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:

D2.4 Data Owners' and Personal Health Data Privacy-Preserving Analytics

Due date of deliverable: 31/082020 Actual submission date: 25/092020

WP2: Operations on Encrypted Health Data and Privacy-Preserving Health Data-Driven Analytics

Dissemination Level: PU

Version: 1.0

ASCLEPIOS is supported by the H2020 Programme under contract no. 826093

Table of Contents

List of Figures and Tables

Figures

Tables

Status, Change History and Glossary

Status

Status:	Name:	Date:	Signature:
Draft:	Kassaye Yitbarek Yigzaw, Alexandra Makhlysheva, Marcela Tuler de Oliveira, Lúcio Henrik A. Reis, Silvia Delgado Olabarriaga, and Christiaan Hillen.	30/08/2020	Kassaye Yitbarek Yigzaw
Reviewed:	Sotiris Koussouris	01/09/2020	Sotiris Koussouris
Approved:	Tamas Kiss	25/09/2020	Tamas Kiss

Table 1: Status change history.

Change History

Version	Date	Author	Modification
0.1	30.03.2020	Christiaan Hillen	Chapter 3, added sections 3.1 and 3.2
0.1a	15.05.2020	Marcela Tuler de Oliveira	Chapter 3, rewrote sections 3.1 and 3.2
0.1b	30.06.2020	Marcela Tuler de Oliveira	Chapter 3, added section 3.3 and 3.4
0.2	10.07.2020	Alexandra Makhlysheva	Chapter 1 and 2
0.3	15.07.2020	Lúcio H. A. Reis	Chapter 3, rewrote section 3.4
0.4	17.07.2020	Kassaye Yigzaw Yitbarek	Rewriting of secure summation protocols
0.4a	24.07.2020	Alexandra Makhlysheva	Text rewriting, references, and formulas

0.5	12.08.2020	Kassaye Yitbarek Yigzaw	Detailed review and rewriting of chapter 1 and 2.
0.6	15.08.2020	Silvia D. Olabarriaga	Chapter 3, detailed review and rewriting of chapter
0.6a	21.08.2020	Alexandra Makhlysheva	Text review
0.7	24.08.2020	Marcela Tuler de Oliveira, Silvia D. Olabarriaga	Finalization of Chapter 3, text review
0.8	24.08.2020	Kassaye Yitbarek Yigzaw	Merge section 3 with the deliverable
0.9	25.08.2020	Kassaye Yitbarek Yigzaw, Alexandra Makhlysheva	Made ready for internal review
1.0	01.09.2020	Alexandra Makhlysheva and Kassaye Yitbarek Yigzaw	Addressing the internal review feedback

Table 2: Deliverable change history.

Glossary

ABAC	Attribute-based access control
ABE	Attribute-based encryption
AMPLE	ASCLEPIOS Models and PoLicies Editors
CEAA	ASCLEPIOS Cybersecurity Encryption Access Analytics
CRUD	Create, read, update, and delete
CSP	Cloud service provider
DBMS	Database management system
DPO	Data protection officer
EMR	Electronic medical records
FE	Functional encryption

FHIR	Fast healthcare interoperability resources
GDPR	General data protection regulation
HE	Homomorphic encryption
PCA	Principal component analysis
PPDDM	Privacy-preserving distributed data mining
PPDSC	Privacy-preserving distributed statistical computation
RA	Registration Authority
SC	Silhouette coefficient
SINE	Secured Intermediate iNformation Exchange
SMC	Secure multi-party computation
SSE	Symmetric Searchable Encryption

SSS	Secret sharing scheme
STTP	Semi-trusted third party
TET	Transparency enhancing tools
UML	Unified modelling language

Table 3: Glossary.

1 Introduction

Healthcare institutions, such as general practitioner offices, hospitals, and laboratories, collect large amount of health data about individuals. Health data reuse has enormous potential for improvement of healthcare quality[1] but raises legitimate privacy concerns from different stakeholder perspectives.

The main goal of the ASCLEPIOS project is to design and develop an e-health framework that will allow patients to store and share their health data in a secure and privacy-preserving way. The WP2 "Operations on Encrypted Health Data and Privacy-Preserving Health Data-Driven Analytics" focuses on the appropriate mechanisms for processing, storing, and sharing data in a secure and privacy-preserving way. In addition, WP2 provides a design of the component allowing authorized stakeholders to perform analytics without breaching users' privacy.

The deliverable is composed of two parts. The first part describes the results of Task 2.5 "Personal Health Data Descriptive Analytics at cross-cloud service provider (CSP) level" where we have developed a tool called Emnet that executes statistics on data distributed across multiple data custodians within a CSP or across multiple CSPs while protecting the privacy of healthcare institutions and patients. The tool is developed based on the security and privacy requirements of the ASCLEPIOS project.

The second part of the deliverable reports the result of Task 2.6 "Data Owners' Privacy Analytics", which contains a set of analytic functions that allow data owners and data controllers to assess the efficacy of the defined access control policies and learn potential improvements for protecting privacy-sensitive data in the medical records stored on the cloud. The metrics provided by these analytic functions are based on logs collected during data processing, including querying, encryption and decryption methods used, processing times, access requests, grants, denials and the profile of a person who processed the data. These provide the data processing transparency and auditability required by the general data protection regulation (GDPR).

[1] Budrionis A, Bellika JG. The learning healthcare system: where are we now? A systematic review. J Biomed Inform. 2016.

2 Personal Health Data Descriptive Analytics at cross-CSP level

2.1 Introduction

Health data reuse, which refers to the use of data for purposes other than those for which the data were initially collected, has enormous potential for improvement of healthcare quality[1]. However, it raises legitimate privacy concerns from different stakeholders. The reuse of health data is guided by a variety of controls from both legal and ethical perspectives, such as privacy rights, data protection regulations, and duties of confidentiality[2]. But legal measures, such as the GDPR, pose challenges for the effective reuse of health data[3,4]. Data reuse techniques must protect the security and privacy of the people and organizations these data represent[5].

The common approach for privacy-preserving processing of data distributed across multiple data sources is a centralized collection of data where each data source de-identifies its data before disclosure. De-identification requires a balance between data utility and privacy where strong privacy protection necessitates significant alterations to the data that considerably reduces the data utility[6].

Privacy-preserving distributed statistical computation (PPDSC), also known as privacy-preserving distributed data mining (PPDDM), is an emerging approach for processing data distributed across multiple data sources and protecting privacy[7,8,9,10]. PPDSC considers the problem of running statistical algorithms on confidential data divided across two or more different data sources without allowing any party to view the private data of another data source. This method reveals

[2] The Nuffield Council on Bioethics (NCOB). The collection, linking and use of data in biomedical research and health care: ethical issues. The Nuffield Council on Bioethics (NCOB); 2015.

[3] Kobayashi S, Kane TB, Paton C. The privacy and security implications of open data in healthcare. Yearb Med Inform. 2018.

[4] Malin B, Goodman K, Section SE for the IYS. Between Access and Privacy: Challenges in Sharing Health Data. Yearb Med Inform. 2018.

[5] Holmes J, Soualmia L, Séroussi B. A 21st century embarrassment of riches: the balance between health data access, usage, and sharing. Yearb Med Inform. 2018.

[6] Dankar FK, El Emam K, Neisa A, Roffey T. Estimating the re-identification risk of clinical data sets. BMC Med Inform Decis Mak. 2012;12:66.

[7] Aldeen YAAS, Salleh M, Razzaque MA. A comprehensive review on privacy preserving data mining. SpringerPlus. 2015.

[8] Aggarwal CC, Yu PS. A general survey of privacy-preserving data mining models and algorithms. Privacy-preserving data mining. New York: Springer; 2008.

[9] Kantarcioglu M. A survey of privacy-preserving methods across horizontally partitioned data. Privacy-preserving data mining. New York: Springer; 2008.

[10] Vaidya J. A survey of privacy-preserving methods across vertically partitioned data. Privacy-preserving data mining. New York: Springer; 2008.

statistics generated from the combined data for a group of data sources, which does not reveal sensitive information about the input.

PPDSC is a special case of a long-studied problem in cryptography called secure multiparty computation (SMC). The field of SMC deals with the question of how to securely compute a functionality on the combined data for a group of data sources without revealing anything apart from the output. However, it does not consider the question of how much information about the input is revealed by that output[11].

Task 2.5 aims to design a solution for running statistical analysis on data distributed across multiple data sources such as healthcare institutions while protecting the privacy of all stakeholders, such as patients, healthcare professionals and healthcare institutions. The developed solution should also scale to a large number of data sources and records to be used in practice.

This section describes the architectural design of a PPDSC tool that supports descriptive statistics on data horizontally partitioned among three or more data sources. We also provide a review of relevant literature that guided our design choice.

2.2 Background

SMC protocols should remain secure even when an adversarial entity controls some subset of the involved parties and wishes to attack the protocol execution. The parties under the control of the adversary are called corrupted and follow the adversary's instructions. SMC protocols are designed with a security model defined with a set of parameters such as (a) cryptographic assumptions, (b) acceptable adversarial behavior, (c) maximum number of corrupted parties, (d) computational bound for an adversary, and (e) when and how the adversary corrupts participating parties[11]. The next subsections describe the necessary background for the PPDSC tool presented in this document.

2.2.1 Adversarial models

There are three common types of adversarial behaviours[11].

The semi-honest adversarial model assumes that even if corrupted parties follow a computation protocol specification, the adversary may try to use the internal state of the corrupted parties, including the messages exchanged during the protocol execution, to learn private information of other uncompromised parties.

The malicious adversarial model considers that corrupted parties arbitrarily deviate from the specifications of a protocol to learn private information of uncompromised parties. Malicious adversarial model uses cryptographic techniques to ensure parties follow the protocol steps.

[11] Lindell Y, Pinkas B. Secure multiparty computation for privacy-preserving data mining. J Priv Confidentiality. 2009.

The covert adversarial model considers that an adversary may also arbitrarily deviate from the protocol specifications, but it does not want to be caught cheating. In this adversarial model, cryptographic techniques are used to ensure that uncompromised parties can detect adversarial behavior with some probability.

Protocols that are considered secure against semi-honest adversaries provide better efficiency and scalability by reducing the privacy guarantee. The semi-honest adversarial model is suitable for the settings in which parties are trusted to follow a computation protocol specification but must run a secure protocol because of legal restrictions for data sharing. This model is also useful to prevent accidental leakage. The security guarantee of semi-honest adversarial model also ensures that an adversary that gained access to a party's database after the execution of a secure protocol cannot learn private information about legitimate parties[11]. Therefore, a semi-honest adversarial model provides a sufficient privacy guarantee in our context while enabling efficient and scalable computation.

2.2.2 Computation models

In SMC protocols, a group of parties jointly executes a set of algorithms on the input data of two or more data sources. The most common computation models are the following[12,13]:

- *Naïve computation model:* data sources jointly execute a set of algorithms on their data.
- *Third-party aided computation model:* one or more third parties aid data sources to achieve more efficient computation. Protocols make assumptions on the trustworthiness of the third parties, such as the third parties have semi-honest adversarial behavior and do not collude with data sources.
- *Outsourced computation model:* data sources share their private data with one or more third parties where the third parties can execute statistics without learning anything about individual records. Data sources share their data using secret sharing schemes (SSS)[14,15] and homomorphic encryption (HE)[16,17].

Naïve and third-party aided computation models maintain access control of data sources where they can decide who can access a specific dataset, when the dataset can be accessed, what analysis can be performed on the dataset, and so on. However, the third-party aided computation model enables more efficient computation. In contrast, data owners do not have control on their

[12] Bohensky MA, Jolley D, Sundararajan V, Evans S, Pilcher DV, Scott I, et al. Data Linkage: A powerful research tool with potential problems. BMC Health Serv Res 2010; 10:346.

[13] Solove DJ. A taxonomy of privacy. Univ Pa Law Rev 2006:477–564

[14] Bogdanov D, Niitsoo M, Toft T, Willemson J. High-performance secure multi-party computation for data mining applications. Int J Inf Secur 2012.

[15] Beimel A. Secret-sharing schemes: a survey. In: Chee YM, Guo Z, Shao F, Tang Y, Wang H, Xing C, editors. Coding Cryptol., Berlin, Germany: Springer; 2011.

[16] Paillier P. Public-Key Cryptosystems Based on Composite Degree Residuosity Classes. Adv. Cryptol. — EUROCRYPT '99, Berlin, Germany: Springer; 1999.

[17] Gentry C. A fully homomorphic encryption scheme. PhD thesis. Stanford University, 2009.

data once data is revealed to third parties with the outsourced computation model. Therefore, we considered third-party aided computation model for the PPDSC tool designed in this deliverable.

2.2.3 Distributed data partitioning

There are three data partitioning models that characterize data distributed across data sources. The models differ based on what attributes are collected and data subjects associated with the data sources.

Horizontal partitioning model characterizes distributed data where data sources collect the same attributes for different sets of individuals. An example of horizontally partitioned data is data distributed across microbiology laboratories. In this deliverable, we assume that data sources collect records of distinct set of patients.

In the *vertical partitioning model*, data sources collect different attributes for the same set of individuals. For example, a hospital and a primary care institution may collect different attributes for the same set of patients.

Hybrid partitioning model is a combination of vertically and horizontally partitioned data.

2.3 Secure summation protocols

Many statistical problems can be decomposed into sub-functions, such as summation, multiplication, scalar product, and comparison, at various stages[18,19]. Therefore, secure protocols for these sub-functions can be used to build a wide variety of statistical problems while protecting privacy. The composition theorem shows that computing a function by composing secure sub-protocols is secure[20]. The efficiency and scalability of the sub-protocols will have a significant effect on the overall performance. The PPDSC tool designed in this deliverable is based on decomposing statistical problems into sub-functions of summation forms and uses secure summation protocols as a building block.

Secure summation protocols compute the summation of private values $\boldsymbol{v_i} \in [\boldsymbol{0}, \boldsymbol{m})$ distributed across $\boldsymbol{N}$ data sources without revealing private values apart from the aggregated result $\boldsymbol{s} = \sum_{\boldsymbol{i=1}}^{\boldsymbol{N}} \boldsymbol{v_i}$. Existing protocols vary on the building blocks they use, such as secret sharing[21,22],

[18] Kantarcioglu M. A survey of privacy-preserving methods across horizontally partitioned data. Privacy-preserving data mining, New York, NY, USA: Springer; 2008.

[19] Clifton C, Kantarcioglu M, Vaidya J, Lin X, Zhu MY. Tools for privacy preserving distributed data mining. ACM SIGKDD Explor News; 2002.

[20] Canetti R. Universally composable security: a new paradigm for cryptographic protocols. Proc. 42nd IEEE Symp. Found. Comput. Sci., IEEE; 2001, p. 136–45.

[21] Bogdanov D, Niitsoo M, Toft T, Willemson J. High-performance secure multi-party computation for data mining applications. Int J Inf Secur 2012.

[22] Beimel A. Secret-sharing schemes: a survey. In: Chee YM, Guo Z, Shao F, Tang Y, Wang H, Xing C, editors. Coding Cryptol., Berlin, Germany: Springer; 2011.

homomorphic encryption[23,24], and randomization of private values. The following subsections review secure summation protocols that are secure against semi-honest adversaries and based on naïve and third-party aided computation models since these computation models enable data owners to maintain access control of their data, which is the objective of the ASCLEPIOS project.

2.3.1 Secret sharing-based secure summation protocol

A simple secure summation protocol is based on the naïve computation model and a linear secret sharing scheme. Let us consider the algorithm $\boldsymbol{Share(v_i)}$ (it is also represented as $[\![\boldsymbol{v_i}]\!]$) that takes a secret value $\boldsymbol{v_i}$ and splits into $\boldsymbol{N}$ shares $(\boldsymbol{v_1^i}, \boldsymbol{v_2^i}, \ldots, \boldsymbol{v_N^i})$, and $\boldsymbol{Rec}(\boldsymbol{v_1^i}, \boldsymbol{v_2^i}, \ldots, \boldsymbol{v_N^i})$ that recreates a secret value from some or all the shares. The protocol has the following steps:

1. Data source D_i runs $Share(v_i) = (v_1^i, v_2^i, \ldots, v_N^i)$ algorithm, where $1 \leq i \leq N$
2. D_i keeps one share v_i^i and sends v_j^i to D_j, where $j \in [1, N]$ and $i \neq j$
3. D_i locally computes the summation of shares $s_i = v_i^1 + v_i^2 + \cdots + v_i^N$

Then, values $\boldsymbol{s_1}, \boldsymbol{s_2}, \ldots, \boldsymbol{s_N}$ that are distributed across the data sources represent a secret share of the sum, $\boldsymbol{s} = \sum_{\boldsymbol{i=1}}^{\boldsymbol{N}} \boldsymbol{v_i}$. The share values are then collected, and the summation result is constructed by running $\boldsymbol{Rec}(\boldsymbol{s_1}, \boldsymbol{s_2}, \ldots, \boldsymbol{s_N})$.

The protocol has a quadratic communication complexity $\boldsymbol{O(N^2)}$. However, if $\boldsymbol{N-1}$ data sources collude, they can find a private value of data source $\boldsymbol{D_i}$.

2.3.2 Homomorphic encryption-based secure summation protocol

The secure summation protocol proposed in Paillier cryptosystem[25] is designed based on the naïve computation model and additive HE[23]. The protocol has the following steps:

1. The input parties form a ring topology, $D_1 \rightarrow D_2 \rightarrow \cdots \rightarrow D_N \rightarrow D_1$
2. D_1 creates an additive HE public and private key pair and sends the public key to all data sources
3. D_1 encrypts its input v_1, $E_{pk^+}(v_1)$, and sends the result to D_2
4. Data source D_i performs the following tasks, where $i \in [2, N-1]$
    a. Receives $E_{pk^+}(\sum_{l=1}^{i-1} v_l)$ from D_{i-1}
    b. Adds $E_{pk^+}(v_i)$ to $E_{pk^+}(\sum_{l=1}^{i-1} v_l)$ and sends the result to D_{i+1}
5. D_N adds $E_{pk^+}(v_N)$ to $E_{pk^+}(\sum_{l=1}^{N} v_l)$ and sends the result to D_1
6. D_1 decrypts $E_{pk^+}(\sum_{l=1}^{N} v_l)$ using the private key to get the actual summation result, $\sum_{l=1}^{N} v_l$

[23] Paillier P. Public-Key Cryptosystems Based on Composite Degree Residuosity Classes. Adv. Cryptol. — EUROCRYPT '99, Berlin, Germany: Springer; 1999.

[24] Gentry C. A fully homomorphic encryption scheme. PhD thesis. Stanford University, 2009.

[25] Kantarcioglu M. A survey of privacy-preserving methods across horizontally partitioned data. Privacy-preserving data mining. New York, NY, USA: Springer; 2008.

The protocol has a linear communication complexity $\boldsymbol{O(N)}$. The protocol remains secure if $\boldsymbol{D_1}$ is honest. If $\boldsymbol{D_1}$ and $\boldsymbol{D_3}$ collude, they can find a private value of data source $\boldsymbol{D_2}$.

2.3.3 Non-cryptographic secure summation protocols

The secure sum protocol proposed in[26,27] is based on a non-cryptographic technique, obscuring secret values with a random number, and the naïve computation model. The protocol has the following steps:

1. The input parties form a ring topology, $D_1 \rightarrow D_2 \rightarrow \cdots \rightarrow D_N \ \rightarrow D_1$
2. D_1 generates a random value $R \in [0, m)$
3. D_1 adds R to its secret value v_1 and sends $s_1 = (R + v_1)$ mod m to D_2
4. Input party D_i performs the following tasks, where $i \in [2, N-1]$
    a. Receives $s_{i-1} = (R + \sum_{l=1}^{i-1} s_l)$ mod m from D_{i-1}
    b. Computes the following and sends the result to D_{i+1}
    $s_i = (s_{i-1} + v_i)$ mod $m = (R + \sum_{l=1}^{i-1} s_l)$ mod m
5. D_N computes $s_N = (s_{N-1} + v_N)$ mod m and sends the result to D_1
6. D_1 computes the sum as follows and broadcasts the result to all input parties
    $s = (s_N - R) \bmod m = \left(R + \sum_{l=1}^{N} s_l\right) - R)$ mod m

The non-cryptographic based protocol has linear communication complexity $\boldsymbol{O(N)}$. This secure summation protocol does not protect the secret value of $\boldsymbol{D_i}$, if $\boldsymbol{D_{i-1}}$ and $\boldsymbol{D_{i+1}}$ collude. Karr et al.[28] proposed an extension to the protocol that makes collusion difficult through constructing a ring topology at runtime and hiding it from the input parties. However, the protocol doubles the communication overhead.

There is another extension[29] to the non-cryptographic based protocol. It is based on the idea of onion routing where $\boldsymbol{D_1}$ creates messages encapsulated in layers of encryption, and each input party $\boldsymbol{D_i}$ peels away a single layer and identifies the previous input party $\boldsymbol{D_{i-1}}$ and the next input party $\boldsymbol{D_{i+1}}$. A digital signature is used to check whether the message originated from $\boldsymbol{D_1}$. Identifying the previous input party $\boldsymbol{D_{i-1}}$ enables $\boldsymbol{D_i}$ to verify that the message has come from the input party intended by $\boldsymbol{D_1}$. Therefore, the data sources, except $\boldsymbol{D_1}$, learn partial knowledge about the ring topology which makes collusion difficult.

[26] Clifton C, Kantarcioglu M, Vaidya J, Lin X, Zhu MY. Tools for privacy preserving distributed data mining. ACM SIGKDD Explor News 2002.

[27] Karr AF, Lin X, Sanil AP, Reiter JP. Secure regression on distributed databases. J Comput Graph Stat 2005

[28] Karr AF, Fulp WJ, Vera F, Young SS, Lin X, Reiter JP. Secure, Privacy-Preserving Analysis of Distributed Databases. Technometrics 2007.

[29] Andersen A, Yigzaw KY, Karlsen R. Privacy preserving health data processing. IEEE 16th Int. Conf. E-Health Netw. Appl. Serv. Heal., IEEE; 2014.

The protocol has a linear communication complexity $\boldsymbol{O(N)}$. However, compared to the original protocol, it introduces additional computation overhead of data encryption and decryption. The protocol remains secure if $\boldsymbol{D_1}$ is honest. If $\boldsymbol{D_1}$ and $\boldsymbol{D_3}$ collude, they can find a private value of data source $\boldsymbol{D_2}$; if $\boldsymbol{D_1}$ colludes with $\boldsymbol{D_i}$ and $\boldsymbol{D_{i+2}}$, they can learn private value of $\boldsymbol{D_{i+1}}$.

Secured Intermediate iNformation Exchange (SINE)[30] is also an extension to the non-cryptographic based protocol. The protocol uses the aid of a semi-trusted third party, denoted as coordinator. A coordinator sends a random number $\boldsymbol{R_c}$ to the first node. The first node adds its own random number to this value and passes the result to the second node. The second node adds its random number to the input and sends the result to the third node. Finally, the coordinator subtracts its own random number from the sum value received from the last data source (which is $\boldsymbol{R_c + R_1 + R_2 + \cdots + R_N}$) to find the summation of the random numbers of all data sources. In addition, each data source sends the summation of its private value and its random number $\boldsymbol{R_i + S_i}$ to the coordinator. To find the summation of private values of all data sources, the coordinator calculates the sums of random numbers and private values received from each data sources and subtracts the sum of random numbers of all data sources.

The protocol has a linear communication complexity $\boldsymbol{O(N)}$. The protocol remains secure for computation between three of more data sources if the coordinator is uncompromised. If the coordinator colludes with $\boldsymbol{D_2}$, they can find a private value of data sources $\boldsymbol{D_1}$; if the coordinator colludes with $\boldsymbol{D_i}$ and $\boldsymbol{D_{i+2}}$, they can learn private value of $\boldsymbol{D_{i+1}}$. However, it is easier to keep the coordinator secure from an outside adversary.

The selection of a secure summation protocol involves a trade-off between security guarantee and computation efficiency. The reviewed protocols have a linear communication complexity $\boldsymbol{O(N)}$, except the secret-sharing based protocol that has a quadratic communication complexity $\boldsymbol{O(N^2)}$. The additive HE (Paillier cryptosystem) adds more computation overhead than the non-cryptographic based protocol. The SINE and onion routing-based protocols provide better privacy guarantee than the original non-cryptographic based protocol. However, the symmetric cryptosystem used for the onion routing-based protocol introduces additional computation overhead.

We have chosen the SINE protocol for the PPDSC tool because it makes a good tread-off between privacy guarantee and computation overhead. Note that the PPDSC tool is based on decomposing statistical problems into sub-functions where a sub-function is computed using a secure summation protocol. The PPDSC involves a coordinator that orchestrates the execution of a statistical problem based on the execution of its sub-functions. Therefore, a secure summation protocol based on third-party aided computation model, in which the third party receives the result, was required.

[30] Wang S, Jiang X, Wu Y, Cui L, Cheng S, Ohno-Machado L. EXpectation Propagation LOgistic REgRession (EXPLORER): Distributed Privacy-Preserving Online Model Learning. J Biomed Inform 2013.

2.3.4 *k*-secure summation protocol

The $\boldsymbol{k}$-secure summation protocol is proposed for scaling existing secure summation protocols based on dividing data sources into groups, privacy peers, and the privacy peers in parallel execute multiple instances of a secure summation protocol[31]. Figure 1 shows an overview of the protocol's execution[31]. The protocol has the following steps:

1. The coordinator partitions the data custodians into groups of k data custodians. Each group of data custodians is denoted as a privacy peer (PP_j).
2. Data sources in each privacy peer PP_j jointly run a secure summation protocol to compute the summation of their private values v_i, $\{\sum v_i\}_{PP_j}$.
3. Each privacy peer PP_j sends the summation result $\{\sum v_i\}_{PP_j}$ to the coordinator.
4. The coordinator locally sums the values $\{\sum v_i\}_{PP_j}$ received from all privacy peers, $s = \sum_{i=1}^{N} v_i$.

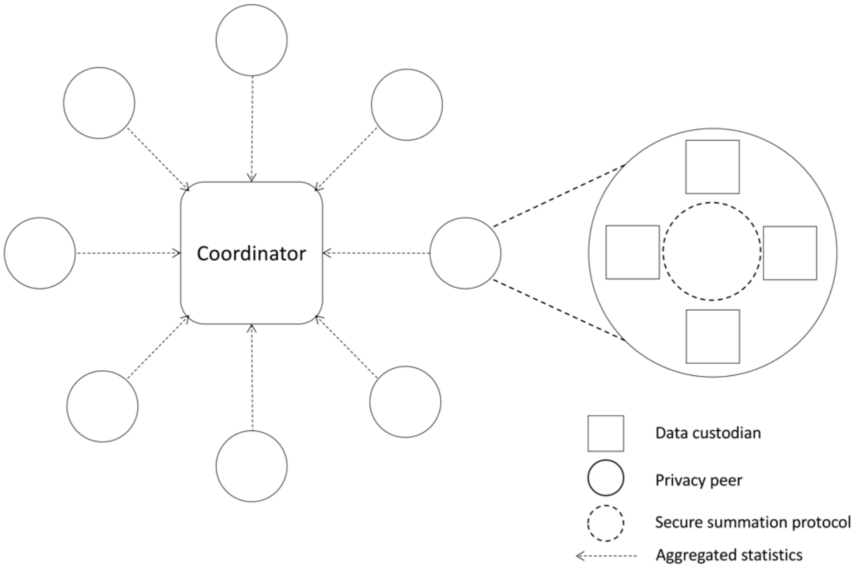


Figure 1: An overview of the *k*-secure summation protocol.

The protocol allows a data source to decide the minimum value of k where it considers that revealing $\{\sum v_i\}_{PP_j}$, which is the result on the combined datasets of $\geq k$ data custodians, has an acceptable privacy risk. Therefore, if the underlying summation protocol is secure, the k-secure summation protocol is secure.

[31] Yigzaw KY. Towards Practical Privacy-Preserving Distributed Statistical Computation of Health Data. UiT The Arctic University of Norway. PhD thesis. 2016.

2.4 Related works

There are several frameworks for PPDSC. The frameworks differ by the following characteristics:

(1) the statistical problems that are supported,
(2) the number of data sources,
(3) how data is partitioned between the data sources,
(4) computation and security models,
(5) building blocks, and
(6) efficiency and scalability.

Existing frameworks for PPDSC include Sharemind[32], SEPIA[33], and VIFF[34]. These frameworks are based on outsourced computation model. Sharemind and SEPIA are based on outsourcing computations to three third parties, whereas the VIFF is based on outsourcing to three or more third parties. The data sources use secret sharing to distribute their private data to a set of third parties, and the third parties execute statistical queries based on composing secure sub-protocols that compute on secret shared data. The frameworks use basic secure sub-protocols such as addition, multiplication, and comparisons.

2.5 Security requirements and assumptions

We assume horizontally partitioned data distributed across multiple healthcare institutions. We assume that information about patients, clinicians, and healthcare institutions cannot be disclosed outside the organization that originally recorded the data. Aggregated information about a clinician, such as performance indicators, is also private information. However, statistics generated from the combined data on a group of healthcare institutions are not considered sensitive information and, thus, can be disclosed. Aggregated statistics includes statistics generated for a group of clinicians from multiple healthcare institutions. Only aggregated statistics generated from at least k number of health institutions can be revealed, where k is defined by the privacy requirements of health institutions.

We assume that outside adversaries may compromise a subset of healthcare institutions by breaking into the network or compromising employees. Parties under the control of an adversary are called corrupted parties. We relied on a standard security assumption called the semi-honest (honest-but-curious) adversarial model.

[32] Bogdanov D, Laur S, Willemson J. Sharemind - A Framework for Fast Privacy-Preserving Computations. In: Jajodia S, Lopez J, editors. Proc. 13th Eur. Symp. Res. Comput. Secur. Comput. Secur., Berlin, Germany: Springer-Verlag; 2008, p. 192–206.
[33] Burkhart M, Strasser M, Many D, Dimitropoulos X. SEPIA: Privacy-preserving Aggregation of Multi-domain Network Events and Statistics. Proc. 19th USENIX Conf. Secur., Berkeley, CA, USA: USENIX Association; 2010, p. 15–15.
[34] Damgård I, Geisler M, Krøigaard M, Nielsen J. Asynchronous multiparty computation: Theory and implementation. Public Key Cryptogr 2009:160–179.

We considered a third-party aided computation model where the third party, denoted as the *coordinator*, satisfies the honest-but-curious adversarial model. The coordinator aids computations without learning private information.

2.6 Architectural design

Figure 2 shows the architectural design of Emnet, the privacy-preserving distributed statistical computation tool for computing on distributed health data. The tool contains software components running at the coordinator and healthcare institutions. The following subsections describe the software components and how the different components work together.

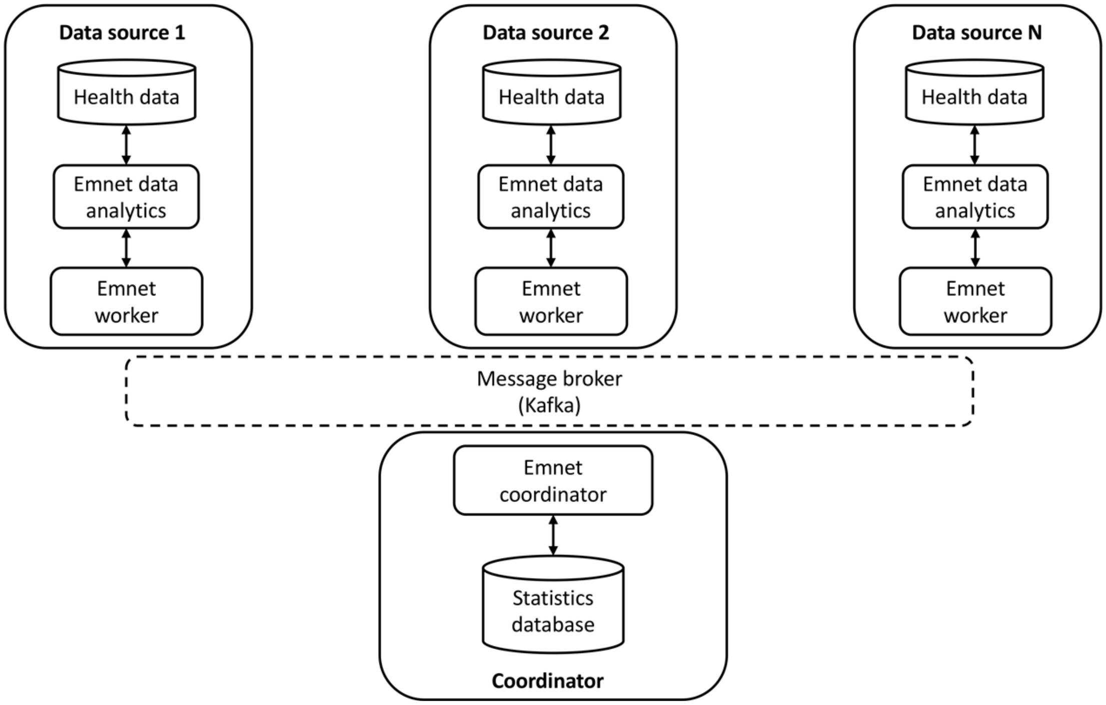


Figure 2: The architectural design of a privacy-preserving distributed statistical computation tool for computing on distributed health data.

2.6.1 Secure communication

Secure communication between Emnet workers and Emnet coordinator are supported through a message-oriented middleware. The middleware contains a message broker, Apache Kafka[35], and communication modules running on Emnet workers and Emnet coordinator. Kafka is a pull-based,

[35] Apache Kafka documentation page https://kafka.apache.org/documentation/

open-source stream-processing platform organizing messages into feeds called topics where clients subscribe and fetch messages from topics. A message sender publishes a message to a Kafka topic and the recipient consumes the message from the topic.

Apache Kafka comes with a lot of security features out of the box including secure communication between message sender/recipient and Kafka. But anyone who can access Kafka can read the message. However, Emnet requires secure end-to-end communication. To achieve this requirement, the sender must encrypt the messages before pushing them over the wire into Kafka, and the recipient needs to decrypt them upon retrieval. To that end, we used a combination of asymmetric (RSA) and symmetric (AES) encryption schemes. Symmetric encryption is preferable for its efficiency. However, it requires that both the sender and receiver use the same key to encrypt or decrypt. Therefore, asymmetric encryption is used for a secure key exchange. Asymmetric encryption uses a separate key for encryption and decryption where the encryption key is public, and the decryption key is private. Therefore, anyone can encrypt a message and only the receiver is able to decrypt it.

Table **4** shows a protocol for secure end-to-end encryption between two entities A and B. A key with two subscripts, $K_{A,B}$, is a symmetric key shared by A and B. A key with one subscript, K_A, is the public key of the corresponding entity. A private key is represented as the inverse of the public key. E_{AES} and E_{RSA} are encryptions with AES and RSA, respectively, and D_{AES} and D_{RSA} are decryptions.

A message sender creates a random symmetric key and encrypts a message with a symmetric encryption algorithm; then encrypts the symmetric key with the public asymmetric key of the message consumer. Once the key is asymmetrically encrypted, the message sender sends it to Kafka together with the encrypted message. The recipient gets the encrypted key, decrypts it with its private key, and uses the key to decrypt the message (see **Table 4**).

Message sender A	*Message receiver B*
1. $K_{A,B}$: generate_symmetric_key()	
2. encrypt_message: E_{AES}(message, $K_{A,B}$)	
3. encrypt_key: E_{RSA}($K_{A,B}$, K_B)	
4. sign_encrypted_message: Sign(E_{AES}(message, $K_{A,B}$), K_A^{-1})	
5. send_message: E_{AES}(message, $K_{A,B}$)|| E_{RSA}($K_{A,B}$, K_B)||Sign(E_{AES}(message, $K_{A,B}$), K_A^{-1})	
	6. verify_signature:Verify(Sign(E_{AES}(message, $K_{A,B}$), K_A^{-1}), K_A, E_{AES}(message, $K_{A,B}$))

	7. decrypt_symmetric_key: $D_{RSA}(E_{RSA}(K_{A,B}, K_B), K_B^{-1})$ 8. decrypt_message: $D_{AES}(E_{AES}(\text{message}, K_{A,B}), K_{A,B})$

Table 4: A secure end-to-end encryption protocol.

2.6.2 Common data model

We assume that participating healthcare institutions may store their data using different data formats and standards. Therefore, to be able to run standardized programs, the data of each healthcare institution must be transformed into a common data model, a data structure that standardizes data across healthcare institutions. Emnet uses an existing common data model defined in[36].

2.6.3 Emnet data analytics

Emnet data analytics provides a restful web service interface to accept queries (dataset definition and statistics) from a local Emnet worker and return query results. A data analytics study computes on a subset of data collected by a data source. Therefore, Emnet allows defining the dataset for a data analytics project. Emnet data analytics accepts a data definition query identified with a project id and stores the query in the health database. Statistics query contains a project id that identifies the project dataset on which the query should be executed. **Table 5** describes the endpoints provided by the Emnet data analytics web service interface.

URL	**Parameters**	**Output description**
POST /emnet/dataset/	<dataset definition>	Returns the query status (if the query is successful).
POST /emnet/statistics/	<statistics query>	Returns statistics result.

Table 5: Emnet data analytics service interface endpoints.

2.6.4 Emnet worker

Emnet worker receives queries (dataset definition and statistics) from Emnet coordinator and locally executes a query using Emnet data analytics services. Then, it jointly executes a secure

[36] Bellika JG, Henriksen T, Hurley J, Marco-Ruiz L, Yigzaw KY, Hailemichael MA. Requirements to the data reuse application programming interface for electronic health record systems. Norwegian Centre for E-health Research; 2017. Available at: https://ehealthresearch.no/rapporter/requirements-to-the-data-reuse-application-programming-interface-for-ehr

summation protocol on the local results of a statistics query with other Emnet workers and coordinator. It also sends the status of a query to the Emnet coordinator.

2.6.5 Emnet coordinator

Emnet coordinator provides a restful web service interface to accept queries (dataset definition and statistics) from users. Emnet is planned for computing a large amount of data distributed across multiple healthcare institutions, therefore, it may require a long time to complete a computation. However, HTTP connections have a timeout because every open connection allocates a certain amount of memory at the server and the client. In addition, the longer the server takes to respond to the client, the higher the chances that the client may lose connection before the server has completed processing the result.

Therefore, the web service interface includes two endpoints for submitting dataset definition and statistics queries that return a response telling the client where to find the results. The client may poll the resource to GET its current progress and will eventually receive the result once the query has completed. A query result is stored in the statistics database. Since the output has its own URI, it is possible to GET it multiple times. Table 6 describes the endpoints provided by the Emnet coordinator web service interface.

URL	Parameters	Output description
POST /emnet/dataset/	<dataset definition>	Returns the project id.
POST /emnet/statistics/	<statistics query>	Returns the query id.
GET /emnet/dataset/	{project_id}	Returns the query status.
GET /emnet/statistics/	{query_id}	Returns the query status or statistics result.

Table 6: Emnet coordinator web service interface endpoints.

Emnet also allows users to configure the execution of queries on a specific time, which is stored in the statistics database. A query configuration contains a query definition and when the query should be executed.

Emnet coordinator also implements functionalities required to jointly execute secure summation protocols with Emnet workers, which are required for statistical computations. The currently implemented secure summation protocols are SINE and k-secure summation protocols.

Emnet coordinator broadcasts a query to Emnet workers and collects the status of a query from Emnet workers. The coordinator only requires the status of a dataset definition query. Section 2.7

provides detail on how statistical queries are executed. Sequence diagrams of Emnet for executing dataset definition and statistics queries are available in the

Annex. I: Sequence diagrams of Emnet.

2.7 Statistical computations

Emnet supports the computation of different statistics on data horizontally partitioned among healthcare institutions. Emnet is based on decomposing statistical problems into sub-functions of summation form that is evidenced by previous studies[Error! Bookmark not defined.,Error! Bookmark not defined.]. The following subsections describe how descriptive statistics such as count, variance, covariance, ratio, mean, standard deviation, percentile, min, max, and Pearson's r are decomposed into sub-computations of summation forms.

Let us consider N healthcare institutions that collect values of variables x_i and y_i for their patients and that the total number of patients is n. The description of the following subsections assumes that x and y are primary variables, and x_i and y_i are values of patient i for these variables, respectively. However, the techniques also support computations on derived variables.

2.7.1 Summation

The summation of private values x_i of all patients across the healthcare institutions, $sum(x) = \sum_{i=1}^{n} x_i$, can be computed in two steps (1) each participating healthcare institution D_j locally computes the sum of the private values x_i of its patients, and (2) a secure summation protocol is executed to aggregate the local results.

2.7.2 Count

Count statistics, for example the total number of eligible patients across the healthcare institutions, is computed in two steps (1) each participating healthcare institution D_j locally counts the number of patients in their institution, $count_j$, and (2) a secure summation protocol is executed to aggregate the local results.

$$count = \sum_{j=1}^{N} count_j$$

2.7.3 Ratio

Ratio statistics, $ratio$, can be decomposed into $f(x)$ and $f(y)$. If $f(x)$ and $f(y)$ are in summation forms, they can be computed as described above.

$$ratio = \frac{f(x)}{f(y)}$$

2.7.4 Mean

Mean statistics, $mean\ (x)$, is a specialized form of ratio statistics where $f(x)$ is $sum\ (x)$ and $f(y)$ is $count$ which are both in summation forms.

$$mean\ (x) = \frac{sum\ (x)}{count}$$

2.7.5 Variance

Variance statistics ((Equation 1), $var\ (x)$, can be decomposed into count, summation and mean statistics. The computation of count and mean statistics are straight forward as described above. Once the mean statistics is computed, each healthcare institution locally computes a derived variable $(x_i - mean\ (x))^2$ based on x_i and $mean(x)$. Then, the summation of the derived variable is computed as described above.

$$var\ (x) \ = \frac{1}{count} \sum_{i=1}^{n} (x_i - mean\ (x))^2 \qquad \text{(Equation 1)}$$

Once a variance is calculated, standard deviation, $sdv(x)$, can be calculated by taking the square root of the variance.

$$sdv\ (x) \ = \sqrt{var\ (x)}$$

2.7.6 Covariance

Covariance ((Equation 2), $covar\ (x, y)$, can be decomposed into count, summation and mean statistics. The computation of count and mean of x and y are straight forward as described above. Once the mean statistics is computed, each healthcare institution locally computes two derived variables such as (1) $(x_i - mean\ (x))$ based on x_i and $mean(x)$ and (2) $(y_i - mean\ (y))$ based on y_i and $mean(y)$. Then, summations of the derived variables are computed as described above.

$$covar\ (x, y) \ = \frac{1}{count} \sum_{i=1}^{n} (x_i - mean\ (x))(y_i - mean\ (y)) \qquad \text{(Equation 2)}$$

2.7.7 Pearson's r

(Equation 3 shows the Pearson's r statistics. **Error! Reference source not found.** shows how Pearson's r can be decomposed into covariance $covar\ (x, y)$ and variance statistics such as $var\ (x)$ and $var\ (y)$.

$$r\ (x, y) \ = \frac{\sum_{i=1}^{n} (x_i - mean\ (x))(y_i - mean\ (y))}{\sqrt{\sum_{i=1}^{n} (x_i - mean\ (x))^2 \sum_{i=1}^{n} (y - mean\ (y))^2}} \qquad \text{(Equation 3)}$$

$$r\ (x,y)\ = \frac{\frac{\sum_{i=1}^{n}(x_i - mean\ (x))(y_i - mean\ (y))}{count}}{\frac{\sqrt{\sum_{i=1}^{n}(x_i - mean\ (x))^2 \sum_{i=1}^{n}(y - mean\ (y))^2}}{count}}$$

$$r\ (x,y) = \frac{\frac{\sum_{i=1}^{n}\left(x_i - mean\ (x)\right)\left(y_i - mean\ (y)\right)}{count}}{\sqrt{\frac{\sum_{i=1}^{n}\left(x_i - mean\ (x)\right)^2 \sum_{i=1}^{n}\left(y - mean\ (y)\right)^2}{count * count}}}$$

$$r\ (x,y)\ = \frac{covar\ (x,y)}{\sqrt{var\ (x) var\ (y)}} \qquad \text{(Equation 4)}$$

2.7.8 Percentile

Percentile p is a measure used in statistics indicating the mth value below which p^{th} percentage of values among values distributed across healthcare institutions, where $m = \left(\frac{p}{100}\right) \times\ n.$ For example, the 25th percentile among 200 values is the 50th value. Therefore, the problem of computing percentile is translated into computing the mth-ranked element.

Unlike the statistics described above, the problem of finding an mth-ranked element involves multiple iterations. There has been an existing secure protocol for computing mth-ranked element[37]. We developed a new mth-ranked element protocol that converge with a significantly small number of iterations than the existing protocol[37].

The basic steps of the protocol are described in Table 7. However, the protocol's optimization and its proofs will be published in a peer reviewed scientific publication.

[37] Aggarwal G, Mishra N, Pinkas B. Secure computation of the kth-ranked element. In: Advances in Cryptology - EUROCRYPT 2004. Lecture Notes in Computer Science. Springer, Berlin, Heidelberg; 2004:40-55.

Input: Each healthcare institution D_j, $1 \leq j \leq N$, has a set of values $\mathcal{R}_j$ of n_j elements. The parameters of the protocol are initialized as $x_{\text{lower}} = a$ and $x_{\text{upper}} = b$. The parameter n_{lower} is the number of elements smaller than x_{lower} and n_{upper} is the number of elements less than or equal to x_{upper}. Iteration count i is initialized with 0.

Output: the mth-ranked element.

1. The coordinator sets the iteration number $i = i + 1$
2. For each healthcare institution D_j:

   If there are elements of $\mathcal{R}_j$ that are in the range $(x_{\text{lower}}, x_{\text{upper}}]$

   (a) compute the median of the elements of $\mathcal{R}_j$ in the range $(x_{\text{lower}}, x_{\text{upper}}]$, m_j^i

   (b) set c_j^i to 1

   Else if there is no element of $\mathcal{R}_j$ are in the range $(x_{\text{lower}}, x_{\text{upper}}]$

   (a) set c_j^i and m_j^i to 0
3. The healthcare institutions and coordinator jointly execute a secure summation protocol for computing:

   (a) $m^i = \sum_{j=1}^{N} m_j^i$

   (b) $c^i = \sum_{j=1}^{N} c_j^i$
4. The coordinator locally computes the pivot $P = {}^{m^i}\!/_{c^i}$ and broadcast P to all healthcare institutions
5. Each healthcare institution D_j locally computes the number of elements in $\mathcal{R}_j$ that are less than or equal to P, denoted l_j^i
6. The healthcare institutions and coordinator jointly execute a secure summation protocol for computing $l^i = \sum_{j=1}^{N} l_j^i$
7. The coordinator:

   (a) outputs “the pivot P” if $l^i = \mathrm{n}_{\text{upper}}$ (This means the pivot P is the m^{th}-ranked element)

   (b) if $(l^i \geq m)$, sets $\mathrm{x}_{\text{upper}} = P$ and $\mathrm{n}_{\text{upper}} = l^i$, broadcasts $\mathrm{x}_{\text{upper}}$ to all Each healthcare institutions, and goes to step 1

   (c) if $(l^i < m)$ sets $\mathrm{x}_{\text{lower}} = P$ and $\mathrm{n}_{\text{lower}} = l^i$, broadcasts $\mathrm{x}_{\text{lower}}$ to all Each healthcare institutions, and goes to step 1

Table 7: The secure mth-ranked element protocol.

Table 8 presents the protocol’s steps for computation of the 3rd-ranked element among values distributed across three healthcare institutions.

	$\boldsymbol{D_1}$	$\boldsymbol{D_2}$	$\boldsymbol{D_3}$	**Coordinator**
Inputs	$\mathcal{R}_1 = \{1,3\}$ $(x_{\text{lower}}, x_{\text{upper}}] =$ (-10,10]	$\mathcal{R}_2 = \{4,6\}$ $(x_{\text{lower}}, x_{\text{upper}}] =$ (-10,10]	$\mathcal{R}_3 = \{3,4,6\}$ $(x_{\text{lower}}, x_{\text{upper}}] =$ (-10,10]	$m = 3$ $(x_{\text{lower}}, x_{\text{upper}}] =$ (-10,10] $[n_{\text{lower}}, n_{\text{upper}}] =$ [0,7]
Iteration $i = 1$				

Step 3	$m_1^1 = 2, c_1^1 = 1$	$m_2^1 = 5, c_2^1 = 1$	$m_3^1 = 4, c_3^1 = 1$	
Step 4				$m^1 = 2 + 5 + 4 = 11$
				$c^1 = 1 + 1 + 1 = 3$
				$P = {}^{11}/_{3} = 3.667$
	$P = 3.667$	$P = 3.667$	$P = 3.667$	
Step 5	$l_1^1 = 2$	$l_2^1 = 0$	$l_3^1 = 1$	
Step 6				$l^1 = 2 + 0 + 1 = 3$
				$l^1 \geq m, x_{\text{upper}} = 3.667, n_{\text{upper}} = 3$
	$x_{\text{upper}} = 3.667$	$x_{\text{upper}} = 3.667$	$x_{\text{upper}} = 3.667$	
Iteration i $= 2$				
Step 3	$m_1^2 = 2, c_1^2 = 1$	$m_2^2 = 0, c_2^2 = 0$	$m_3^2 = 3, c_3^2 = 1$	
Step 4				$m^2 = 2 + 0 + 3 = 5$
				$c = 1 + 0 + 1 = 2$
				$P = {}^{5}/_{2} = 2.5$
	$P = 2.5$	$P = 2.5$	$P = 2.5$	
Step 5	$l_1^2 = 1$	$l_2^2 = 0$	$l_3^2 = 0$	
				$l^2 = 1 + 0 + 0 = 1$
Step 6				$l^2 < m, x_{\text{lower}} = 2.5, n_{\text{lower}} = 1$
	$x_{lower} = 2.5$	$x_{lower} = 2.5$	$x_{lower} = 2.5$	
Iteration i $= 3$				
Step 3	$m_1^3 = 3, c_1^3 = 1$	$m_2^3 = 0, c_2^3 = 0$	$m_3^3 = 3, c_3^3 = 1$	
Step 4				$m^3 = 3 + 0 + 3 = 6$
				$c^3 = 1 + 0 + 1 = 2$
				$P = {}^{6}/_{2} = 3$
	$P = 3$	$P = 3$	$P = 3$	
Step 5	$l_1^3 = 2$	$l_2^3 = 0$	$l_3^3 = 1$	
Step 6				$l^3 = 2 + 0 + 1 = 3$
				$l^3 = n_{\text{upper}} = 3$, the m^{th}-ranked element is $P = 3$

Table 8: Example execution of the secure 3^{rd}-ranked element protocol among three healthcare institutions.

2.7.9 Min

Min value of values distributed across healthcare institutions can be computed with the mth-ranked element protocol for $m = 1$.

2.7.10 Max

Max value of values distributed across healthcare institutions can be computed with the mth-ranked element protocol for $m = \text{n}$.

2.8 Disclosure control

Emnet ensures that statistics are computed on the combined data for a group of data sources without revealing anything apart from the output (Section 2.9). However, statistics results might lead to inferential disclosure given prior knowledge and repeated queries. For example, the result of a query counting the number of cancer patients age greater than 110 years may reveal the diagnosis of a patient given a background knowledge of the oldest person in the community.

Statistical disclosure limitation techniques have been developed to protect inferential disclosure. These techniques are classified into query restriction where one or more queries are denied from a sequence of queries and perturbation where query results are perturbed in such a way that privacy is protected[38]. However, these techniques were developed for a centrally collected data. Further research is needed for developing disclosure control techniques for distributed data in the context of Emnet.

2.9 Security analysis

Emnet uses exiting secure summation protocols such as the SINE[30] and the k-secure summation[31] protocols. Their security is briefly discussed in Section 2.3. The composition theorem shows that computing a function by composing secure sub-protocols is secure[20]. Therefore, if a summation protocol is proved to be secure, then the computation of statistical problems such as count, variance, covariance, ratio, mean, standard deviation, Pearson's r, percentile, min, and max, which are summation-based, is secure.

2.10 Implementation

Figure 2 shows the architectural design of Emnet. The Emnet data analytics service was implemented in Python, and its REST API was implemented using the Django Rest Framework[39]. The Emnet data analytics service computes against health data stored in a MySQL database using an existing common data model[36]. The Emnet worker and Emnet coordinator were implemented in Java using actor-based model, an abstraction that makes it easier to write concurrent, parallel, and distributed systems, provided by Akka framework[40]. We used Apache Kafka as a message broker[35]. In Annex. I: Sequence diagrams of Emnet, it is demonstrated how different actors jointly execute dataset definition and statistics queries. Emnet coordinator uses Spring Boot framework[41] for the implementation of the REST API and Quartz library[42] for schedule execution of queries.

[38] Adam NR, Worthmann JC. Security-control Methods for Statistical Databases: A Comparative Study. ACM Comput Surv 1989; 21:515–556.

[39] Django REST framework https://www.django-rest-framework.org

[40] Akka framework https://akka.io/

[41] Spring Boot https://spring.io/projects/spring-boot

[42] Quartz job scheduling library http://www.quartz-scheduler.org/

3 Data Owners' Privacy Analytics

Electronic Medical Records have been developed to integrate patient data and help healthcare professionals to make the best decisions[43]. Despite the benefits achieved with all the data collected, shared and available during healthcare practice, several challenges are faced by the healthcare organizations, namely, ensuring patient privacy and the right use of patient data[44]. Since 1995/96, the concern on personal data protection has been stated on laws such as the "Data Protection Act"[45] and "Health Insurance Portability and Accountability Act"[46]. Nonetheless, in May 2018, the GDPR[47] came to reinforce the processes of personal data protection. Under the GDPR, healthcare organizations have the obligation of demonstrating accountability for the fulfilment of the regulation requirements, which relies on their ability to demonstrate that appropriate procedural security measures are being applied and, most importantly, that they are compliant with the GDPR[48]. Moreover, the GDPR includes article 12 "Transparent information, communication and modalities for the exercise of the rights of the data subject"[49], which is dedicated to data processing transparency for the patients, who are the data subjects of an EMR.

Task 2.6 "Data Owners' Privacy Analytics" is dedicated to addressing such transparency demands. Privacy analytics refers to providing information to the patients about the operations on their data, and to verify whether the data processing was legitimate. A patient's data processing event generates system and application logging information about how, by whom, and under what circumstances the event happened. For applications built using the ASCLEPIOS framework, this information is directly related to access control policies and how they are enforced. The policy enforcement component in ASCLEPIOS records logs of all attempted and authorized data access operations, and these logs are further used to implement privacy analytics functions that enable transparency towards the data subject and his/her representatives. A new module is proposed here to perform these analytics functions in the ASCLEPIOS framework, consolidating the work carried out in the context of Task 2.6.

The task aims to research and define mechanisms for privacy analytics in the scope of applications built using the ASCLEPIOS framework. This essentially consists of logging and disclosing information about the patient data access operations (5W1H) to the patient or to the

[43] R. J. Cruz-Correia, P. M. Vieira-Marques, A. M. Ferreira, F. C. Almeida, J. C. Wyatt, And A. M. Costa-Pereira, "Reviewing The Integration Of Patient Data: How Systems Are Evolving In Practice To Meet Patient Needs," Bmc Med. Inform. Decis. Mak., 2007.

[44] A. Ferreira, R. Cruz-Correia, L. Antunes, D. Chadwick, "Access Control: How Can It Improve Patients Healthcare?" Stud. Heal. Technol. Inform., Vol. 127, Pp. 65–76, 2007.

[45] "Data Protection Act", Official Journal L281.P.0031-0050,1995.

46 U. States, "Health Insurance Portability and Accountability Act Of 1996. Public Law 104-191" Us Statut. Vol.110, Pp.1936 –2103, 1996.

[47] The General Data Protection Regulations (GDPR) are European Union regulations and can be found here: European Parliament and Council of European Union (2016) Regulation (EU) 2016/679.

[48] Correia, Liliana Sá, Ricardo Cruz Correia, and Pedro Pereira Rodrigues. "Illegitimate HIS access by healthcare professionals: scenarios, use cases and audit trail-based detection model." Procedia Computer Science 164 (2019): 629-636.

[49] Art. 12 GDPR "Transparent information, communication and modalities for the exercise of the rights of the data subject" https://gdpr-info.eu/art-12-gdpr/

DPO. In ASCLEPIOS, these operations consist of data access requests that are performed through Attribute-Based Access Control (ABAC) and Attribute-Based Encryption (ABE) schemes. Therefore, in this task, the following main activities were carried out:

- understand the components and communication flow for attribute-based access control considering data access requests,
- identify the points for logging information about these requests,
- define the information to be logged and the entity responsible for their storage and access control,
- define mechanisms to provide privacy analytics functions to patients and DPOs, and offer these through a new ASCLEPIOS module,
- considering the ASCLEPIOS framework was not available yet at the time of writing, develop a prototype implementation of the new module to demonstrate the proposed mechanisms.

The results of these activities are documented in the next sections.

The main goal of the ASCLEPIOS Privacy Analytics Module (APAM) is to provide tools to enable checking if data processing is taking place as intended and authorized. The module addresses the perspectives of two main stakeholders: the patient (data subject of the EMR) and the data protection officer (DPO) of the healthcare organization (data controller of the EMR). APAM includes functions to retrieve and present data access history, as well as functions to detect abnormal or illegitimate operations. Furthermore, APAM offers functions to discover exceptions that indicate erroneous or illegitimate access, as well as to assist in identifying the policies that permitted this access, such that these can be revised to prevent future illegitimate data processing. APAM offers a set of analytics functions that can be used by healthcare organizations to comply with the GDPR and to check the efficiency and effectiveness of the access control policies. Moreover, APAM offers interfaces based on the audit logs to reveal insights for the patient about the usage of his/her data in a user-friendly fashion through an interactive interface.

This section describes the results of work carried out in Task 2.6. We first present the background information about the task and its motivation for GDPR compliance and Transparency enhancing tools in Section 5.1. In Section 5.2, we summarize the data flow for access control in ASCLEPIOS and discuss the opportunities for logging the data processing events. In Section 5.3, we define the audit logs, where they are stored and who can process them. In Section 5.4, we present the architecture and components of APAM and describe its prototype implementation with some demonstration of its use. In Section 5.5, we discuss the current results and future work.

3.1 Background

Here we cover various background aspects regarding privacy analytics: stakeholders, privacy auditability requirements, requirements raised for the ASCLEPIOS framework, and the GDPR transparency requirements.

3.1.1 Stakeholders

The goal of Task 2.6 is "*to define and serve a set of analytics to "data owners", and to allow them to easily and visually understand how third parties (and which ones) manage their data*". However, ownership over the electronic medical records is still an open debate between the patients and the healthcare providers. According to the GDPR, there are four fundamental roles regarding personal data processing: (1) the data controller, (2) the data protection officer, (3) the data subject, and (4) the data processor. Below we discuss how these roles are exercised in a typical healthcare situation.

(1) The data controller is the healthcare organization. Together with one or more organizations, it jointly determines "why" and "how" personal data should be processed. Healthcare organizations are classified as joint controllers and must enter an arrangement setting out their respective responsibilities for complying with the GDPR rules designed specifically for healthcare data processing. Moreover, the data controller determines who shall be responsible for compliance with data protection rules and how data subjects can exercise their rights in practice; in other words, to allocate responsibility[50].

(2) The data protection officer (DPO) is responsible for monitoring the compliance of the data controller to the GDPR. The DPO monitors all the core activities of the controller or the processor. This systematic and regular monitoring takes place at large scale due to the virtue of the nature, scope, and purposes of the data processing activity. The DPO requires expert knowledge and adequate access to logs about the personal data processed by the controller or the processor. Such data protection officers, whether they are an employee of the controller, should be able to perform their duties and tasks independently[51].

(3) The data subject, or the patient of the EMR, is an identifiable natural person who can be identified, directly or indirectly, by reference to a name, an identification number, location data, an online identifier or to one or more factors specific to the physical, physiological, genetic, mental, economic, cultural or social identity of that natural person[52].

(4) The data processor processes personal data only on behalf of the data controller. The processing of personal data in the healthcare case should also be regarded to be lawful when it is necessary to protect an interest essential for the life of the data subject or of another natural person. Processing is also necessary for compliance with a legal obligation to which the controller is subject, for the performance of a task carried out in the public interest or the exercise of official authority vested in the controller, and also when the data subject has given consent for the processing of his or her personal data for one or more specific purposes[53]. "Processing" means

[50] GDPR Art. 29 data protection working party https://ec.europa.eu/justice/article-29/documentation/opinion-recommendation/files/2010/wp169_en.pdf

[51] GDPR Recital 97 "Data Protection Officer" https://gdpr-info.eu/recitals/no-97/

[52] Art. 4 GDPR "Definitions" https://gdpr-info.eu/art-4-gdpr/

[53] Art. 6 GDPR "Lawfulness of processing" https://gdpr-info.eu/art-6-gdpr/

any operation or set of operations performed on (sets of) personal data, whether by automated means, such as create, read, update, and delete (CRUD)[54].

3.1.2 Privacy auditability requirements

The auditability requirements were collected and identified through a combination of interviews conclusions from the Deliverable D1.1 "Technical, Security, Healthcare and Data Privacy Requirements" and analyzing both legal requirements from the GDPR and standards (ISO 18308 and ISO 27001). This collection of requirements guided the modelling of APAM and the ASCLEPIOS framework towards complying with the GDPR.

H-AUD1. A cloud-based e-health framework SHALL maintain an audit trail of personal data processing.

H-AUD2. *A cloud-based e-health framework SHALL specifically identify access that has overridden policies (e.g., in a medical emergency situation)*.

H-AUD3. *A cloud-based e-health framework SHALL protect the integrity of the audit trail.*

H-AUD4. *A cloud-based e-health framework SHALL enable authorized access to the audit trail*.

H-AUD5. *A cloud-based e-health framework SHALL ensure the audit trail maintains records of disclosures of the audit trail itself."*

3.1.3 Transparency and trust

Article 12 of the GDPR "Transparent information, communication and modalities for the exercise of the rights of the data subject"[55] is dedicated to transparency for the data subject, which in the scope of ASCLEPIOS is the patient. The GDPR recitals transcript below explains the principle of the transparency, procedures for the exercise of the right of the data subjects and information obligations.

- *GDPR Recital 58 "The Principle of Transparency"[56]:* **"**The principle of transparency requires that any information addressed to the public or the data subject be concise, easily accessible and easy to understand, and in clear and plain language and, additionally, where appropriate, visualization be used. Such information could be provided in electronic form, for example, when addressed to the public, through a website. This is of particular relevance in situations where the proliferation of actors and the technological complexity practice making it difficult for the data subject to know and understand whether, by whom and for what purpose personal data relating to him or her are being collected, such as in

[54] Art. 4 GDPR "Definitions" https://gdpr-info.eu/art-4-gdpr/

[55] Art. 12 GDPR "Transparent information, communication and modalities for the exercise of the rights of the data subject" https://gdpr-info.eu/art-12-gdpr/

[56] GDPR Recital 58 "The Principle of Transparency" https://gdpr-info.eu/recitals/no-58/

the case of online advertising. Given that children merit specific protection, any information and communication, where processing is addressed to a child, should be in such a clear and plain language that the child can easily understand."

- *GDPR Recital 59 "Procedures for the Exercise of the Rights of the Data Subjects"[57]:* "Modalities should be provided for facilitating the exercise of the data subject's rights under this Regulation, including mechanisms to request and, if applicable, obtain, free of charge, in particular, access to and rectification or erasure of personal data and the exercise of the right to object. The controller should also provide means for requests to be made electronically, especially where personal data are processed by electronic means. The controller should be obliged to respond to requests from the data subject without undue delay and at the latest within one month and to give reasons where the controller does not intend to comply with any such requests".
- *GDPR Recital 60 "Information Obligation"[58]:* "The principles of fair and transparent processing require that the data subject be informed of the existence of the processing operation and its purposes. The controller should provide the data subject with any further information necessary to ensure fair and transparent processing, considering the specific circumstances and context in which the personal data are processed. Furthermore, the data subject should be informed of the existence of profiling and the consequences of such profiling. Where the personal data are collected from the data subject, the data subject should also be informed whether he or she is obliged to provide the personal data and of the consequences, where he or she does not provide such data. That information may be provided in combination with standardized icons to give in an easily visible, intelligible, and clearly legible manner, a meaningful overview of the intended processing. Where the icons are presented electronically, they should be machine-readable."

In addition to the GDPR requirements about transparency, the requirements for the data access logs contents to fulfil the transparency requirements of data processing have been expressed according to the 5W1H principle model[59,60] (what, when, where, by whom, why and how). These are explained in
Table **9**, which summarizes various recommendations found in the literature[61].

[57] GDPR Recital 59 "Procedures for the Exercise of the Rights of the Data Subjects" https://gdpr-info.eu/recitals/no-59/

[58] GDPR Recital 60 "Information Obligation" https://gdpr-info.eu/recitals/no-60/

[59] G. Yang, L. Cai, A. Yu and D. Meng, "A General and Expandable Insider Threat Detection System Using Baseline Anomaly Detection and Scenario-Driven Alarm Filters," 2018 17th IEEE International Conference On Trust, Security And Privacy In Computing And Communications/ 12th IEEE International Conference On Big Data Science And Engineering (TrustCom/BigDataSE), New York, NY, 2018, pp. 763-773, doi: 10.1109/TrustCom/BigDataSE.2018.00110

[60] Son, Jiseong et al. 'Dynamic Access Control Model for Privacy-Preserving Personalized Healthcare in Cloud Environment'. 1 Jan. 2016: S123 – S129

[61] Abbas A, Khan SU. A review on the state-of-the-art privacy-preserving approaches in the e-health clouds. IEEE J Biomed Health Inform. 2014;18(4):1431-1441. doi:10.1109/JBHI.2014.2300846

What data was processed?	The activity objects and the optional attributes; The logs must have the data subject identifier and the data source/entity.
When did data processing happen?	The event time or period.
Where did data processing happen?	The event location, such as device, IP, and department.
Who processed the data?	The processor, such as username, group ID and role.
Why was the data processed?	Logged events must be attributable to identified users for provenance purposes. The processor should provide the purpose of the request. It could be implied on the request form, or it needs to be submitted by the processor.
How was the data processed?	The activity type including the domain type and operation, such as CRUD.

Table 9: 5W1H principle model applied to healthcare data processing for transparency purpose.

In the literature, Explanation-for-trust[62] is a term used for mechanisms to display how the data processing works and is secured, by revealing details of the measures used to guarantee compliance to the privacy requirements. Transparency-Enhancing Technologies (TETs)[63] are mechanisms that provide information, or explanations, that can help to fulfil transparency requirements. TETs make the underlying processes more transparent and enable data subjects to understand better the implications that arise due to their decision to disclose personal data. Multiple approaches are using TET following *preventive*, *detective,* and *corrective* approaches.

In the scope of this task, we consider two main stakeholders: patients and DPO. Below we summarize preventive, and detective approaches for increasing transparency for these stakeholders.

Preventive approach

Patients (or their legal representatives) need upfront information about what is being done with their data, and why this is done. From the patient perspective, this can be a concise statement that persons without a technical or legal background should be able to understand. In addition to this statement, a link to an online resource where more detailed information can be obtained. A

[62] Pieters, W. Explanation and trust: what to tell the user in security and AI? Ethics Inf Technol 13, 53–64 (2011). https://doi.org/10.1007/s10676-010-9253-3

[63] Zimmermann, C. (2015). A categorization of Transparency-Enhancing Technologies. preprint arXiv:1507.04914.

description of the process used to grant or deny access to medical records can also be presented to patients, as well as information about the security measures and the continued monitoring of possible confidentiality breaches.

A DPO relies on the assessment of user action and interaction to monitor and prevent a potential privacy breach[64]. This can be done through (automatic) means to identify outlier behaviour followed by a review of access policies and the provisioning of attributes. This requires full access to data access activity logs. No clear malicious intent can be determined in the preventive approach. However, this process is fundamental to the detective approach discussed below.

Detective approach

Patients need basic statistics about how their data are being used following the 5W1H, as presented in

Table **9**. This enables the patient to detect whether access to the data conforms to expectation. For example, if his/her data is accessed from a healthcare center that the patient has not visited. This functionality can also be present in a user-friendly interface, where she/he checks if the access policies presented in the preventive approach are fulfilled.

Detection of (potential) data breaches is an important task of a DPO. Using the DPOs expertise about legitimate access, the analysis of audit trail logs can be used to detect anomalous events, based on previously identified malicious events or outlier's detection mechanisms. Such events can be investigated further and serve as input for future detection as well.

Anomaly detection is the identification of rare observations that differ from the general distribution of a population. It is based on the premise that anomalous behaviour substantially deviates from normal behaviour[65]. The efficiency of detecting such behaviour when accessing sensitive data plays a crucial role in applying security methods. Conducting anomaly detection with support of machine learning tools can help further automation. Machine learning algorithms can be applied with or without supervision. Supervised anomaly detection requires a dataset with labelled data, both normal and anomalous samples[66]. As few to no labelled data are typically available, unsupervised learning methods are commonly used.

Unsupervised anomaly detection, or outlier detection, requires no labelled data[67]. This type of method is not used to predict future behaviour, but to identify groupings and find rare events or

[64] Boss, Scott R., et al. "If someone is watching, I'll do what I'm asked: mandatories, control, and information security." *European Journal of Information Systems* 18.2 (2009): 151-164.

[65] Alex Witkamp. Applying Unsupervised Learning on Hospital Audit Logs for Anomaly Detection. Master's thesis, University of Amsterdam, July 2020.

[66] Li Sun, Steven Versteeg, Serdar Boztas, and Asha Rao. Detecting anomalous user behaviour using an extended isolation forest algorithm: An enterprise case study. CoRR, abs/1609.06676, 2016.

[67] Gareth James, Daniela Witten, Trevor Hastie, and Robert Tibshirani. An Introduction to Statistical Learning. Springer New York, 2013.

anomalies based on the assumption that normal data instances are the most common. The approach can be extended through semi-supervised learning or novelty detection[68].

3.2 ASCLEPIOS access control data flow

In this section, we present an overview of the ASCLEPIOS access control mechanisms for safeguarding the EMR systems, which are fully described Deliverable D3.2 "ASCLEPIOS Models Editor and *Interpretation* Mechanism". We first summarize the most relevant concepts for the privacy analytics task, with a focus on the data and control flow that takes place for data access authorization. Secondly, we discuss a proposal for capturing logs in the ASCLEPIOS framework.

3.2.1 Access Control Data Flow

In ASCLEPIOS, data access control is based on two types: Attribute-based Access Control (ABAC) and Attribute-based Encryption (ABE). The ABAC mechanism protects data by restricting attempts/requests to process the EMR based on the attributes of the requestor as well as the data resource. The characteristics of requests, resources and access operations can dynamically involve contextual attributes. ABE is used along with SSE, to build a hybrid encryption scheme. First, a user generates a symmetric key for a Symmetric Searchable Encryption (SSE) scheme and encrypts her data locally before outsourcing them to the CSP to be stored in an encrypted database. Then, she encrypts the SSE key using ABE and binds a policy to the resulting ciphertext. The ciphertext is also uploaded to the CSP to be stored in the KeyTray enclave. Finally, if a user wishes to access the encrypted data, she first needs to request access to the encrypted key, which she will be able to decrypt if and only if her attributes satisfy the policy bound on the ciphertext of the symmetric key.

The model in Figure 3 is a high-level representation of the data flows within the access control system on the ASCLEPIOS framework. It is based on two different layers of authorization control. First, the ABAC layer permits or denies access encrypted EMRs, governed by the policies defined through "ASCLEPIOS Models and PoLicies Editors" (AMPLE) which is described in Deliverable D3.2 "ASCLEPIOS Models Editor and *Interpretation* Mechanism". Second, the ABE layer handles the encryption of the SSE key described on Deliverable D2.2[69]. For each user and each access, the access control process depicted in **Figure 3** is repeated. In this model, the User Interface represents an application that uses the ASCLEPIOS framework to implement secure access to the EMR.

The access control process illustrated in Figure 3 goes as follows. First, a user authenticates through the communication user interface providing her credentials to the Registration Authority (RA). After being authenticated, the RA collects the user attributes from the attribute provisioner and sends them along with the authorization token and the user's ABE key to the user. For any

[68] Xiaojin Zhu and Andrew B. Goldberg. Introduction to semi-supervised learning. Synthesis Lectures on Artificial Intelligence and Machine Learning, 3(1):1–130, January 2009.

[69] Ruben Groot Roessink, Arash Vahidi, Alexandros Bakas, Alexandr Zalitko, Antonis Michalas. Deliverable D2.2 "Attribute-Based Encryption, Dynamic Credentials and Ciphertext Delegation and Integration in Medical Devices"

data processing request, the user attributes are passed on to the granular retrieval service that determines if access should be granted according to the defined policies. Upon gaining access permission from the ABAC policy enforcement, the user can start interacting with the system by sending CRUD requests to the granular retrieval. The data processing is allowed according to the users' access rights. After that, to be able to process the data, the user must decrypt the symmetric key using her ABE key, and then use the symmetric key to decrypt and encrypt data.

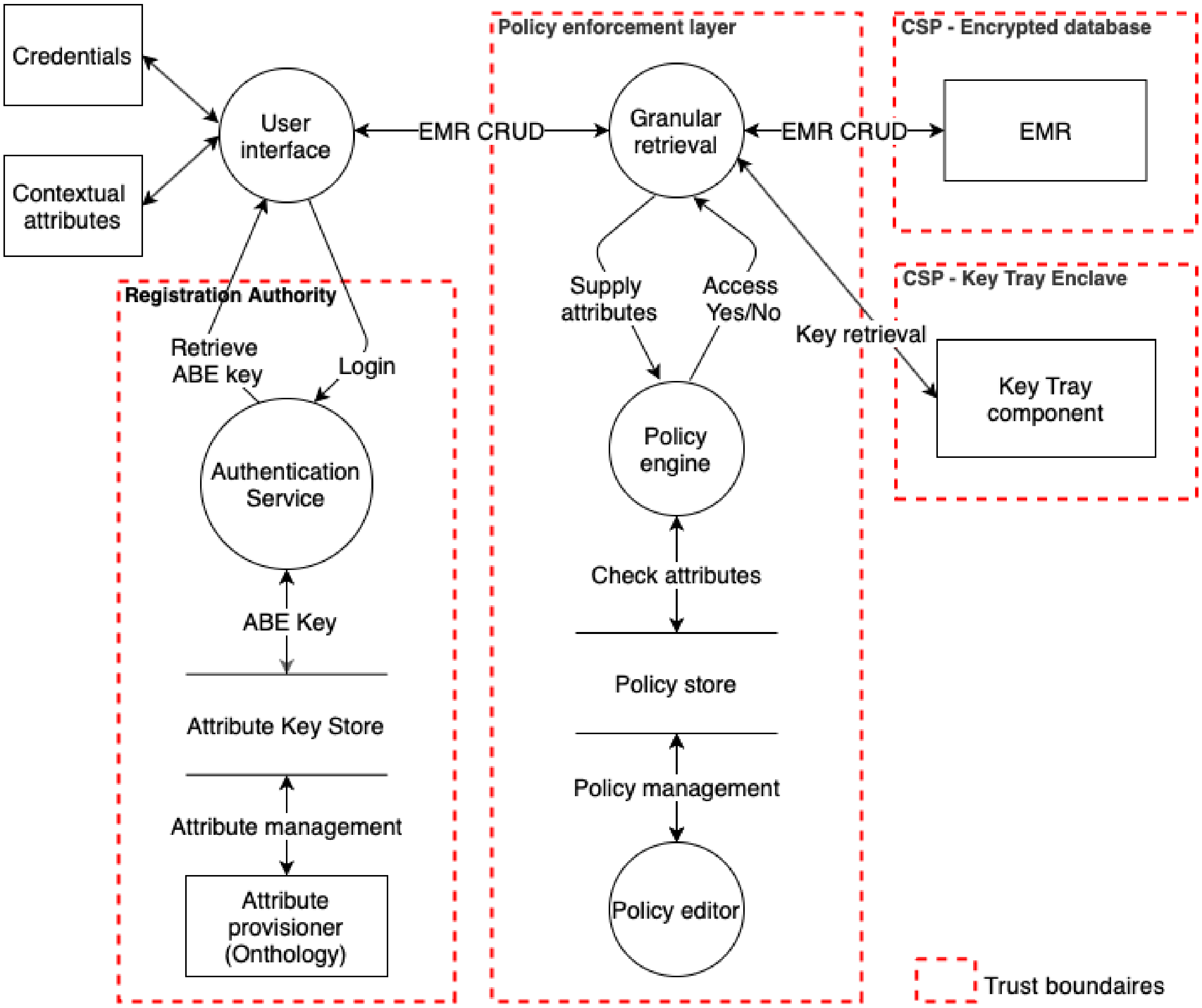


Figure 3: Data flow diagram of ASCLEPIOS attribute-based access control mechanism.

3.2.2 Logs capture

The data flow model presented in Figure 3 was used for discussions to identify the best source and location to capture and store the audit logs. Within the current ASCLEPIOS setup, however, no ASCLEPIOS services offer audit logs, mainly due to security concerns. Therefore, currently, the user interface is the only single point at which all information can be captured about how users access the EMR system.

After interactions among the project partners, it was decided that the audit logs will be captured in the user interface and stored at the RA's server because the RA already stores the user attributes and authenticates the users. Also, this also enables other components to consume them for monitoring and analytics. A potential component to use the audit logs from the RA is the "ASCLEPIOS Cybersecurity, Encryption and Access Analytics for Healthcare Providers" (CEAA) module presented in the Deliverable D2.3 "GDPR-compliant and Privacy-Preserving Analytics for Healthcare Providers". The CEAA is responsible for delivering insights to system administrators about encryption and decryption activities, data access patterns, normal and abnormal behaviours, cyber threats, and security incidents. Note that CEAA does not cover analytics from the individual perspective of each patient but works based mainly on aggregated data expressing system usage.

Figure 4 illustrates the current approach to capture and store audit logs about data processing events in the ASCLEPIOS framework. Note that responsibilities are split between the user interface and the RA. Upon logging in, a unique user ID is obtained from the user's attributes that qualify him/her for the ABAC engine. This ID is retrieved from the RA. Once the authenticated user receives the authorization token from the RA, she/he can start interacting with the system by sending CRUD requests to the police enforcement point, wherein the user requests access to an EMR. The information contained in the request is essential to characterize the data subject of the event, the requested data, the time and location of access, and the purpose of the access. The last part of the interaction that can be logged is the granting or denying of the CRUD request based on the ABAC engine and after based on the ABE key that was supplied.

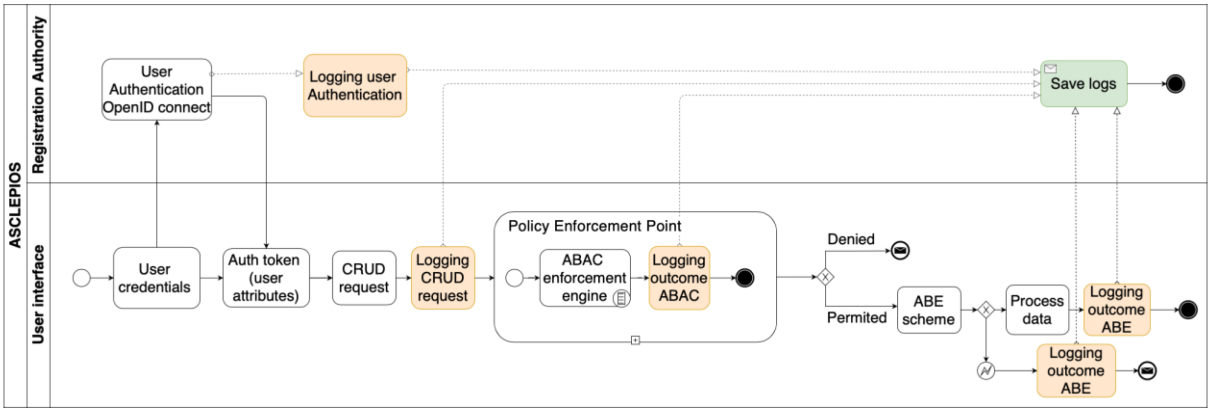


Figure 4: Proposal of interaction among the user interface and the Registration Authority, highlighting the moments for logging information regarding the process event.

Note that the above is preliminary integration suggestions. In Work Package WP5 "Platform Integration and Finalization", the connection and communication between the various

ASCLEPIOS components and the audit logs will be further analyzed and defined for future implementation. This might affect the way the logs are captured and their content. The APAM design and implementation documented below need to be revised when the integration aspects become clearer.

3.3 Audit logs

Audit logs are important to help data managers and DPOs to monitor data activities and detect potential security breaches. DPOs on healthcare organizations are interested in monitoring the audit logs to ensure that healthcare professionals follow all documented protocols and requirements. Audit logs also play an important role in transparency, as they form the resource that tracks information about the events related to the patient's EMR and can be used to present to the patient what is being done with his/her data, when and how.

In this section, we refer to the AuditEvent[70] data model specified by FHIR[71], which is a standard for exchanging healthcare information electronically. Then, we contextualize the audit event attributes presented on the standard, and the envisioned and expected attributes of the audit logs to be used in ASCLEPIOS, in particular by APAM, to support privacy analytics.

3.3.1 FHIR AuditEvent model

Our proposal is based on the FHIR standard for exchanging healthcare information electronically. The standard defines the data model for recording events that can be consumed for auditing, namely AuditEvent[67] This model is intended for use by security and system administrators; security and privacy information managers, which in our case is the DPO, and for transparency toward the patient. All the actors involved in an auditable event should record an AuditEvent, for example, applications that access the medical record for consultation or modification. The FHIR AuditEvent structure offers fields such as event code, the action performed, purpose of use, the outcome that characterizes the event (fail, denial, or success), etc.

Figure 5 shows the Unified Modeling Language (UML) diagram of FHIR AuditEvent[67] and represents the structure of an AuditEvent entry. FHIR declares AuditEvent as a relational database with two main relationships: Agent and Entity. The *Agent* table contains the data processor attributes such as role, policies, name, location, the purpose of use, etc. And it can be extended with the *Network* information used by the agent to the event. The *Entity* table contains the attributes of the data processed, such as data type, data description, a query performed, etc. An Entity can be extended with the class *Detail* for extra attributes.

[70] HL7 FHIR Audit Event https://www.hl7.org/fhir/auditevent.html

[71] HL7 FHIR https://www.hl7.org/fhir/

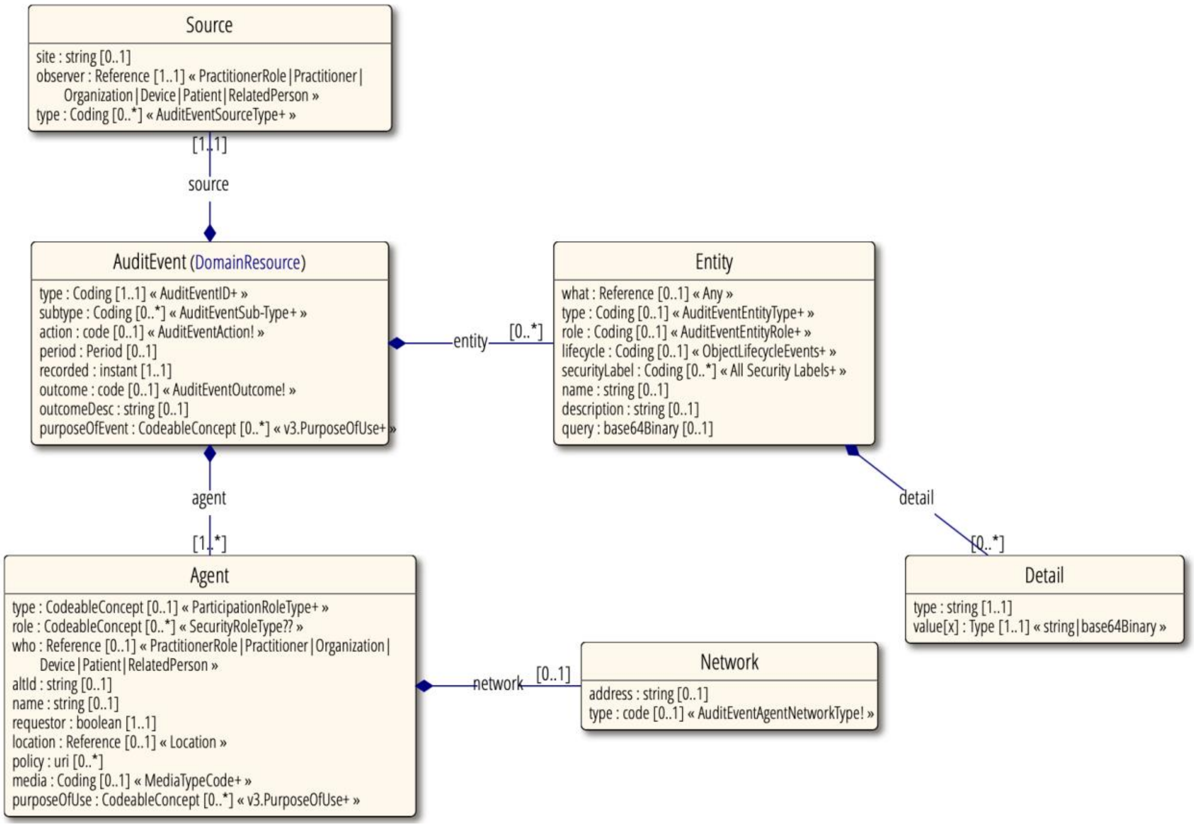

Source
site : string [0..1]
observer : Reference [1..1] « PractitionerRole|Practitioner|Organization|Device|Patient|RelatedPerson »
type : Coding [0..*] « AuditEventSourceType+ »
[1..1]
source
AuditEvent (DomainResource)
type : Coding [1..1] « AuditEventID+ »
subtype : Coding [0..*] « AuditEventSub-Type+ »
action : code [0..1] « AuditEventAction! »
period : Period [0..1]
recorded : instant [1..1]
outcome : code [0..1] « AuditEventOutcome! »
outcomeDesc : string [0..1]
purposeOfEvent : CodeableConcept [0..*] « v3.PurposeOfUse+ »
entity
[0..*]
Entity
what : Reference [0..1] « Any »
type : Coding [0..1] « AuditEventEntityType+ »
role : Coding [0..1] « AuditEventEntityRole+ »
lifecycle : Coding [0..1] « ObjectLifecycleEvents+ »
securityLabel : Coding [0..*] « All Security Labels+ »
name : string [0..1]
description : string [0..1]
query : base64Binary [0..1]
agent
[1..*]
detail
[0..*]
Agent
type : CodeableConcept [0..1] « ParticipationRoleType+ »
role : CodeableConcept [0..*] « SecurityRoleType?? »
who : Reference [0..1] « PractitionerRole|Practitioner|Organization|Device|Patient|RelatedPerson »
altId : string [0..1]
name : string [0..1]
requestor : boolean [1..1]
location : Reference [0..1] « Location »
policy : uri [0..*]
media : Coding [0..1] « MediaTypeCode+ »
purposeOfUse : CodeableConcept [0..*] « v3.PurposeOfUse+ »
Detail
type : string [1..1]
value[x] : Type [1..1] « string|base64Binary »
network
[0..1]
Network
address : string [0..1]
type : code [0..1] « AuditEventAgentNetworkType! »

Figure 5: FHIR's AuditEvent UML diagram

3.3.2 Envisioned audit logs for ASCLEPIOS

Here we describe the essential attributes that the audit logs must contain to achieve APAM's objectives. FHIR's AuditEvent[67] specification defines most of them, and additional attributes can be extended in the model as a relationship. These two fundamental additional attributes are the healthcare professional *Organization* and *Data Subject* identification. These attributes will be used to implement access control to APAM services: the DPO must be able to process only the audit logs in which the *Agent* belongs to the DPO's *Organization*. The same applies to the patient, who must be able to process only the audit logs in which she/he is the data subject.

Table **10** presents the essential attributes. Note that this is a proposal that will be considered in WP5 during integration activities for ASCLEPIOS framework, and revised accordingly.

Attribute name	Description
Event type	Provides a list of codable events. e.g. import/export, query, patient record, procedure record etc.
Action	Describes the actions which can be CRUD operations or an Execute action which represents the execution of some application's method or function.
Time recorded	Data-time or timestamp of when the event was saved in the audit log. FHIR also has an attribute named Period that represents the interval of time during which the action was performed.
Period	Start and end date-time or timestamp of the event. Can also be represented by a number.
Outcome	Describes the answer given to the event, e.g. permitted, denied, or failed.
Outcome description	FHIR enables describing the outcome result, and this should be used to explain if the outcome is related to the ABAC or the ABE.
Purpose of use	FHIR provides a list of possible codable purposes of events, e.g. Emergency treatment, Break the glass.
Agent	FHIR uses the term agent to represent the actor that performed the event and action, as well as the actor's attributes such as name, role, department, location, network address, etc. In our proposal, the agent is the data processor, who can be the healthcare professional or the patient
Source	FHIR defines the source as the system reporting in an event, which in our case is the healthcare application.
Entity	Object accessed or used and details about the data object, e.g., DICOM Image, Prescription, Medical Condition.
Data subject	FHIR does not specify the data subject of an event; however, we added the identifier of the data subject., which is the patient. The data subject can also be the agent of an event.

Organization	We define healthcare organizations as a Type of the Agent and represent the Organization in a separate field. This enables access control for the DPOs to access the respective audit logs.

Table 10: Envisioned audit logs attribute content.

3.3.3 Envisioned privacy and analytics audit questions

Below we show some examples of audit questions in **Error! Reference source not found.** that can be answered using the essential attributes from the audit logs listed in

Table **10**. These questions illustrate the type of questions about privacy analytics from the perspective of stakeholders, indicating functions to be supported by APAM. An exhaustive list of audit questions and metrics is out of scope here. Instead, we provide concrete insights into the information that needs to be extracted from the audit logs and the analytics tools necessary to process them. Note that many of the audit questions are in principle applicable by patients, regarding their own EMR, or a DPO, regarding the EMR of all or a group of patients in the Organization. The difference between the two is only the selected audit log entries where to apply the functions. Also note that, in practice, more complex filtering operations are expected, resulting in combinations of the presented questions. These questions served to inspire the APAM prototype implementation presented in Section 3.4.

Table 11 demonstrates an example of privacy analytics audit questions and how the audit log attributes can be used to answer them. The envisioned analytics tools are represented in boldface.

Stake-holder	Audit question	Metrics and process to answer
Patient or DPO	How many and which organizations that requested access to the patient's EMR had access denied?	Apply a **filter** to select Events with Outcome that was Denied or Failed. **Group** by Organization and **count**. Display name, counts and distribution of denied/failed events per organization (**chart**).
Patient and DPO	What is the most occurring purpose for the data access requests?	**Group** the Events by the Purpose of use, and **count**. Display the Purpose of use with the highest count.

Patient and DPO	Which and how many **actions** were done with the patient's EMR?	**Filter** and **group** the Events by each CRUD Action and **count.** Display distribution of CRUD actions (**chart**)
Patient and DPO	Which professionals accessed the patient's EMR in this organization? Which **roles** do they have in the organization and to which **departments** they belong? What is the distribution?	Filter Events for given Organization. Group by role and department of the Agents. **Count** and get the **percentage** for each role and department. Display distribution and counts (**chart**).
Patient and DPO	How long has lasted the access to the patient record? Provide min, max, and mean **duration times**.	Retrieve Period for all Events and calculate **Minimum**, **Average** and **Maximum** duration**.** Display results.
DPO	How many access requests were performed from outside the organization? Provide **region/city/country/continent**	F**ilter** Events where the Source location's information diverges from the Organization's location. **Count** and calculate the **percentage**, display distributions.
DPO	Which professionals had the most access requests **outcome** as denied? Show the top 10.	**Filter** Events by Outcome being failed or denied. **Group** by Agent identification (healthcare professional) and **count**. Display Agent identification for professionals with the ten largest counts.
DPO	Which data access **events** deviate from normal and should be verified? Show top 100.	Apply machine learning algorithms over the entire database and identify possible **outliers**. **Sort** them by the degree of deviation (e.g. suspicion level). Display information about the top 100 Events.

Table 11: Example of privacy analytics audit questions.

3.3.4 Audit logs access control

Audit logs are related to sensitive EMR data sources, and they can disclose private information about the data subjects or the data processors (in this case, the patients and the healthcare professionals that accessed the information). For example, the knowledge that the patient's EMR was accessed from the oncology department might reveal that the patient is suspected of having cancer. Also, logs can contain information that can be misused to derive security policies, for example, through granted and denied requests. Even information that is disclosed in aggregated form might be revealing, for example, that no employees of a given department accessed a patient's EMR, which could indicate a treatment mistake. Therefore, strict access control needs to be implemented to protect sensitive information both for patients and for healthcare professionals involved in the audit logs events.

Audit logs serve a variety of tasks in an organization, from legal requirements to quality control and more technical security monitoring. It is possible that part of the log content can only be disclosed in the context of a certain task. Here we focus on the perspective of privacy analytics, which involves the perspectives of the DPO and the patient.

At the organization level, the DPO can access and process every audit log event where the agent is one of the organization employees. At the patient level, the patient can access and process every audit log where she/he was the data subject. However, to protect the identity of the healthcare professional, the agent name must be anonymized using other identifications, such as the role, department, and organization. For example, the patient knows that a doctor from the oncology department of the Amsterdam UMC accessed her EMR, but she does not know the name of the doctor.

Another important aspect is that nobody should be allowed to tamper or delete any event on the audit log. Moreover, every data processing of the audit logs must be traceable to guarantee the integrity and allow monitoring the access control policies to the audit logs in a system level. Those requirements are listed in the H-AUD3, H-AUD4 and H-AUD5 of the auditory requirements highlighted in the Subsection 5.1.2. We envision that ABAC should be used for the audit logs in ASCLEPIOS, but this will be defined in WP5.

3.4 ASCLEPIOS Privacy and Analytics Module

APAM is a toolkit for privacy analytics, acting as a TET and an analytics tool. The objective is to provide information about the processing events on the patients' EMR for two main stakeholders: the patient and the organization's DPO.

For the patient, APAM offers awareness and transparency about the policies implemented to safeguard the EMRs and metrics about all data processing activities on their EMRs. Moreover, APAM helps the organization's DPO to understand the way in which the healthcare professionals (data processors), process the patients' EMR, and to monitor if they follow the required security procedures. Through this monitoring, DPOs can detect misconduct and adapt access control policies.

APAM offers tools to process the audit logs through user interfaces. It gives means for the patient and DPO to request metrics about the data processing on EMRs, and to analyze and visualize those metrics. Towards these goals, APAM has the following capabilities:

- It offers calculation of metrics with predetermined functions,

- It enables specification of parameters for filtering the logs of interest, for example, patient, organization, or period,
- It implements outlier detection methods that can highlight cases that should be investigated more closely, and in this manner guide the DPO to inspect possible deviations from adequate professional behaviour,
- It provides visualization through charts commonly used to represent the metrics.

In this section, we introduce the conceptual architecture of APAM and the constraints for the current implementation, which is not yet integrated with the ASCLEPIOS framework. Then, we present how we implemented the main components and the services of the current APAM prototype.

3.4.1 Architecture

The conceptual architecture of APAM assumes that the Registration Authority, which is the entity responsible for user authentication in the ASCLEPIOS framework, will store the audit logs. Moreover, the architecture was designed to enable applications to invoke APAM services and choose to present the responses of these services as part of the application interface or to use the APAM visualization interface through the web browser directly.

Figure 6 presents the architecture of APAM in the ASCLEPIOS framework, also showing the main components it is connected to, namely the Registration Authority and the application User interface. The components running inside APAM rely on a shared data source, i.e. the audit logs held by the Registration Authority. APAM offers two ways to integrate with any application: the REST API and a web-based visualization interface. The REST API allows the healthcare application to interact with APAM by sending requests and using the responses for computing customized metrics and charts for visualization. Moreover, the patient and DPO can also directly use APAM's visualization interface, which is a web interface running inside the APAM server that provides predefined charts.

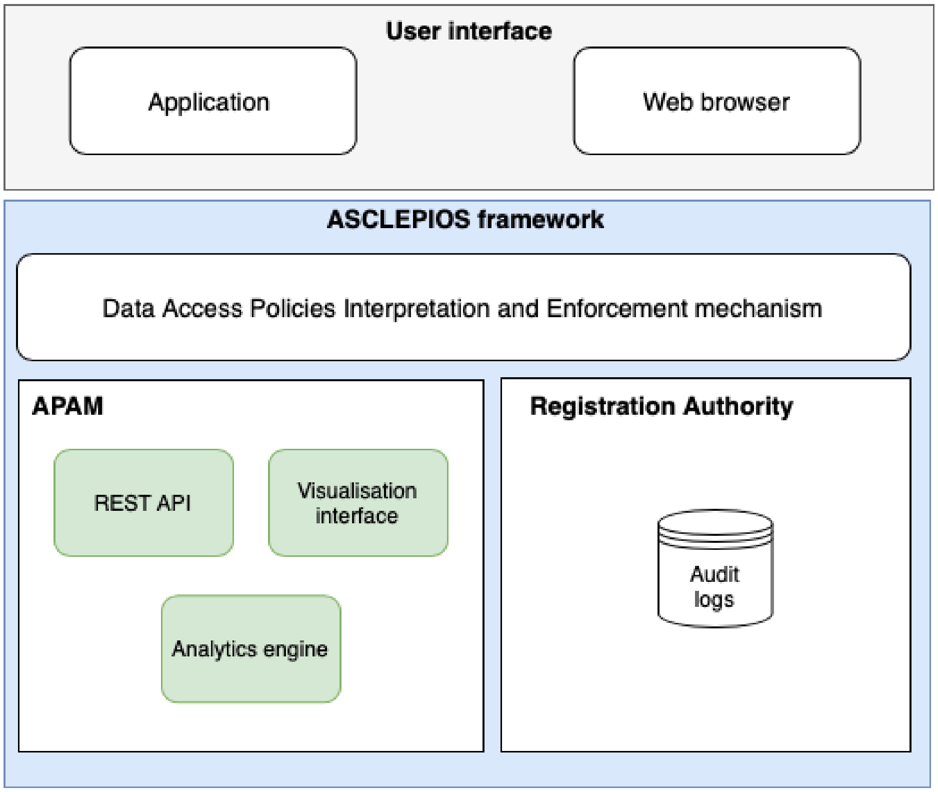


Figure 6: Architecture diagram showing APAM (components in green), the Registration Authority (responsible for the audit logs), Data Access Policies Interpretation and Enforcement mechanism (responsible for access control), and the User Interface (application and web browser).

3.4.2 Constraints

Task 2.6 raised the requirements for auditability and the necessity of a single audit log data source in the ASCLEPIOS framework. Until that point, however, such logs were not considered as an integral part of the ASCLEPIOS framework. The development of APAM is directly related to the development of the audit log data source and the access control models that will be offered by other components of the ASCLEPIOS framework. Therefore, APAM's development has been constrained by the early stage of considerations about logs in the project, and the start of integration efforts being carried out in WP5. This concerns the absence of a Registration Authority component, and therefore no audit logs database, as well as no defined policies for access control to the supposed logs.

For demonstration purposes, we therefore implemented the components that should be provided by the ASCLEPIOS framework in the future. We created a database containing synthetic audit logs, however disregarding any access control mechanism. Such demonstration enables illustrating privacy analytics tools and investigating alternatives for APAM implementation and its services.

3.4.3 Implementation

Figure 7 shows the software and services used to implement the prototype following the architecture presented in Figure 6. The audit logs database is placed inside APAM, and the visualization interface was expanded and implemented using generic packages Logstash[72], Elasticsearch[73] and Kibana[74]. Below we present an overview of the services and libraries used to implement the APAM prototype. The full documentation and open source code of APAM will be available in the Gitlab repository[75] of the ASCLEPIOS project.

[72] Elastic Logstash https://www.elastic.co/logstash
[73] ElasticSearch https://www.elastic.co/elasticsearch/
[74] Elastic Kibana https://www.elastic.co/kibana
[75] ASCLEPIOS APAM REST API https://gitlab.com/asclepios-project/apam

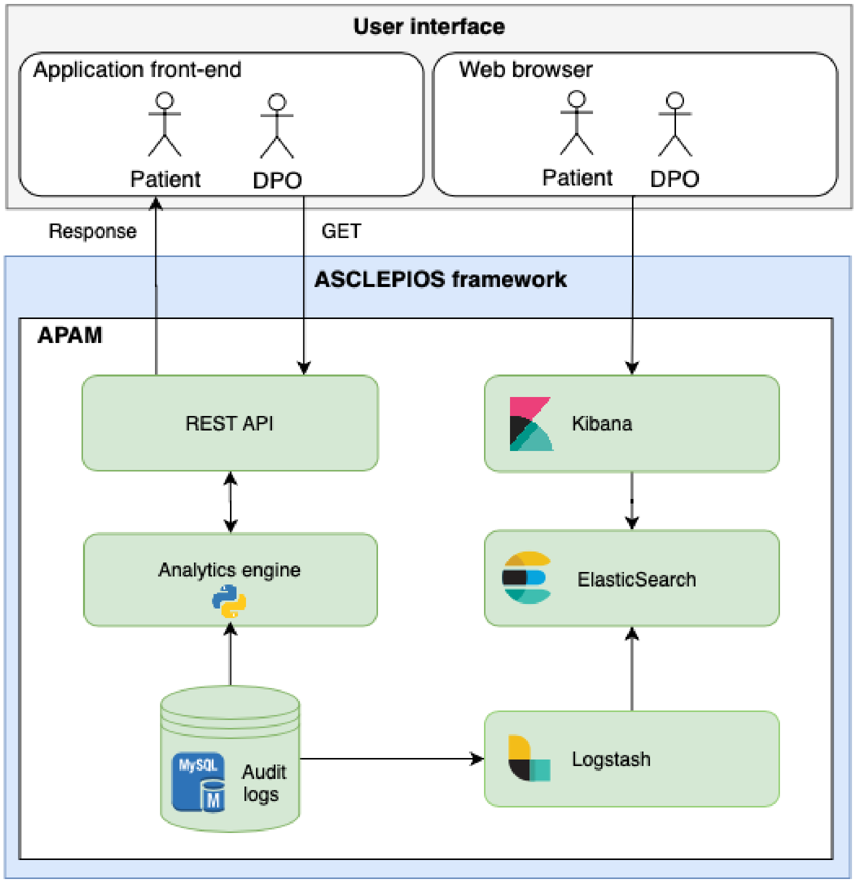


Figure 7: APAM's prototype implementation diagram with services available and software used.

The audit logs database was implemented using MySQL, and the analytical engine was implemented using the programming language Python. Moreover, we use Pandas[76] and Scikit-learn[77] libraries to perform regular statistics and analytics functions using machine learning algorithms. We created the web interface using Kibana and implemented a REST API using the Django Rest Framework[39]. Table 12 summarizes the services and their respective role in the prototype implementation.

Services	Role
MySQL[78]	Database management system (DBMS) used to deploy the

[76] Pandas https://pandas.pydata.org/pandas-docs/stable/reference/api/pandas.DataFrame.html
[77] Scikit-learn https://scikit-learn.org/stable/
[78] MySQL WorkBench https://www.mysql.com/products/workbench/

	audit logs.
Logstash[79]	Feeds ElasticSearch with data from the database. In this case, from MySQL.
ElasticSearch[80]	Indexes the data that is ingested into it and makes them available to Kibana
Kibana[81]	Offers visualization for the data indexed by ElasticSearch.
REST API	Handles HTTP requests and returns the output data to applications.
Analytics Engine	Performs analytics functions on the data.

Table 12: Services that compose APAM.

There are two data flows in APAM from the audit logs until the end-user: by the REST API and by the web interface. First, the audit logs can be retrieved by the application through the REST API calls. An application performs a request to the REST API, which retrieves the audit logs from the database and processes them with the Analytics Engine functions. After processing the request, the REST API sends the response to the application in JSON format. Second, the logs can be consulted through an interactive web interface, using functions that retrieve the logs from the database and ingest them into ElasticSearch for data indexing. Then, Kibana consumes the logs from ElasticSearch to apply filtering, run analytics and to create a dashboard with various types of visualizations.

Below we describe each component of the APAM implementation in detail.

3.4.4 Synthetic audit logs

APAM's current audit log model was built based on the FHIR AuditEvent specification. Figure 8 shows the relational model using the attributes derived from the envisioned audit logs described in Section 3.3.2. The model was created with MySQL WorkBench and then deployed with a MySQL relational database.

[79] Elastic Logstash https://www.elastic.co/logstash
[80] ElasticSearch https://www.elastic.co/elasticsearch/
[81] Elastic Kibana https://www.elastic.co/kibana

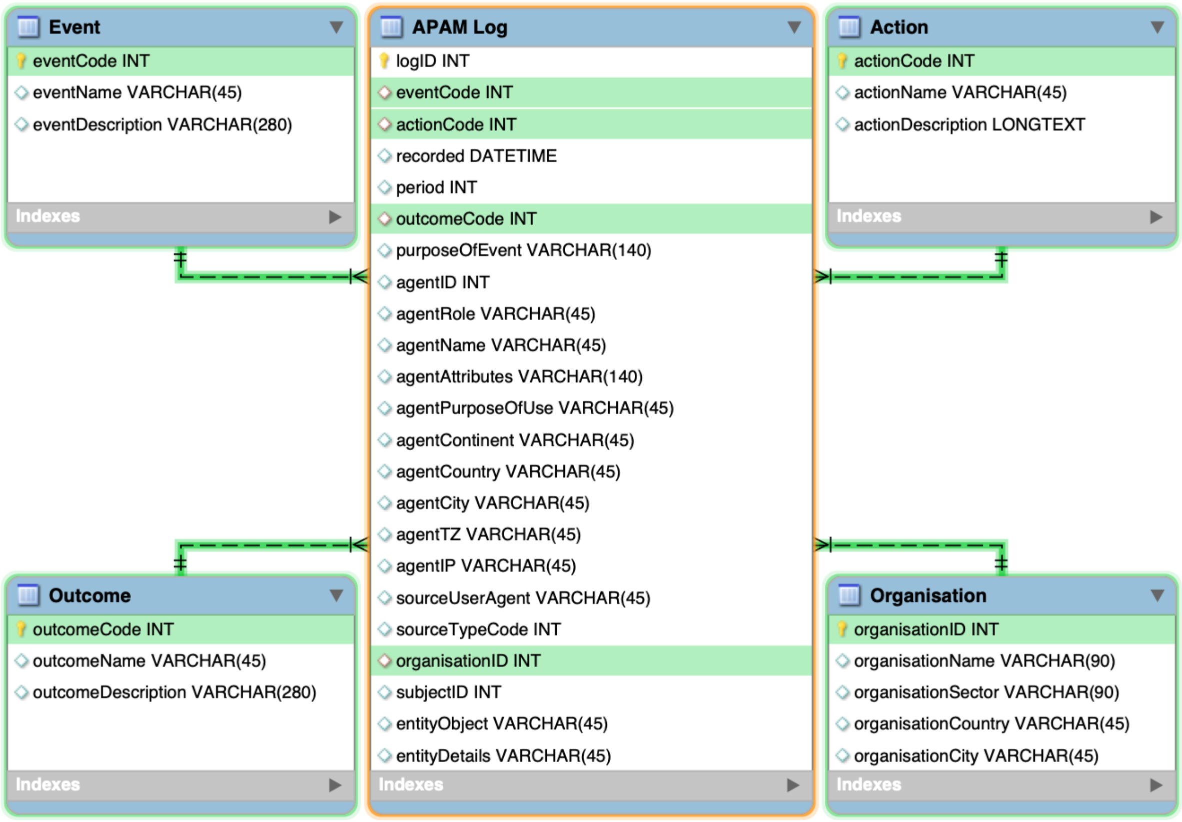


Figure 8: Relational audit log model for APAM.

The model in Figure 8 was used to populate the synthetic logs database, including patients, EMR data, healthcare professionals, and activity data corresponding to EMR access CRUD operations. We generated fake names, roles, addresses, attributes etc. The organizations were populated with real healthcare organization’s information. To simulate access from multiple regions of the world, we used a geolocation generator, which provides attributes such as country codes, coordinates, IP addresses, time zone, etc. Finally, we simulated random CRUD operations and their outcome on the patient's EMR in a range of time. The audit logs simulator was implemented using the Python package Faker[82] and IP2Geotools[83] libraries.

3.4.5 Analytics engine

The APAM analytics engine offers functions to perform analytics on the audit logs, such as basic statistics and anomaly detection techniques. APAM returns to the application the results of the

[82] Faker, fake data generator https://github.com/joke2k/faker
[83] IP2Geotools, API for geolocations https://pypi.org/project/ip2geotools/

analytics functions and the list of events where they were applied through the REST API. Here we present the functions that the analytics engine offers, and the interaction through the REST API will be present in the section 5.4.6.

Basic statistics

The basic statistics functions were implemented using the Pandas library[84]. APAM offers functions to count the number of occurrences of a given value of an attribute, or of events with combined attributes, in a set of audit logs that can filter by time. After counting, APAM calculates the percentage of the values occurrences as well as the maximum, minimum, mean, median values and standard deviation. For example, it is possible to count the total number of "create" actions performed per day in the last week, then calculate the mean, median and standard deviation of the number of "create" actions in the week, and then identify the days that had the maximum and minimum number of these actions. APAM also offers the basis statistics for other numerical attributes, such as the duration of an event.

Anomaly detection

Deliverable D2.3 “GDPR-compliant and Privacy-Preserving Analytics for Healthcare Providers” describes the taxonomy of the most used algorithms for anomaly detection. It defines unsupervised methods as capable of detecting outliers in an unlabeled dataset, which is the case of APAM. The assumption is that most of the instances have a common behaviour considered normal, so the instances that fit least to this common behaviour can be interpreted as most anomalous. This can be done using by a clustering approach[85], where similar instances are grouped into clusters based on similarity measures. Instances that do not belong to the group, or that are the farthest to the center of the group, are considered outliers, and interpreted as anomalies. Some unsupervised ML-algorithms for clustering approach are k-means, k-medoids and EM clustering[86].

The prototype implementation of APAM provides the k-means clustering algorithm for outlier detection using the Scikit-learn library[87]. The k-means clustering algorithm creates groups of similar events into clusters that are characterized by their centroids. No predefined classification is required for the points. The assumption is that each cluster defines some type of normal behaviour, and the distance from a point to the cluster's centroid can be considered as a deviation from “normal”. The events that are most distant from the respective centroid is considered an anomaly or outlier[88]. In APAM, a list of the top anomalous events is returned by the outlier detection function.

[84] Pandas https://pandas.pydata.org

[85] Garcia-Teodoro, Pedro, et al. "Anomaly-based network intrusion detection: Techniques, systems and challenges." *computers & security* 28.1-2 (2009): 18-28.

[86] Agrawal, Shikha, and Jitendra Agrawal. "Survey on anomaly detection using data mining techniques. "*Procedia Computer Science* 60 (2015): 708-713.

[87] Scikit-learn k-means https://scikit-learn.org/stable/modules/generated/sklearn.cluster.KMeans.html

[88] MacQueen, James. "Some methods for classification and analysis of multivariate observations." *Proceedings of the fifth Berkeley symposium on mathematical statistics and probability*. Vol. 1. No. 14. 1967.

In APAM the implementation of this outlier detection mechanism with k-means also relies on the use of Principal Component Analysis (PCA)[89], which is a dimensionality-reduction method for setting the most relevant attributes of the large set. By applying this principle, we cluster the audit logs in k groups, and, after re-dimensioning and clustering, we take the farthest points from each centroid. The DPO can define parameters such as k (the number of clusters), number of principal components and the time interval of the logs to be considered. However, APAM offers support that automatically chooses the optimal values for k and PCA parameters, to enable the APAM to detect anomalies with as less human intervention as possible.

The adaptive support to define k is based on highest silhouette coefficient (SC)[90]. APAM uses the silhouette score function[91] to calculate SC for a predefined range of possible k's. The silhouette score function measures how close each cluster point is to points in neighbouring clusters. The number of clusters with the highest value from the silhouette coefficient analysis is automatically chosen as k.

The PCA parameter is determined based on the explained variance that we want in our re-dimensioned attributes features, which is set as a parameter in Scikit-learn PCA function[92]. The explained variance is the percentage of information that we want to retain while decreasing the dimension. The attributes with low or any variance are considered to have flat data and do not impact in the representation of the information. In APAM, the default explained variance is 95% and can be changed. Therefore, when the DPO uses the adaptive support functions the only parameter that she/he needs to input is the top percentage to consider anomaly.

REST API

REST API is a web application often built over HTTP protocol that allows multiple applications to perform HTTP actions by making requests to its endpoints. APAM REST API provides endpoints to which the applications can send requests. Each endpoint performs some function of the Analytics Engine on the audit logs and sends the response containing metrics and the list of related attributes back to the application that issued the request. For APAM endpoints, the parameters of the request specify the target agents (all healthcare professionals from the organization or a specific one), the data subject (all patients or a specific one), and also diverse filtering options to compose more complex queries on the audit logs. For example, it is possible to choose to filter by date range, organization, and agent role. The response provided by the REST API can be further processed by the application to implement custom analysis and visualizations in the application's user interface.

[89] Ding, Chris, and Xiaofeng He. "K-means clustering via principal component analysis." *Proceedings of the twenty-first international conference on Machine learning*. 2004.

[90] Rousseeuw, Peter, Silhouettes: A graphical aid to the interpretation and validation of cluster analysis, Journal of Computational and Applied Mathematics, 20(20), 53–65, 1987

[91] Scikit-learn Silhouette score https://scikit-learn.org/stable/modules/generated/sklearn.metrics.silhouette_score.html

[92] Scikit-learn PCA function https://scikit-learn.org/stable/modules/generated/sklearn.decomposition.PCA.html

Examples of the functions provided by APAM's REST API are described in Table 13. All the functions retrieve, filter and process the audit logs using the Analytics Engine. The URL path begins with *APAM_IP/APAM/**. The call can specify as parameters all the attributes of the audit logs listed in **Table 10**. The full content of available endpoints, functions and parameters can be found in the APAM documentation in the Gitlab repository[93]**.**

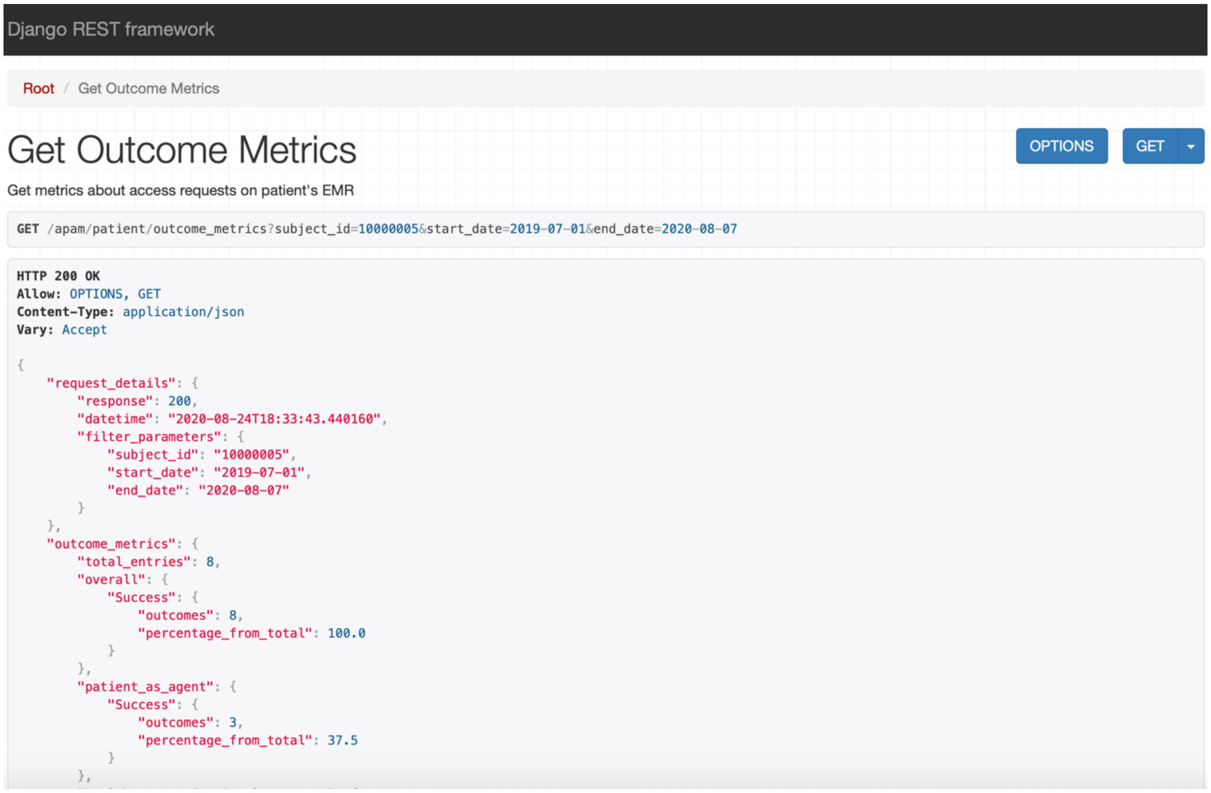
URL	**Parameters**	**Output description**
/patient/**full_log**	start_date, end_date, start_hour, end_hour, subject_id.	Returns all the data access Events on the given subject/patient for the given date range.
/patient/**outcome_metrics**	start_date, end_date, subject_id, organisation_name, source_city, source_country, agent_role, agent_action etc.	Returns metrics about the outcome of the data access events, how many denied, permitted, and failed, grouping by organization and agent roles.
/patient/**access_period_metrics**	start_date, end_date, subject_id, organisation_name, source_city, source_country, agent_role, agent_action. etc.	Returns metrics about the duration of access to the patient's EMR grouping by organizations, agent roles and actions.
/patient/**action_metrics**	start_date, end_date, subject_id, organisation_name, source_city, source_country, agent_role, agent_action etc.	Returns metrics about the actions performed on the patient's EMR, grouped by organization and agent roles.

[93] ASCLEPIOS APAM REST API https://gitlab.com/asclepios-project/apam

/organisation/**full_log**	start_date, end_date, start_hour, end_hour, organisation_id.	Returns all the organization audit logs for the given date range.
/organisation/**outcome_metrics**	start_date, end_date, organisation_id, source_country, source_city, agent_id, agent_role, agent_action, subject_id etc.	Returns metrics related to the outcome of the request. How many denied, permitted, and failed, grouped by agent roles.
/organisation/**location_metrics**	start_date, end_date, organisation_id, source_country, source_city, agent_id, agent_role, agent_action, subject_id etc.	Returns metrics related to the location (country and city) of where the request access was made. Grouping by each access location, the actions and outcome.
/organisation/**action_metrics**	start_date, end_date, start_hour, end_hour, organisation_id, source_country, source_city, agent_id, agent_role, agent_action, subject_id etc.	Returns metrics related to the actions performed. How many of each action, grouped by agent_roles.
/organisation/**time_period_access**	start_date, end_date, organisation_id, source_country, source_city, agent_id, agent_role, agent_action, subject_id etc.	Returns metrics related to the time of the access, grouped by agent_roles.
/organisation/**outlier_detection**	start_date, end_date, start_hour, end_hour, farthest_percent, k_number, pca_number	Returns a list of the events classified as outliers.

Table 13: APAM REST API: available endpoints and functions performed.

Figure 9**: Example of a request to the APAM_IP/apam/patient/outcome_metrics endpoint.** shows an example of a request to the REST API *APAM_IP/APAM/patient/outcome_metrics* endpoint, with the introduction of filter parameters during the request to specify the time range to be considered.



Get Outcome Metrics

Get metrics about access requests on patient's EMR

```
GET /apam/patient/outcome_metrics?subject_id=10000005&start_date=2019-07-01&end_date=2020-08-07
```

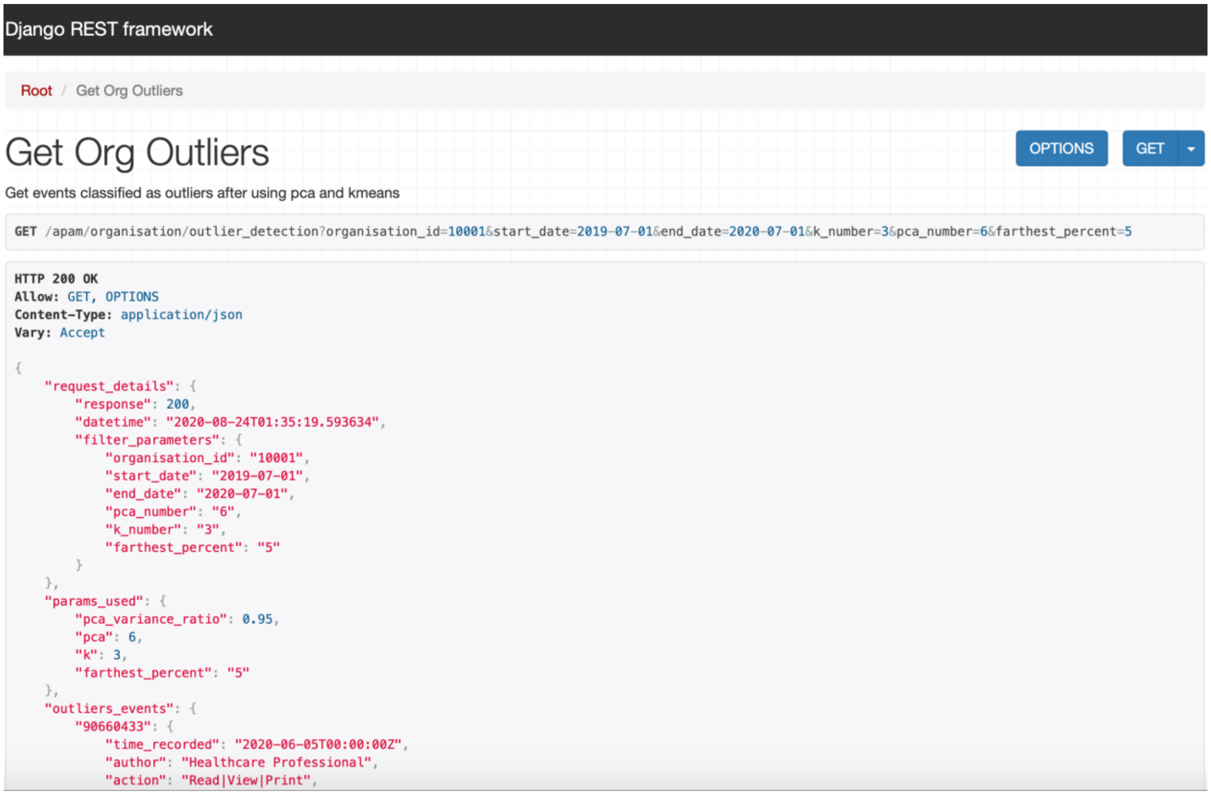
```
HTTP 200 OK
Allow: OPTIONS, GET
Content-Type: application/json
Vary: Accept

{
    "request_details": {
        "response": 200,
        "datetime": "2020-08-24T18:33:43.440160",
        "filter_parameters": {
            "subject_id": "10000005",
            "start_date": "2019-07-01",
            "end_date": "2020-08-07"
        }
    },
    "outcome_metrics": {
        "total_entries": 8,
        "overall": {
            "Success": {
                "outcomes": 8,
                "percentage_from_total": 100.0
            }
        },
        "patient_as_agent": {
            "Success": {
                "outcomes": 3,
                "percentage_from_total": 37.5
            }
        },
```

Figure 9: Example of a request to the APAM_IP/apam/patient/outcome_metrics endpoint.

The response of the REST API call shown in Figure 9 contains details about the request done and the metrics calculated. In this example the response HTTP code was 200, the date when the request was performed was on 24th August of 2020, and the parameters used for filtering were the subject_id, start_date and end_date. The metrics are returned separated and grouped by organizations, agent roles, actions etc.

Figure 10: Example of request for outlier detection using the APAM_IP/apam/organisation/outlier_detection endpoint.

Figure 10 shows an example of the request for outlier detection. In this example, we have passed the number of clusters (k_number), the number of principal components analysis (pca_number) and the percentage of the farthest events from each cluster centroid (top_percent). The output contains the details about the request and the list of events considered outliers.



Get Org Outliers

Get events classified as outliers after using pca and kmeans

```
GET /apam/organisation/outlier_detection?organisation_id=10001&start_date=2019-07-01&end_date=2020-07-01&k_number=3&pca_number=6&farthest_percent=5
```

```
HTTP 200 OK
Allow: GET, OPTIONS
Content-Type: application/json
Vary: Accept

{
    "request_details": {
        "response": 200,
        "datetime": "2020-08-24T01:35:19.593634",
        "filter_parameters": {
            "organisation_id": "10001",
            "start_date": "2019-07-01",
            "end_date": "2020-07-01",
            "pca_number": "6",
            "k_number": "3",
            "farthest_percent": "5"
        }
    },
    "params_used": {
        "pca_variance_ratio": 0.95,
        "pca": 6,
        "k": 3,
        "farthest_percent": "5"
    },
    "outliers_events": {
        "90660433": {
            "time_recorded": "2020-06-05T00:00:00Z",
            "author": "Healthcare Professional",
            "action": "Read|View|Print",
```

Figure 10: Example of request for outlier detection using the APAM_IP/apam/organisation/outlier_detection endpoint.

3.4.6 Visualization interface

APAM's visualization web interface provides charts for patients and DPOs that show the results of basic statistics on the audit logs. The charts are created following the same metrics provided by the REST API. The visualization interface allows the users to interact with APAM using the web browser directly. It does not require any integration of APAM with the application.

For the patient, the web interface shows charts about events of data processing on her/his medical record. This includes 5W1H information, indicating the healthcare professional roles, department, organizations, and origin of access, as well as the action performed on the patient's data, on which data and the outcome (failed or denial attempts).

For a DPO, the web interface shows charts containing information about access to data done by all professionals from the organization wherein the DPO is responsible. For example, the DPO can see and monitor the locations where the data was accessed or check the most performed CRUD functions. As another example, in a detective approach, the DPO can visualize and track the audit logs from a specific professional or visualize the events that diverge from the normal behaviour of the professional's role.

Patient and DPO can also specify filters that directly affect the content of the metrics and charts, e.g. the date interval of the audit logs. For demonstration purposes, APAM's visualization interface was created using Kibana. The created charts illustrated below show how the output from the REST API is displayed to the end-user.

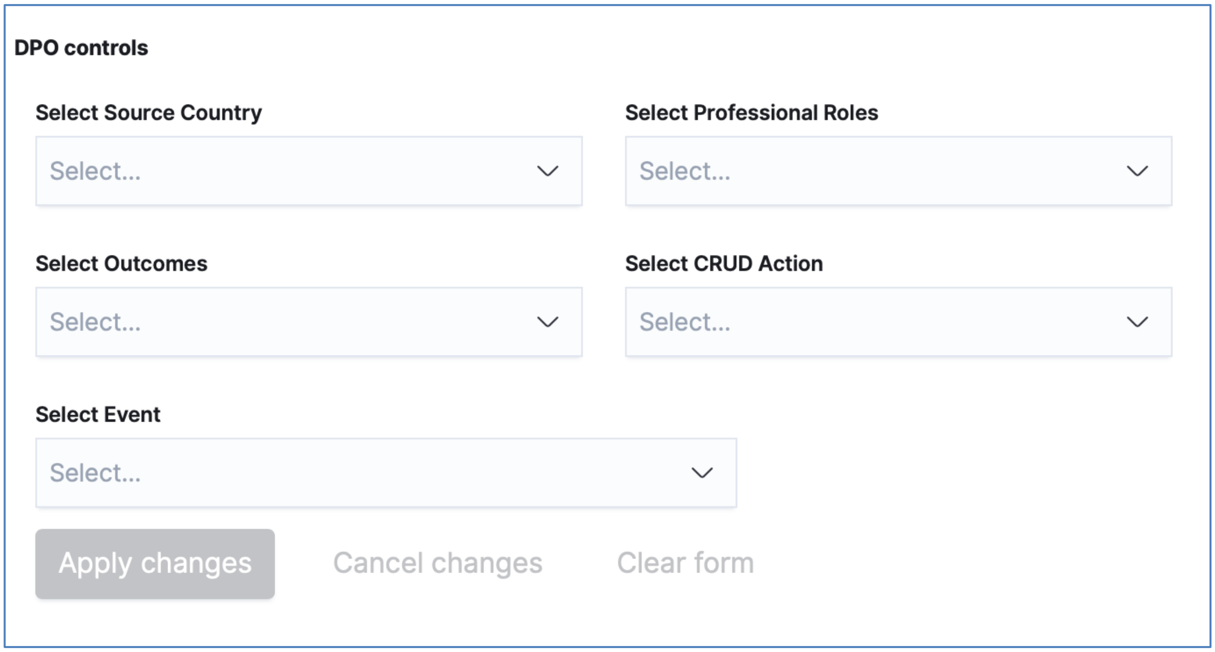


Figure 11: APAM visualization: dashboard filters for DPO.

Figure 11**: APAM visualization: dashboard filters for DPO.** shows an example of filters that can be used by the DPO, and **Figure 12** shows an example of a chart presenting the actions performed by the healthcare professionals of a given organization. In this example, the DPO can see that no records have been deleted in this organization.

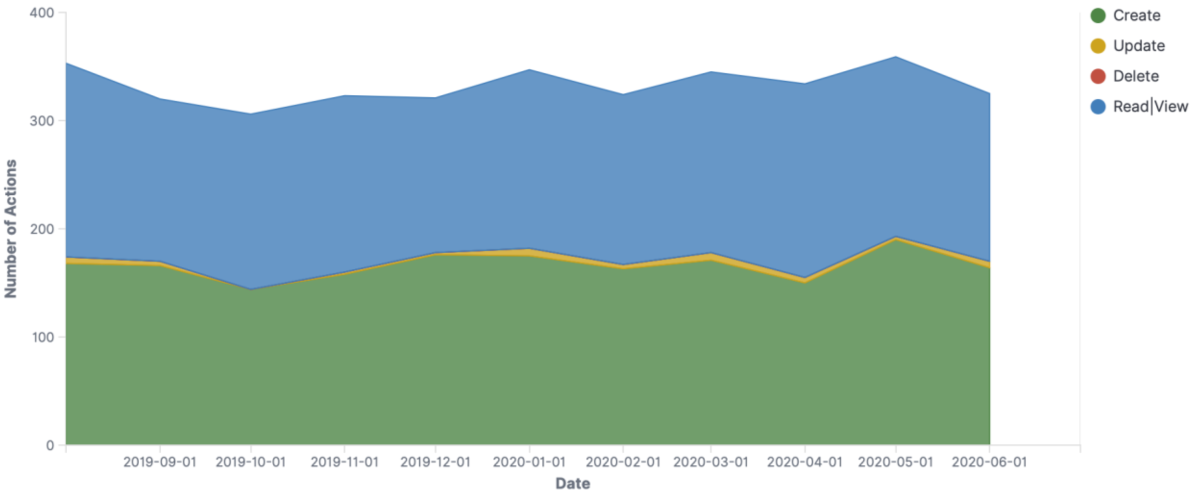


Figure 12: Line chart showing the number of actions performed by date.

Figure 13**: Pie chart with statistics grouped by department and role** shows a pie chart presenting the percentage of each department and the percentage of roles in the department from all the requests to access the patient's EMR. This type of chart can be applied to various attributes, such as the metrics retrieved from grouping by actions and outcomes or grouping by regions of access and actions.

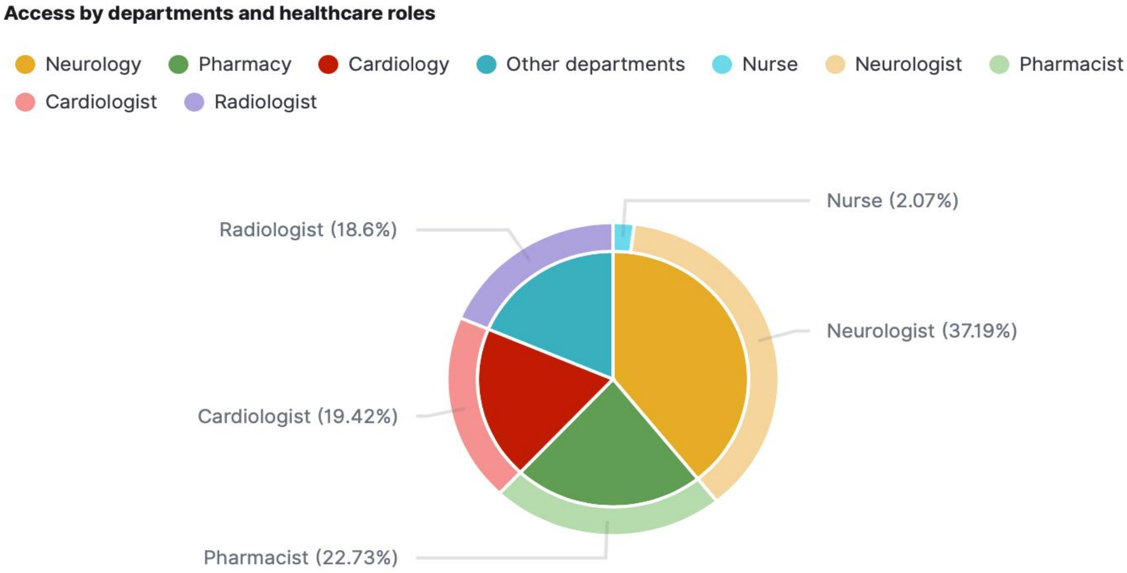
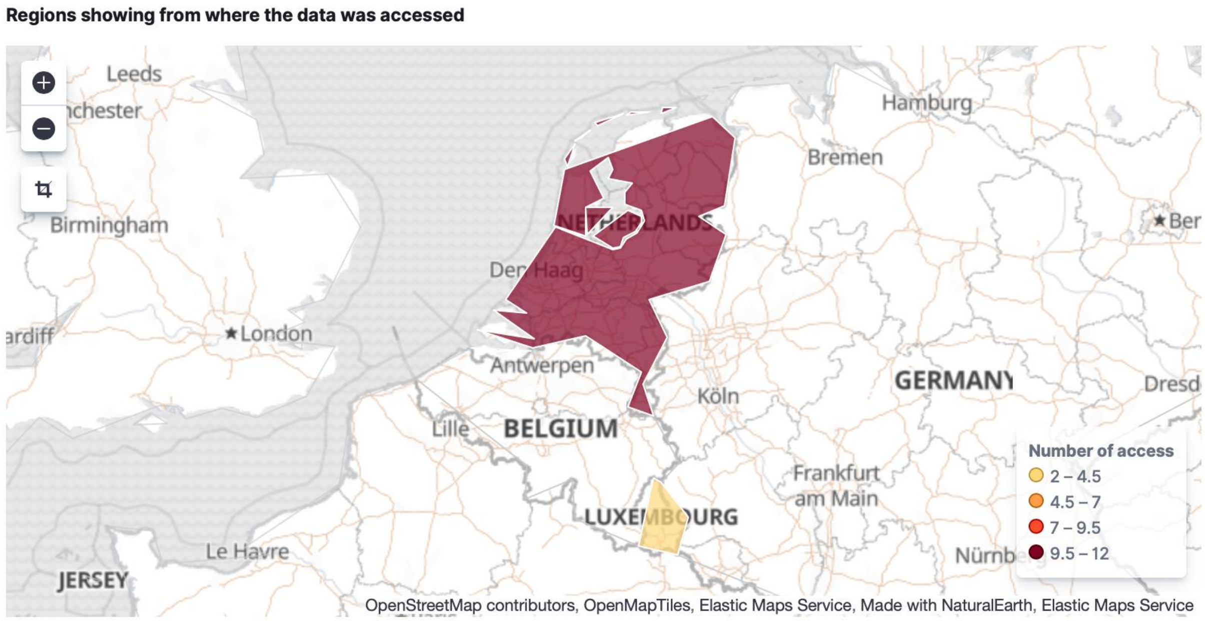


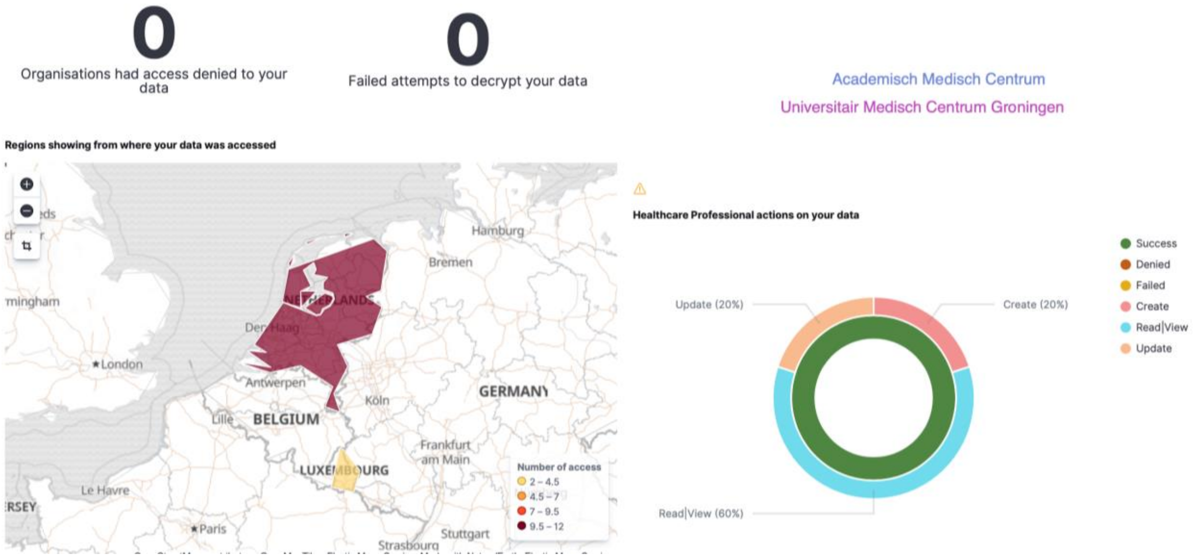
Figure 13: Pie chart with statistics grouped by department and role.



Regions showing from where the data was accessed

Figure 14: Map showing from which countries the patient's EMR was accessed.

Figure 14**: Map showing from which countries the patient's EMR was accessed** shows an example of a region map showing the countries from where some patient's data was accessed.



0
Organisations had access denied to your data

0
Failed attempts to decrypt your data

Academisch Medisch Centrum
Universitair Medisch Centrum Groningen

Regions showing from where your data was accessed

Healthcare Professional actions on your data

Figure 15: An example of the patient's dashboard.

Figure 15 shows the dashboard containing charts and explicit metrics that allows the patient to know what is going on with his/her data. The figure shows the number of denied requests or failed attempts to decrypt the data, what organizations have accessed the data, and from where. It also shows the actions performed and their outcome.

4 Conclusions

The deliverable reported the design of a tool, called Emnet, that enables privacy-preserving statistical computation on data distributed across multiple data sources. Emnet computes based on decomposing statistical algorithms into sub-computations of summation forms that are executed using secure summation protocols. To that end, the tool currently implemented two secure summation protocols such as the SINE and the k-secure summation protocols. It is known that secure summation protocols are efficient and scalable. In addition, the record level computations are locally executed at the data sources. Therefore, Emnet is efficient and scalable.

The deliverable described how different statistics, summation, count, ratio, mean, min, max, percentile, variance, covariance, and Pearson's r statistics can be computed on data distributed across three or more data sources. Sample Emnet queries and results are available in Annex. II: Sample Emnet queries. We planned to implement more statistics and perform experimental evaluations. We also planned to implement the rest of the statistics. Studies have shown that many more statistics can be decomposed into sub-computations of summation forms, and, therefore, can be implemented in Emnet.

Emnet currently computes on plain health data distributed across healthcare institutions. There is a plan to enable computation on health data encrypted using SSE scheme. To that end, we will replace Emnet data analytics service with Functional Encryption (FE) service, developed as part of Task T2.3 "GDPR-compliant and Functional Encryption-enabled Prescriptive Analytics for Healthcare Providers"[94].

The deliverable presented a background study about privacy analytics for the data owner, raising responsibilities and requirements for privacy and transparency on the healthcare domain where the data subject (patient) represents the data owner, and the DPO represents the interests of the data subject in each organization. In ASCLEPIOS, both patient and DPO are considered stakeholders in privacy analytics.

During the background study regarding privacy analytics, including stakeholders, privacy auditability requirements, the requirements for the ASCLEPIOS framework, as well as transparency requirements from the GDPR, we became more aware of the importance and how sensitive the audit logs are. Furthermore, we identified the absence of a model for the audit logs and a process to capture and store them in the ASCLEPIOS framework. After discussions among the project partners, it was decided that the Registration Authority component should store the logs captured during the data access requests from the user interface. Moreover, we proposed an audit logs model and dataflow to capture the logs as a suggestion for development during the integration task in WP5.

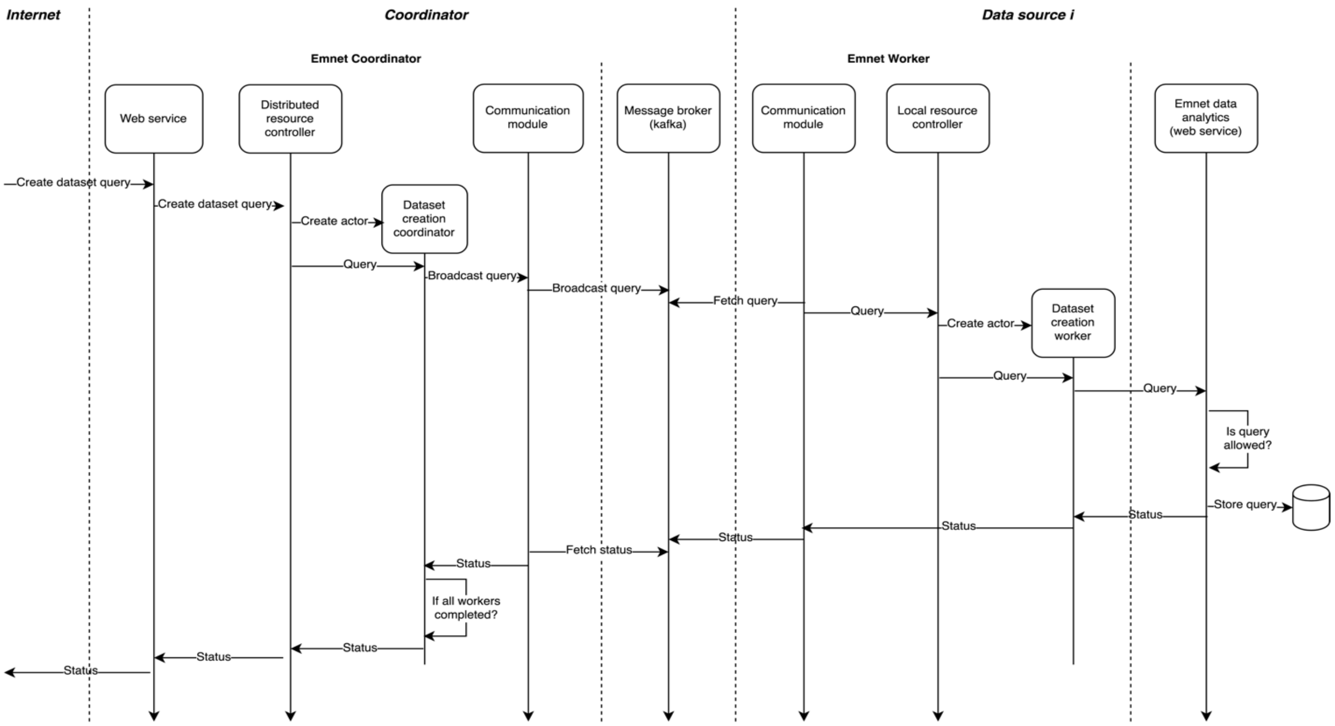
[94] Evmorfia Biliri, Nefeli Bountouni. D2.3 "GDPR - compliant and Privacy-Preserving Analytics for Healthcare Providers". May 2020.

The main result of Task 2.6 is the ASCLEPIOS Privacy Analytics Module (APAM). The design of APAM considered the two stakeholders mentioned above, patient and DPO, and the main objective of guaranteeing transparency about the patient's privacy and enabling analytics on the audit logs. APAM offers a toolkit for audit logs analytics in which the users can interact and determine the attributes that they want to analyze and the metrics that they want to visualize. Furthermore, APAM offers to the DPO a basic event outlier detection tool through the k-means clustering algorithm. The implemented outlier detection function has support for automatically defining the necessary parameters, with no human intervention necessary for configuration. APAM offers a REST API to enable integration into any healthcare applications. Finally, APAM also used ElasticSearch and Kibana to prototype and illustrate the privacy analytics tools that can be implemented from audit logs.

Annex. I: Sequence diagrams of Emnet

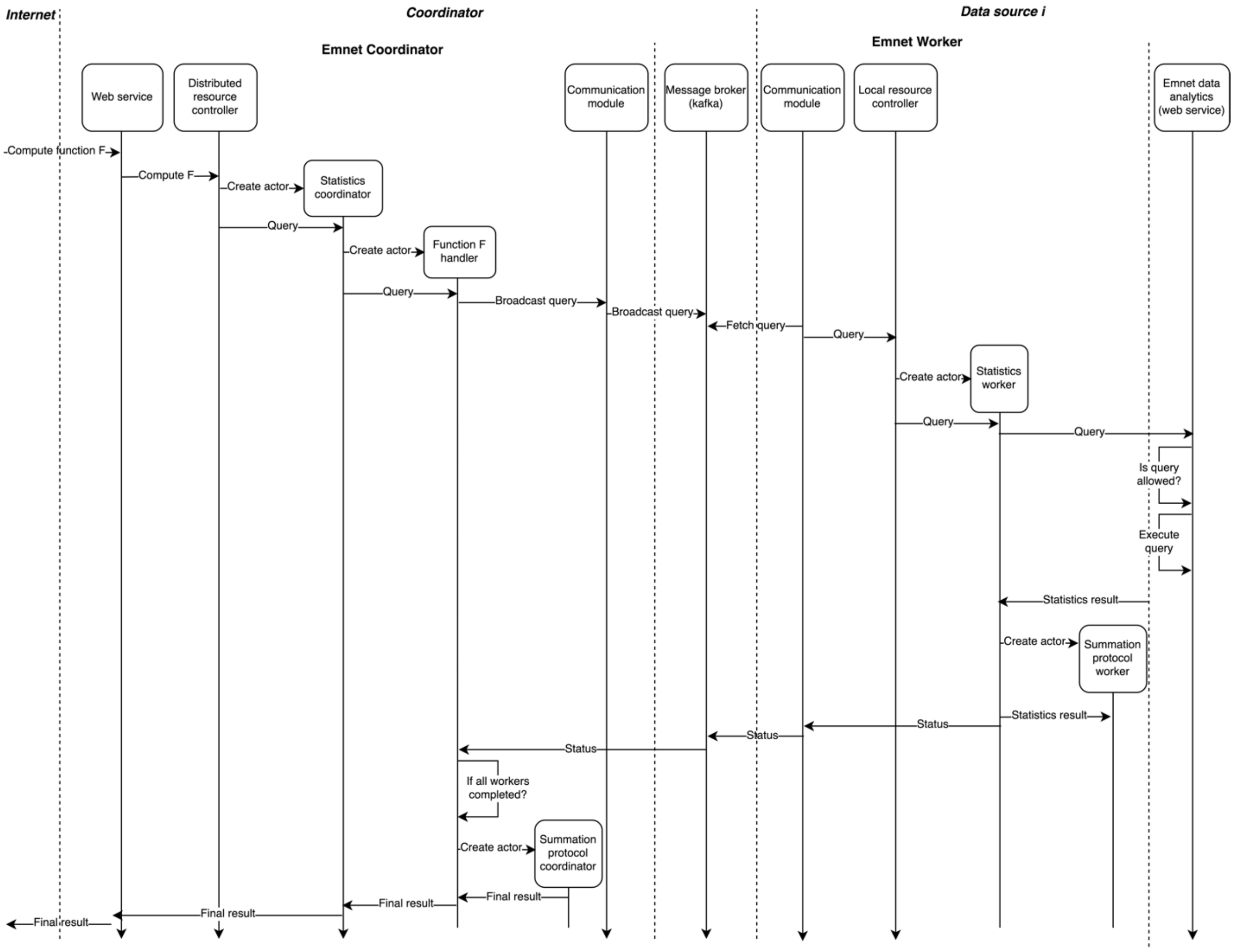
Annex. I.A: Dataset definition sequence diagram

The following sequence diagram shows how the different components of Emnet execute a dataset definition query.



Annex. I.B: Statistical computation sequence diagram

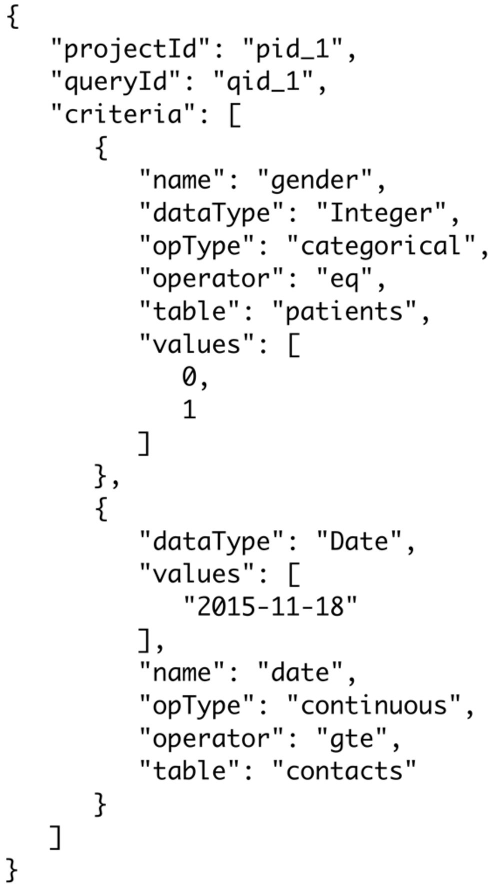
The following sequence diagram shows how the different components of Emnet execute a statistics query.



Annex. II: Sample Emnet queries and result

Annex. II.A: Dataset definition query

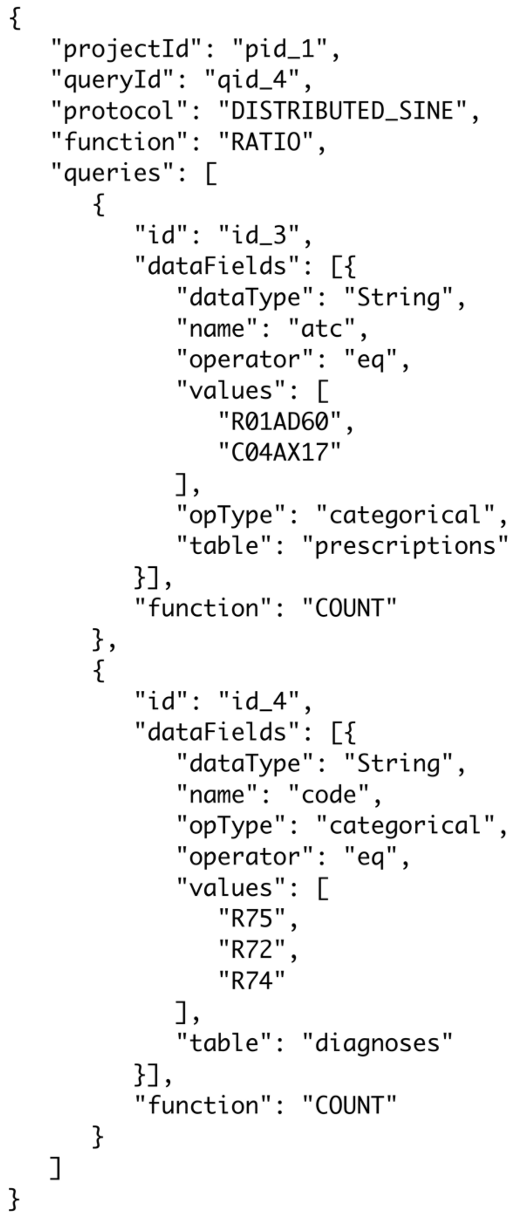
The following dataset definition query selects records of both male and female patients who had a healthcare visit after November 18, 2015.



```
{
   "projectId": "pid_1",
   "queryId": "qid_1",
   "criteria": [
      {
         "name": "gender",
         "dataType": "Integer",
         "opType": "categorical",
         "operator": "eq",
         "table": "patients",
         "values": [
            0,
            1
         ]
      },
      {
         "dataType": "Date",
         "values": [
            "2015-11-18"
         ],
         "name": "date",
         "opType": "continuous",
         "operator": "gte",
         "table": "contacts"
      }
   ]
}
```

Annex. II.B: Statistics query

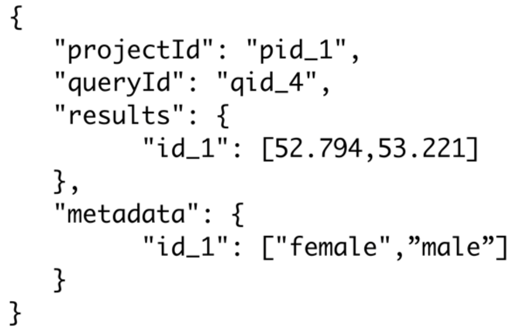
The following query computes the ratio of the number of selected drugs prescribed to the number of patients diagnosed with selected diagnoses.



```
{
   "projectId": "pid_1",
   "queryId": "qid_4",
   "protocol": "DISTRIBUTED_SINE",
   "function": "RATIO",
   "queries": [
      {
         "id": "id_3",
         "dataFields": [{
            "dataType": "String",
            "name": "atc",
            "operator": "eq",
            "values": [
               "R01AD60",
               "C04AX17"
            ],
            "opType": "categorical",
            "table": "prescriptions"
         }],
         "function": "COUNT"
      },
      {
         "id": "id_4",
         "dataFields": [{
            "dataType": "String",
            "name": "code",
            "opType": "categorical",
            "operator": "eq",
            "values": [
               "R75",
               "R72",
               "R74"
            ],
            "table": "diagnoses"
         }],
         "function": "COUNT"
      }
   ]
}
```

Annex. II.C: Statistics result

The following query computes the average age of female and male patients in the study population.



```
{
   "projectId": "pid_1",
   "queryId": "qid_4",
   "results": {
         "id_1": [52.794,53.221]
   },
   "metadata": {
         "id_1": ["female",”male”]
   }
}
```