysis of the cybersecurity landscape in the healthcare domain and identified core cyber threats, such as e-mail phishing, ransomware, insider threats, hijacking of IT-medical equipment, DoS attacks and new age threats. The analysis showed that cyber threats in healthcare do not stem only from malicious actions, but also from human-error and third-party failures. In this context the deliverable presented both methodological frameworks and technical solutions that can help healthcare providers towards putting in place an effective preventive and proactive cybersecurity strategy. The detection of abnormal patterns across the voluminous and heterogeneous data generated daily by a healthcare organisation’s systems was shown to be a challenge that needs to be addressed to allow timely identification and handling of malicious, or otherwise insecure behaviours. In this direction, the deliverable examined numerous data analytics and visualisation techniques that can be used to implement more targeted solutions for the detection of such patterns by the security analysts/ administrators of healthcare organisations.

D2.3 GDPR-compliant and Privacy-Preserving Analytics for Healthcare Providers

Last but not least, the deliverable presented the ASCLEPIOS Cybersecurity, Encryption and Access Analytics (CEAA) component, which is responsible for delivering insights about encryption and decryption activities, data access patterns, normal and abnormal behaviours, cyber threats and security incidents to healthcare providers in the scope of the ASCLEPIOS framework. In this direction, the deliverable first defined the scope of the tool, which was organised across three axes: (a) general cybersecurity level system monitoring, (b) behavioural analysis and incident detection and investigation and (c) extraction of insights regarding scalability and performance aspects of the monitored services. The deliverable presented the metrics that were identified as important to be computed and monitored in this context and provided insights into the extracted requirements and considerations for the CEAA design. The architecture of the tool, which is based on popular and mature open source technologies, was presented and the core data processing workflow was described. Specifically, the deliverable presented how CEAA collects and ingests logs from the ASCLEPIOS services and the way these logs are processed, normalised, contextualised, analysed, indexed and presented to offer meaningful insights for the operations and systems of healthcare providers regarding data access patterns. The significant role of the tool's interactive visualisations in assisting security administrators in the identification and exploration of interesting/ abnormal and potentially threatening events, was finally presented through an indicative showcase of the tool's interface.

References

- [1] Abdalla, M., Catalano, D., Fiore, D., Gay, R., & Ursu, B. (2018, August). Multi-input functional encryption for inner products: function-hiding realizations and constructions without pairings. In Annual International Cryptology Conference (pp. 597-627). Springer, Cham.
- [2] Takashima, K. (2014, December). Recent Topics on Practical Functional Encryption. In 2014 Second International Symposium on Computing and Networking (pp. 21-27). IEEE.
- [3] Boneh, D., Sahai, A., & Waters, B. (2011, March). Functional encryption: Definitions and challenges. In Theory of Cryptography Conference (pp. 253-273). Springer, Berlin, Heidelberg.
- [4] Abdalla, M., Bourse, F., De Caro, A., & Pointcheval, D. (2015, March). Simple functional encryption schemes for inner products. In IACR International Workshop on Public Key Cryptography (pp. 733-751). Springer, Berlin, Heidelberg.
- [5] Agrawal, S., Gorbunov, S., Vaikuntanathan, V., & Wee, H. (2013, August). Functional encryption: New perspectives and lower bounds. In Annual Cryptology Conference (pp. 500-518). Springer, Berlin, Heidelberg.
- [6] De Caro, A., Iovino, V., Jain, A., O'Neill, A., Paneth, O., & Persiano, G. (2013, August). On the achievability of simulation-based security for functional encryption. In Annual Cryptology Conference (pp. 519-535). Springer, Berlin, Heidelberg.
- [7] Wee, H. (2014, September). Functional encryption and its impact on cryptography. In International Conference on Security and Cryptography for Networks (pp. 318-323). Springer, Cham.
- [8] Xu, R., Joshi, J. B., & Li, C. (2019, July). CryptoNN: Training Neural Networks over Encrypted Data. In 2019 IEEE 39th International Conference on Distributed Computing Systems (ICDCS) (pp. 1199-1209). IEEE.
- [9] Sans, E. D., Gay, R., & Pointcheval, D. (2018). Reading in the Dark: Classifying Encrypted Digits with Functional Encryption. IACR Cryptology ePrint Archive, 2018, 206.
- [10] van de Kamp, T., Peter, A., Everts, M. H., & Jonker, W. (2017, November). Multi-client predicate-only encryption for conjunctive equality tests. In International Conference on Cryptology and Network Security (pp. 135-157). Springer, Cham.
- [11] Sharma, D., & Jinwala, D. (2015, December). Functional encryption in IoT e-health care system. In International Conference on Information Systems Security (pp. 345-363). Springer, Cham.
- [12] Garg, S., Gentry, C., Halevi, S., Raykova, M., Sahai, A., & Waters, B. (2016). Candidate indistinguishability obfuscation and functional encryption for all circuits. *SIAM Journal on Computing*, 45(3), 882-929.
- [13] Ananth, P., & Sahai, A. (2016, January). Functional encryption for turing machines. In Theory of Cryptography Conference (pp. 125-153). Springer, Berlin, Heidelberg.
- [14] Goldwasser, S., Gordon, S. D., Goyal, V., Jain, A., Katz, J., Liu, F. H., ... & Zhou, H. S. (2014, May). Multi-input functional encryption. In Annual International Conference on the Theory and Applications of Cryptographic Techniques (pp. 578-602). Springer, Berlin, Heidelberg.
- [15] Agrawal, S., Libert, B., & Stehlé, D. (2016, August). Fully secure functional encryption for inner products, from standard assumptions. In Annual International Cryptology Conference (pp. 333-362). Springer, Berlin, Heidelberg.
- [16] Dupont, P. A., & Pointcheval, D. (2017, April). Functional Encryption with Oblivious Helper. In Proceedings of the 2017 ACM on Asia Conference on Computer and Communications Security (pp. 205-214).
- [17] Katz, J., Sahai, A., & Waters, B. (2008, April). Predicate encryption supporting disjunctions, polynomial equations, and inner products. In annual international

D2.3 GDPR-compliant and Privacy-Preserving Analytics for Healthcare Providers

- conference on the theory and applications of cryptographic techniques (pp. 146-162). Springer, Berlin, Heidelberg.
- [18] Shamir, A. (1984, August). Identity-based cryptosystems and signature schemes. In Workshop on the theory and application of cryptographic techniques (pp. 47-53). Springer, Berlin, Heidelberg.
- [19] Boyen, X., & Waters, B. (2006, August). Anonymous hierarchical identity-based encryption (without random oracles). In Annual International Cryptology Conference (pp. 290-307). Springer, Berlin, Heidelberg.
- [20] Boneh, D., Sahai, A., & Waters, B. (2012). Functional encryption: a new vision for public-key cryptography. *Communications of the ACM*, 55(11), 56-64.
- [21] ASCLEPIOS D1.2 ASCLEPIOS Reference Architecture, Security and E-health Use Cases, and Acceptance Criteria
- [22] Waters, B. (2011, March). Ciphertext-policy attribute-based encryption: An expressive, efficient, and provably secure realization. In International Workshop on Public Key Cryptography (pp. 53-70). Springer, Berlin, Heidelberg.
- [23] Goyal, V., Pandey, O., Sahai, A., & Waters, B. (2006, October). Attribute-based encryption for fine-grained access control of encrypted data. In Proceedings of the 13th ACM conference on Computer and communications security (pp. 89-98).
- [24] Shen, E., Shi, E., & Waters, B. (2009, March). Predicate privacy in encryption systems. In Theory of Cryptography Conference (pp. 457-473). Springer, Berlin, Heidelberg.
- [25] Kim, S., Lewi, K., Mandal, A., Montgomery, H., Roy, A., & Wu, D. J. (2018, September). Function-hiding inner product encryption is practical. In International Conference on Security and Cryptography for Networks (pp. 544-562). Springer, Cham.
- [26] Bishop, A., Jain, A., & Kowalczyk, L. (2015, November). Function-hiding inner product encryption. In International Conference on the Theory and Application of Cryptology and Information Security (pp. 470-491). Springer, Berlin, Heidelberg.
- [27] Datta, P., Dutta, R., & Mukhopadhyay, S. (2016). Functional encryption for inner product with full function privacy. In Public-Key Cryptography–PKC 2016 (pp. 164-195). Springer, Berlin, Heidelberg.
- [28] Tomida, J., Abe, M., & Okamoto, T. (2016, September). Efficient functional encryption for inner-product values with full-hiding security. In International Conference on Information Security (pp. 408-425). Springer, Cham.
- [29] Kim, S., Kim, J., & Seo, J. H. (2019). A new approach to practical function-private inner product encryption. *Theoretical Computer Science*, 783, 22-40.
- [30] Agrawal, S., Agrawal, S., Badrinarayanan, S., Kumarasubramanian, A., Prabhakaran, M., & Sahai, A. (2013). Function Private Functional Encryption and Property Preserving Encryption: New Definitions and Positive Results. *IACR Cryptology ePrint Archive*, 2013, 744.
- [31] Abdalla, M., Gay, R., Raykova, M., & Wee, H. (2017, April). Multi-input inner-product functional encryption from pairings. In Annual International Conference on the Theory and Applications of Cryptographic Techniques (pp. 601-626). Springer, Cham.
- [32] Lin, H. (2017, August). Indistinguishability obfuscation from SXDH on 5-linear maps and locality-5 PRGs. In Annual International Cryptology Conference (pp. 599-629). Springer, Cham.
- [33] Naveed, M., Agrawal, S., Prabhakaran, M., Wang, X., Ayday, E., Hubaux, J. P., & Gunter, C. (2014, November). Controlled functional encryption. In Proceedings of the 2014 ACM SIGSAC Conference on Computer and Communications Security (pp. 1280-1291).
- [34] ElGamal, T. (1985). A public key cryptosystem and a signature scheme based on discrete logarithms. *IEEE transactions on information theory*, 31(4), 469-472.

D2.3 GDPR-compliant and Privacy-Preserving Analytics for Healthcare Providers

- [35] Baltico, C. E. Z., Catalano, D., Fiore, D., & Gay, R. (2017, August). Practical functional encryption for quadratic functions with applications to predicate encryption. In Annual International Cryptology Conference (pp. 67-98). Springer, Cham.
- [36] Marc, T., Stopar, M., Hartman, J., Bizjak, M., & Modic, J. (2019, September). Privacy-Enhanced Machine Learning with Functional Encryption. In European Symposium on Research in Computer Security (pp. 3-21). Springer, Cham.
- [37] Ryffel, T., Sans, E. D., Gay, R., Bach, F., & Pointcheval, D. (2019). Partially encrypted machine learning using functional encryption. arXiv preprint arXiv:1905.10214.
- [38] Chung, K. M., Katz, J., & Zhou, H. S. (2013, December). Functional encryption from (small) hardware tokens. In International Conference on the Theory and Application of Cryptology and Information Security (pp. 120-139). Springer, Berlin, Heidelberg.
- [39] Gorbunov, S., & Vinayagamurthy, D. (2016). Functional Encryption from Secure Enclaves. IACR Cryptology ePrint Archive, 2016, 1071.
- [40] Sahai, A., & Seyalioglu, H. (2010, October). Worry-free encryption: functional encryption with public keys. In Proceedings of the 17th ACM conference on Computer and communications security (pp. 463-472).
- [41] Goldwasser, S., Kalai, Y., Popa, R. A., Vaikuntanathan, V., & Zeldovich, N. (2013, June). Reusable garbled circuits and succinct functional encryption. In Proceedings of the forty-fifth annual ACM symposium on Theory of computing (pp. 555-564).
- [42] Gorbunov, S., Vaikuntanathan, V., & Wee, H. (2012, August). Functional encryption with bounded collusions via multi-party computation. In Annual Cryptology Conference (pp. 162-179). Springer, Berlin, Heidelberg.
- [43] Yao, A. C. C. (1986, October). How to generate and exchange secrets. In 27th Annual Symposium on Foundations of Computer Science (sfcs 1986) (pp. 162-167). IEEE.
- [44] Goyal, V., Jain, A., Koppula, V., & Sahai, A. (2015, March). Functional encryption for randomized functionalities. In Theory of Cryptography Conference (pp. 325-351). Springer, Berlin, Heidelberg.
- [45] Barak, B., Goldreich, O., Impagliazzo, R., Rudich, S., Sahai, A., Vadhan, S., & Yang, K. (2012). On the (im) possibility of obfuscating programs. *Journal of the ACM (JACM)*, 59(2), 1-48.
- [46] Alwen, J., Barbosa, M., Farshim, P., Gennaro, R., Gordon, S. D., Tessaro, S., & Wilson, D. A. (2013, December). On the relationship between functional encryption, obfuscation, and fully homomorphic encryption. In IMA International Conference on Cryptography and Coding (pp. 65-84). Springer, Berlin, Heidelberg.
- [47] Hoekstra, M., Lal, R., Pappachan, P., Phegade, V., & Del Cuvillo, J. (2013). Using innovative instructions to create trustworthy software solutions. *HASP@ ISCA*, 11(10.1145), 2487726-2488370.
- [48] Fisch, B., Vinayagamurthy, D., Boneh, D., & Gorbunov, S. (2017, October). Iron: functional encryption using Intel SGX. In Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security (pp. 765-782).
- [49] Orenbach, M., Lifshits, P., Minkin, M., & Silberstein, M. (2017, April). Eleos: ExitLess OS services for SGX enclaves. In Proceedings of the Twelfth European Conference on Computer Systems (pp. 238-253).
- [50] Schuster, F., Costa, M., Fournet, C., Gkantsidis, C., Peinado, M., Mainar-Ruiz, G., & Russinovich, M. (2015, May). VC3: Trustworthy data analytics in the cloud using SGX. In 2015 IEEE Symposium on Security and Privacy (pp. 38-54). IEEE.
- [51] Tramer, F., & Boneh, D. (2018). Slalom: Fast, verifiable and private execution of neural networks in trusted hardware. arXiv preprint arXiv:1806.03287.
- [52] Lewko, A., Okamoto, T., Sahai, A., Takashima, K., & Waters, B. (2010, May). Fully secure functional encryption: Attribute-based encryption and (hierarchical) inner product encryption. In Annual International Conference on the Theory and Applications of Cryptographic Techniques (pp. 62-91). Springer, Berlin, Heidelberg.

D2.3 GDPR-compliant and Privacy-Preserving Analytics for Healthcare Providers

- [53] D'agostino, R. B., Vasan, R. S., Pencina, M. J., Wolf, P. A., Cobain, M., Massaro, J. M., & Kannel, W. B. (2008). General cardiovascular risk profile for use in primary care. *Circulation*, 117(6), 743-753.
- [54] Xiang, G., Wang, D., Yu, B., & Li, A. (2019). Aggregation Tree Statistical Computing Based on Functional Encryption. *Wuhan University Journal of Natural Sciences*, 24(2), 116-124.
- [55] Bos, J. W., Lauter, K., & Naehrig, M. (2014). Private predictive analysis on encrypted medical data. *Journal of biomedical informatics*, 50, 234-243.
- [56] O'Neill, A. (2010). Definitional Issues in Functional Encryption. *IACR Cryptology ePrint Archive*, 2010, 556.
- [57] Ligier, D., Carpov, S., Fontaine, C., & Sirdey, R. (2017, August). Information leakage analysis of inner-product functional encryption based data classification. In 2017 15th Annual Conference on Privacy, Security and Trust (PST) (pp. 303-3035). IEEE.
- [58] Murdock, K., Oswald, D., Garcia, F. D., Van Bulck, J., Gruss, D., & Piessens, F. (2020). Plundervolt: Software-based Fault Injection Attacks against Intel SGX. In 2020 IEEE Symposium on Security and Privacy (SP).
- [59] Bukasa, S. K., Lashermes, R., Le Bouder, H., Lanet, J. L., & Legay, A. (2017, September). How TrustZone could be bypassed: Side-channel attacks on a modern system-on-chip. In IFIP International Conference on Information Security Theory and Practice (pp. 93-109). Springer, Cham.
- [60] Veale, M., Binns, R., & Edwards, L. (2018). Algorithms that remember: model inversion attacks and data protection law. *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 376(2133), 20180083.
- [61] Elliot, M., O'hara, K., Raab, C., O'Keefe, C. M., Mackey, E., Dibben, C., ... & McCullagh, K. (2018). Functional anonymisation: Personal data and the data environment. *Computer Law & Security Review*, 34(2), 204-221.
- [62] Hanauer, D. A., Preib, R., Zheng, K., & Choi, S. W. (2014). Patient-initiated electronic health record amendment requests. *Journal of the American Medical Informatics Association*, 21(6), 992-1000.
- [63] Michalas, A., Bakas, A., Dang, H. V., & Zaltiko, A. (2019, November). MicroSCOPE: Enabling Access Control in Searchable Encryption with the Use of Attribute-Based Encryption and SGX. In *Nordic Conference on Secure IT Systems* (pp. 254-270). Springer, Cham.
- [64] Bakas, A., & Michalas, A. (2019, October). Modern family: A revocable hybrid encryption scheme based on attribute-based encryption, symmetric searchable encryption and SGX. In *International Conference on Security and Privacy in Communication Systems* (pp. 472-486). Springer, Cham.
- [65] Michalas, A. (2019, April). The lord of the shares: Combining attribute-based encryption and searchable encryption for flexible data sharing. In *Proceedings of the 34th ACM/SIGAPP Symposium on Applied Computing* (pp. 146-155).
- [66] Ananth, P., Brakerski, Z., Segev, G., Vaikuntanathan, V.: From selective to adaptive security in functional encryption. In: *Annual Cryptology Conference*. pp. 657 - 677. Springer (2015)
- [67] European Union Agency for Network and Information Security (2018). *ENISA Threat Landscape Report 2018*. ENISA. <https://doi.org/10.2824/622757>
- [68] O'Brien, G., Edwards, S., Littlefield, K., McNab, N., Wang, S., & Zheng, K. (2018) *Securing Wireless Infusion Pumps in Healthcare Delivery Organizations*. U.S Department of Commerce, National Institute of Standards and Technology (NIST). <https://doi.org/10.6028/NIST.SP.1800-8>
- [69] Cybersecurity Act of 2015, Section 405(d) Task Group (2015). *Health Industry Cybersecurity Practices: Managing Threats and Protecting Patients, Resources and Templates*. <https://www.phe.gov/Preparedness/planning/405d/Documents/resources-templates-508.pdf> (last access: 28/05/2020)

D2.3 GDPR-compliant and Privacy-Preserving Analytics for Healthcare Providers

- [70] Cybersecurity Act of 2015, Section 405(d) Task Group (2015). *Health Industry Cybersecurity Practices: Managing Threats and Protecting Patients*. <https://www.phe.gov/Preparedness/planning/405d/Documents/HICP-Main-508.pdf> (last access: 28/05/2020)
- [71] European Union Agency for Network and Information Security (2016) Smart Hospitals: Security and Resilience for Smart Health Service and Infrastructures. ENISA. <https://www.enisa.europa.eu/publications/cyber-security-and-resilience-for-smart-hospitals> (last access: 28/05/2020)
- [72] Tounsi, W., & Rais, H. (2018). A survey on technical threat intelligence in the age of sophisticated cyber attacks. *Computers & security*, 72, 212-233.
- [73] Alshamrani, A., Myneni, S., Chowdhary, A., & Huang, D. (2019). A survey on advanced persistent threats: Techniques, solutions, challenges, and research opportunities. *IEEE Communications Surveys & Tutorials*, 21(2), 1851-1877.
- [74] Healthcare Information and Management Systems Society (2019). 2019 HIMSS Cybersecurity Survey. HIMSS. https://www.himss.org/sites/hde/files/d7/u132196/2019_HIMSS_Cybersecurity_Survey_Final_Report.pdf (last access: 28/05/2020)
- [75] U.S Department of Commerce, National Institute of Standards and Technology (2018). *Framework for Improving Critical Infrastructure Cybersecurity* (version1.1). NIST. <https://doi.org/10.6028/NIST.CSWP.04162018>
- [76] Boddy, A., Hurst, W., Mackay, M., & El Rhalibi, A. (2016, August). A study into detecting anomalous behaviours within healthcare infrastructures. In 2016 9th International Conference on Developments in eSystems Engineering (DeSE) (pp. 111-117). IEEE.
- [77] Cybersecurity Act of 2015, Section 405(d) Task Group (2015). Technical Volume 1: Cybersecurity Practices for Small Health Care Organizations. <https://www.phe.gov/Preparedness/planning/405d/Documents/tech-vol1-508.pdf> (last access: 28/05/2020)
- [78] Cybersecurity Act of 2015, Section 405(d) Task Group (2015). Technical Volume 1: Cybersecurity Practices for Medium and Large Health Care Organizations. <https://www.phe.gov/Preparedness/planning/405d/Documents/tech-vol2-508.pdf><https://www.phe.gov/Preparedness/planning/405d/Documents/tech-vol1-508.pdf> (last access: 28/05/2020)
- [79] Bachlechner, D., La Fors, K., & Sears, A. M. (2018, December). The Role of Privacy-Preserving Technologies in the Age of Big Data. In *Proceedings of the 13th Pre-ICIS Workshop on Information Security and Privacy* (Vol. 1).
- [80] European Union Agency for Network and Information Security (2020). *Procurement Guidelines for Cybersecurity in Hospitals*. ENISA. <https://doi.org/10.2824/943961>
- [81] Mishra, P., Varadharajan, V., Tupakula, U., & Pilli, E. S. (2018). A detailed investigation and analysis of using machine learning techniques for intrusion detection. *IEEE Communications Surveys & Tutorials*, 21(1), 686-728.
- [82] Faysel, M. A., & Haque, S. S. (2010). Towards cyber defense: research in intrusion detection and intrusion prevention systems. *IJCSNS International Journal of Computer Science and Network Security*, 10(7), 316-325.
- [83] Nadiammai, G. V., & Hemalatha, M. (2014). Effective approach toward Intrusion Detection System using data mining techniques. *Egyptian Informatics Journal*, 15(1), 37-50.
- [84] Butun, I., Morgera, S. D., & Sankar, R. (2013). A survey of intrusion detection systems in wireless sensor networks. *IEEE communications surveys & tutorials*, 16(1), 266-282.
- [85] Garcia-Teodoro, P., Diaz-Verdejo, J., Maciá-Fernández, G., & Vázquez, E. (2009). Anomaly-based network intrusion detection: Techniques, systems and challenges. *computers & security*, 28(1-2), 18-28.

D2.3 GDPR-compliant and Privacy-Preserving Analytics for Healthcare Providers

- [86] Hindy, H., Brosset, D., Bayne, E., Seem, A., Tachtatzis, C., Atkinson, R., & Bellekens, X. (2018). A taxonomy and survey of intrusion detection system design techniques, network threats and datasets. arXiv preprint arXiv:1806.03517.
- [87] Chawla, N. V., Japkowicz, N., & Kotcz, A. (2004). Special issue on learning from imbalanced data sets. ACM SIGKDD explorations newsletter, 6(1), 1-6.
- [88] Ahmed, M., Mahmood, A. N., & Hu, J. (2016). A survey of network anomaly detection techniques. Journal of Network and Computer Applications, 60, 19-31.
- [89] Chandola, V., Banerjee, A., & Kumar, V. (2009). Anomaly detection: A survey. ACM computing surveys (CSUR), 41(3), 1-58.
- [90] Hochenbaum, J., Vallis, O. S., & Kejariwal, A. (2017). Automatic anomaly detection in the cloud via statistical learning. arXiv preprint arXiv:1704.07706.
- [91] Rousseeuw, P. J., & Hubert, M. (2018). Anomaly detection by robust statistics. Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery, 8(2), e1236.
- [92] James, N. A., Kejariwal, A., & Matteson, D. S. (2016, December). Leveraging cloud data to mitigate user experience from 'breaking bad'. In 2016 IEEE International Conference on Big Data (Big Data) (pp. 3499-3508). IEEE.
- [93] Agrawal, S., & Agrawal, J. (2015). Survey on anomaly detection using data mining techniques. Procedia Computer Science, 60, 708-713.
- [94] Gupta, M., Gao, J., Aggarwal, C. C., & Han, J. (2013). Outlier detection for temporal data: A survey. IEEE Transactions on Knowledge and Data Engineering, 26(9), 2250-2267.
- [95] Li, J., Pedrycz, W., & Jamal, I. (2017). Multivariate time series anomaly detection: A framework of Hidden Markov Models. Applied Soft Computing, 60, 229-240.
- [96] Akcay, S., Atapour-Abarghouei, A., & Breckon, T. P. (2018, December). Ganomaly: Semi-supervised anomaly detection via adversarial training. In Asian Conference on Computer Vision (pp. 622-637). Springer, Cham.
- [97] Chauhan, S., & Vig, L. (2015, October). Anomaly detection in ECG time signals via deep long short-term memory networks. In 2015 IEEE International Conference on Data Science and Advanced Analytics (DSAA) (pp. 1-7). IEEE.
- [98] Che, Z., Purushotham, S., Cho, K., Sontag, D., & Liu, Y. (2018). Recurrent neural networks for multivariate time series with missing values. Scientific reports, 8(1), 1-12.
- [99] Filonov, P., Lavrentyev, A., & Vorontsov, A. (2016). Multivariate industrial time series with cyber-attack simulation: Fault detection using an lstm-based predictive data model. arXiv preprint arXiv:1612.06676.
- [100] Malhotra, P., Vig, L., Shroff, G., & Agarwal, P. (2015, April). Long short term memory networks for anomaly detection in time series. In Proceedings (Vol. 89). Presses universitaires de Louvain.
- [101] Malhotra, P., Ramakrishnan, A., Anand, G., Vig, L., Agarwal, P., & Shroff, G. (2016). LSTM-based encoder-decoder for multi-sensor anomaly detection. arXiv preprint arXiv:1607.00148.
- [102] Su, Y., Zhao, Y., Niu, C., Liu, R., Sun, W., & Pei, D. (2019, July). Robust Anomaly Detection for Multivariate Time Series through Stochastic Recurrent Neural Network. In Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining (pp. 2828-2837).
- [103] Laptev, N., Amizadeh, S., & Flint, I. (2015, August). Generic and scalable framework for automated time-series anomaly detection. In Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (pp. 1939-1947).
- [104] Thiagarajan, K., & Ulapane, N. (2020). A Temporal Forecasting Driven Approach Using Facebook's Prophet Method for Anomaly Detection in Sewer Air Temperature Sensor System.
- [105] Dietterich, T. G. (2002). Ensemble learning. The handbook of brain theory and neural networks, 2, 110-125.

D2.3 GDPR-compliant and Privacy-Preserving Analytics for Healthcare Providers

- [106]Vanerio, J., & Casas, P. (2017, August). Ensemble-learning approaches for network security and anomaly detection. In Proceedings of the Workshop on Big Data Analytics and Machine Learning for Data Communication Networks (pp. 1-6).
- [107]Li, Q., Meng, S., Zhang, S., Wu, M., Zhang, J., Ahvanooey, M. T., & Aslam, M. S. (2019). Safety risk monitoring of cyber-physical power systems based on ensemble learning algorithm. *IEEE Access*, 7, 24788-24805.
- [108]Theissler, A. (2017). Detecting known and unknown faults in automotive systems using ensemble-based anomaly detection. *Knowledge-Based Systems*, 123, 163-173.
- [109]Zhong, Y., Chen, W., Wang, Z., Chen, Y., Wang, K., Li, Y., ... & Li, K. (2020). HELAD: A novel network anomaly detection model based on heterogeneous ensemble learning. *Computer Networks*, 169, 107049.
- [110]Zhang, T., Liao, Q., & Shi, L. (2014, March). Bridging the gap of network management and anomaly detection through interactive visualization. In 2014 IEEE Pacific Visualization Symposium (pp. 253-257). IEEE.
- [111]Shi, Y., Liu, Y., Tong, H., He, J., Yan, G., & Cao, N. (2019). Visual Analytics of Anomalous User Behaviors: A Survey. arXiv preprint arXiv:1905.06720.
- [112]Lakkaraju, K., Yurcik, W., & Lee, A. J. (2004, October). NVisionIP: netflow visualizations of system state for security situational awareness. In Proceedings of the 2004 ACM workshop on Visualization and data mining for computer security (pp. 65-72).
- [113]Teoh, S. T., Ma, K. L., Wu, S. F., & Jankun-Kelly, T. J. (2004). Detecting flaws and intruders with visual data analysis. *IEEE Computer Graphics and Applications*, 24(5), 27-35.
- [114]Tao, J., Shi, L., Zhuang, Z., Huang, C., Yu, R., Su, P., ... & Chen, Y. (2018, April). Visual analysis of collective anomalies through high-order correlation graph. In 2018 IEEE Pacific Visualization Symposium (PacificVis) (pp. 150-159). IEEE.
- [115]Livnat, Y., Agutter, J., Moon, S., Erbacher, R. F., & Foresti, S. (2005, June). A visualization paradigm for network intrusion detection. In Proceedings from the sixth annual IEEE SMC information assurance workshop (pp. 92-99). IEEE.
- [116]Fischer, F., & Keim, D. A. (2014). Nstreamaware: Real-time visual analytics for data streams to enhance situational awareness. In Proceedings of the Eleventh Workshop on Visualization for Cyber Security (pp. 65–72). ACM.
- [117]Mansmann, F. (2008). Visual analysis of network traffic: Interactive monitoring, detection, and interpretation of security threats (Doctoral dissertation).
- [118]Kent, K., & Souppaya, M. (2006). *Guide to Computer Security Log Management (Special Publication 800-92)*. U.S Department of Commerce, National Institute of Standards and Technology.
- [119]ASCLEPIOS D2.1 “Symmetric Searchable Encryption and Integration in Medical Devices”
- [120]ASCLEPIOS D2.2 “Attribute-Based Encryption, Dynamic Credentials and Ciphertext Delegation and Integration in Medical Devices”
- [121]ASCLEPIOS D3.1 “ASCLEPIOS Security and Policies Model”
- [122]ASCLEPIOS D3.2 “ASCLEPIOS Models Editor and Interpretation Mechanism”
- [123]ASCLEPIOS D3.3 “Context-aware ABAC Enforcement Mechanism”