

An architecture for secure data management in medical research and aided diagnosis

Micael Cardoso Gonçalves Pedrosa

Tesis doctoral UDC / 2021

Directores:

Julián Alfonso Dorado De La Calle

Carlos Manuel Azevedo Costa

Programa de doctorado en Tecnologías de la Información y las Comunicaciones



Dr. Julián Alfonso Dorado De La Calle, Profesor catedrático del área de Ciencias de la Computación e inteligencia Artificial de la Universidade da Coruña.

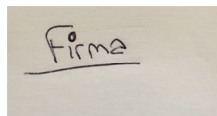
Dr. Carlos Manuel Azevedo Costa, Profesor asociado con agregación del departamento de Electrónica, Telecomunicaciones e Informática de la Universidad de Aveiro.

AUTORIZAN:

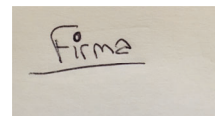
La presentación para su depósito de la tesis que dirige y que fue realizado por D. Micael Cardoso Gonçalves Pedrosa con título “An architecture for secure data management in medical research and aided diagnosis”.

Y para que así conste, firma esta autorización en A Coruña, a 12 Septiembre 2021

Los directores de la tesis



Fdo. Julián Alfonso Dorado De La Calle



Fdo. Carlos Manuel Azevedo Costa

*A todas esas personas que han perdido parte
de su valioso tiempo en enseñarme cosas nuevas*

Acknowledgements

First and foremost I would like to thank my thesis directors Dr. Julián Alfonso Dorado De La Calle and Dr. Carlos Manuel Azevedo Costa for their wise advice and dedication provided during the infinite hours of simple tutoring which would hardly have been possible to carry out this thesis. I would also like to show appreciation to Dr. André Ventura da Cruz Marnoto Zúquete for his expertise and revision of cryptographic concepts, which helped me to evolve in this field.

I want to thank all members of Centro de Investigación en Tecnologías de la Información y las Comunicaciones de A Coruña (CITIC) and the Institute of Electronics and Informatics Engineering of Aveiro (IEETA) for the good reception I had from day one, which would not have been possible without Dr. Diogo Rodrigo Marques Pratas helping me to take my first steps towards the world of research.

Lastly, I have to mention this work was partially financed by the ERDF - European Regional Development Fund through the Operational Programme for Competitiveness and Internationalization - COMPETE 2020 Programme, and by National Funds through the FCT - Fundação para a Ciência e a Tecnologia (Portuguese Foundation for Science and Technology) within project CMUP-ERI/TIC/0028/2014.

” *Our virtues and our failings are inseparable, like force and matter. When they separate, man is no more.*

— **Nikola Tesla**

Resumo

O Regulamento Xeral de Protección de Datos (GDPR) implantouse o 25 de maio de 2018 e considérase o desenvolvemento máis importante na regulación da privacidade de datos dos últimos 20 anos. As multas fortes defínense por violar esas regras e non é algo que os centros sanitarios poidan permitirse ignorar. O obxectivo principal desta tese é estudar e propoñer unha capa segura/integración para os curadores de datos sanitarios, onde: a conectividade entre sistemas illados (localizacións), a unificación de rexistros nunha visión centrada no paciente e a compartición de datos coa aprobación do consentimento sexan as pedras angulares de a arquitectura proposta. Esta proposta faculta ao interesado cun papel central, que permite controlar a súa identidade, os perfís de privacidade e as subvencións de acceso. Ten como obxectivo minimizar o medo á responsabilidade legal ao compartir os rexistros médicos mediante o uso da anonimización e facendo que os pacientes sexan responsables de protexer os seus propios rexistros médicos, pero preservando a calidade do tratamento do paciente. A nosa hipótese principal é: os conceptos Distributed Ledger e Self-Sovereign Identity son unha simbiose natural para resolver os retos do GDPR no contexto da saúde? Requírense solucións para que os médicos e investigadores poidan manter os seus fluxos de traballo de colaboración sen comprometer as regulacións. A arquitectura proposta logra eses obxectivos nun ambiente descentralizado adoptando perfís de privacidade de datos illados.

Resumen

El Reglamento General de Protección de Datos (GDPR) se implementó el 25 de mayo de 2018 y se considera el desarrollo más importante en la regulación de privacidad de datos en los últimos 20 años. Las fuertes multas están definidas por violar esas reglas y no es algo que los centros de salud puedan darse el lujo de ignorar. El objetivo principal de esta tesis es estudiar y proponer una capa segura/de integración para curadores de datos de atención médica, donde: la conectividad entre sistemas aislados (ubicaciones), la unificación de registros en una vista centrada en el paciente y el intercambio de datos con la aprobación del consentimiento son los pilares de la arquitectura propuesta. Esta propuesta otorga al titular de los datos un rol central, que le permite controlar su identidad, perfiles de privacidad y permisos de acceso. Su objetivo es minimizar el temor a la responsabilidad legal al compartir registros médicos utilizando el anonimato y haciendo que los pacientes sean responsables de proteger sus propios registros médicos, preservando al mismo tiempo la calidad del tratamiento del paciente. Nuestra hipótesis principal es: ¿son los conceptos de libro mayor distribuido e identidad autosuficiente una simbiosis natural para resolver los desafíos del RGPD en el contexto de la atención médica? Se requieren soluciones para que los médicos y los investigadores puedan mantener sus flujos de trabajo de colaboración sin comprometer las regulaciones. La arquitectura propuesta logra esos objetivos en un entorno descentralizado mediante la adopción de perfiles de privacidad de datos aislados.

Abstract

The General Data Protection Regulation (GDPR) was implemented on 25 May 2018 and is considered the most important development in data privacy regulation in the last 20 years. Heavy fines are defined for violating those rules and is not something that healthcare centers can afford to ignore. The main goal of this thesis is to study and propose a secure/integration layer for healthcare data curators, where: connectivity between isolated systems (locations), unification of records in a patient-centric view and data sharing with consent approval are the cornerstones of the proposed architecture. This proposal empowers the data subject with a central role, which allows to control their identity, privacy profiles and access grants. It aims to minimize the fear of legal liability when sharing medical records by using anonymisation and making patients responsible for securing their own medical records, yet preserving the patient's quality of treatment. Our main hypothesis is: are the Distributed Ledger and Self-Sovereign Identity concepts a natural symbiosis to solve the GDPR challenges in the context of healthcare? Solutions are required so that clinicians and researchers can maintain their collaboration workflows without compromising regulations. The proposed architecture accomplishes those objectives in a decentralized environment by adopting isolated data privacy profiles.

Contents

1	Introduction	1
1.1	Contributions	4
1.2	Document structure	5
2	Background	7
2.1	Legal framing	7
2.2	Digital identities	9
2.2.1	History	9
2.2.2	Self sovereign identities	11
2.2.3	Citizen’s card	12
2.3	Distributed ledger technology	13
2.3.1	DLT landscape	15
2.4	Cryptographic tools	17
2.4.1	Schnorr’s signatures	19
2.4.2	Threshold secret sharing	20
2.4.3	Lagrange interpolation	20
2.4.4	Verifiable secret sharing	22
2.4.5	Joint random VSS	22
2.4.6	Implicit notation	23
3	Related Work	25
3.1	GDPR impacts on medical systems	25
3.1.1	Identification and data linkage	26
3.1.2	Electronic consent	27
3.1.3	Principles of data management	27
3.1.4	Automated decision-making	28
3.2	Identity management	28
3.2.1	Sovereign identities	29
3.3	Security and privacy of EHR	30
3.3.1	Authorisation grant	31
3.3.2	Pseudonymised vs anonymised records	32
3.3.3	Break-the-glass	34
3.3.4	Encrypted records	35
3.4	DLT adoption	36

3.5	Metadata frameworks	38
3.5.1	Medical standards	39
3.5.2	Adopting semantic web	41
3.6	Related projects	41
3.6.1	Electronic health records	42
3.6.2	Science gateways	44
3.6.3	Identity and sovereignty	46
4	Hypothesis	49
4.1	Motivation	49
4.2	Research goals	50
4.3	Hypothesis	50
5	Proposed System	53
5.1	Architecture overview	53
5.2	System components	54
5.3	Threat model	56
6	Identity Management	59
6.1	Master digital identity	59
6.1.1	Trusted link groups	60
6.1.2	Card evolution	61
6.1.3	Registry	62
6.1.4	Security analysis	63
6.1.5	Results	65
6.2	Anonymous profiles	65
6.2.1	Anchors	67
6.2.2	Security Analysis	68
6.3	Authorisation	68
6.3.1	Method	70
6.3.2	Security analysis	72
6.3.3	Results and discussion	73
7	Pseudonymity	75
7.1	Master key setup	76
7.1.1	Master key setup security proof	77
7.1.2	Results and discussion	78
7.2	P-ID method	79
7.2.1	Security analysis	80
7.2.2	Results and discussion	81
7.3	Explicit and implicit consent	81
7.3.1	Implicit consent (iP-ID)	82

7.3.2	Explicit consent (eP-ID)	84
7.3.3	Impacts on the DICOM standard	85
7.3.4	Verifiable iP-ID	85
7.3.5	iP-ID security proof	86
7.3.6	Other security considerations	88
7.3.7	Results and discussion	89
7.4	Experimental setup	91
8	Data Encryption	93
8.1	Architecture	93
8.1.1	Distributed ledger	94
8.1.2	Distributed file system	95
8.2	Storage methodology	96
8.2.1	Key integrity check	96
8.2.2	Read/Write procedures	97
8.3	Security Analysis	98
8.3.1	Security proof	98
8.3.2	Informal security analysis	100
8.4	Results and discussion	101
8.4.1	File encryption/decryption performance	101
8.4.2	Meta-data encryption/decryption performance	102
8.4.3	Discussion	102
8.5	Experimental setup	104
9	Anonymous Access Token	105
9.0.1	Architecture	105
9.0.2	Contribution	106
9.1	Setup and threat model (extension)	108
9.2	Method	109
9.3	Security proof	112
9.3.1	Token forgery	112
9.3.2	Pseudonymity disclosure	115
9.4	Results and discussion	117
9.4.1	Discussion	118
9.5	Experimental setup	118
10	Conclusion	119
	Bibliography	123
	Appendix A Glossary	141
	Appendix B List of figures	147

Introduction

The purpose of this introduction is to make a brief description of the problem dealt with in this doctoral thesis and the context in which it is framed. Finally, the content of each of the chapters in which this document is structured is summarized.

The data explosion crisis is a reality in radiology. A report on radiological exam consumptions from Portugal [1] points out a high growth of ultrasound and mammography between the years of 2002 and 2006. However, these are often trapped in silos requiring the patients to carry a printed version of the exams between those silos. Moreover, according to Eric Topol, “10% of all scans in the United States are repeated unnecessarily, simply because patients cannot get hold of their past records and scans¹. In the case of medical imaging, the situation is critical since some modalities are based on ionising radiation. For instance, a person is typically exposed to about 60 μ Sv in a survey of abdominal X-rays and between 6,9 mSv and 10 mSv in a CT scan [2, 3], where the values for annual environmental radiation are situated between 2,5 mSv and 3 mSv. One of the most common reasons given for this closed setup is the fear of liability, namely being legally responsible for the patient privacy breach. Should the physician or the healthcare institution be legally responsible for any security breach? Should the patient be responsible for controlling the security of his own data?

Such situation presents a challenge not only for storage and data flow [4], but also puts a burden on physicians already in short supply [5], especially on mass screening programs where the disease coverage is low. A Computer-Aided Diagnosis (CAD) system that could rule out the presence of a disease with a high degree of accuracy and thus relieve radiologists from reading a significant proportion of images would be highly desirable. This opens opportunities for third-parties to provide CAD solutions as a service for healthcare organisations. However, potential clients of such services are concerned about the perception of justice in automated-decision making algorithms [6], as also the possible privacy issues under the flag of public

¹<https://www.theatlantic.com/health/archive/2019/06/why-arent-electronic-health-records-better/592387>

health interest [7]. CAD solutions can improve early diagnosis (a key to high rate of survival) and lower the healthcare cost. A good solution should be able to transpose the best practices of specialists to automated software, helping the decision of internal physicians. Moreover, the challenges of CAD systems are migrating from the technical aspects of performance bottleneck to the realm of security and legal constraints.

In this field, imaging information systems are denominated as Picture Archive and Communication System (PACS) and many providers already supply solutions that can promise high scalability and availability according to a “pay-as-you-go” business model in a Cloud-based service². They offer high scalability and increasing opportunities for telemedicine and cross border data exchange. However, architectures that open doors to the public network also increase the surface attack for hackers. A survey report conducted by Veritas³ from 900 business decision-makers around the world indicates that 31% believe that their enterprise is already General Data Protection Regulation (GDPR) compliant, but the analysis of the data by experts found that only 2% actually appear to be compliant. The false perception or even the uncertainty of the results suggests that the regulation is not easy to interpret. Conjugated with the heavy fines that are defined for violating those rules, it is not something that healthcare centres can afford to ignore. The regulation is lengthy, and the best approach for an initial high coverage is to lock down all data. However, there are many exceptions (i.e. medical emergencies) and implicit consent roads that break the closed box approach. Such exceptions fall into the **break-the-glass** security category [8].

Management of confidential data is fundamental to the work of clinicians and many already use smartphone applications for clinical communications [9]. However, at the moment clinicians are unaware of why they must take great care about the technical solutions they use for professional collaboration. Providing direct access of Electronic Health Records (EHR) to data owners will not only be a right under the GDPR, but it will also offer different opportunities from the patient and research perspectives like, for instance, requesting on-demand diagnosis and second opinions from other medical practitioners; fast and preliminary diagnosis from providers using machine learning techniques [10]; be notified and contribute to clinical trials where it is very difficult to find qualified patients; and offering medical data for rare disease research groups. Also, by collecting multidisciplinary datasets (health history, food and activity habits, medication prescriptions, patient location, real-time biometric

²<https://cloud.google.com/healthcare>

³<https://www.veritas.com/content/dam/Veritas/docs/reports/gdpr-report-ch2-en.pdf>

monitoring, etc) into a single trustable repository under the user control, it will enable interesting use cases: automatic tracking and analysis of life habits; contribute to the dissemination of information related with infectious diseases and its control; contribute to statistical analysis on the efficiency of medications; remote healthcare and elderly assistance.

Accessing such an amount of sensitive information requires fine control of security and trust. The conflict between privacy and scientific progress makes medicine research a sensitive topic. Data protection and sharing are now major themes in clinical research. The way this is handled today is by working with **anonymous** datasets. Anonymous data is not personal data for the purposes of GDPR, therefore no GDPR consent is required. However, the threshold for anonymization is very high. Patient anonymity is more complex than initially anticipated, DNA sequences can be linked to real-world human identities and faces can be reconstructed with 3D Magnetic Resonance Imaging (MRI) [11]. In general, these techniques are difficult [12] and not always possible [13, 14], thus so, constraining the use of certain datasets without proper consent. Moreover, anonymisation of EHR can lead to issues in future research. For instance, HeLa cells line that descends from cervical cancer cells collected from an involuntary donor, Henrietta Lacks [15], in 1951, is associated with many scientific advances. From studying the behavior of salmonella inside human cells [16] to poliomyelitis [17] and papillomaviruses [18] vaccines.

Consent, anonymity and privacy were not issues at the time, as well as any existence of a legal framework. The research has surely saved many lives, however, according to Lawrence Lacks, her eldest son, cells are commercialized for profit while her own family is unable to pay for health care. From Lawrence words, “My mother would be so proud that her cells saved lives..., pharmaceutical companies is making money but my wife had a stroke and my son has to care for her at home because they kicked her out of the hospital when her insurance ran out”. Moreover, this story has raised other ethical concerns [19], such as disclosing genetic traits borne by surviving family members. HeLa cells are immortal cells (reproduced in vitro) until nowadays. Sadly, this trait combined with the lack of tracing and misidentification of such cells contaminates the scientific literature [20], hindering other types of research.

If patient data cannot be anonymised efficiently while maintaining the necessary information for research purposes, there should be at least consent from the owner for that specific purpose. There is a lot of work to be made in providing a secure methodology to access and integrate existing medical records.

1.1 Contributions

Existing solutions are focused on solving anonymity/pseudonymity, encryption, key-management, break-the-glass and shareable records as independent problems. To the best of our knowledge, there is no consistent architecture that combines all of those features in a GDPR context with a compatible PACS interface. The following publications (from the author of this thesis) present original contributions that identify and tackle these challenges:

- Journal publication “SCREEN-DR: Collaborative platform for diabetic retinopathy” published in “International Journal of Medical Informatics” with DOI [www.doi.org/10.1016/j.ijmedinf.2018.10.005](https://doi.org/10.1016/j.ijmedinf.2018.10.005). A platform to annotate fundus images, aiming at the creation of machine learning algorithms to facilitate the screening process.
- Conference paper “GDPR impacts and opportunities for computer-aided diagnosis. Guidelines and legal perspectives” presented at “2019 IEEE 32nd International Symposium on Computer-Based Medical Systems (CBMS)” with DOI [www.doi.org/10.1109/CBMS.2019.00128](https://doi.org/10.1109/CBMS.2019.00128). Identifies main GDPR challenges, using CAD as a use-case. Identity management and pseudonymisation are identified as important blockages to achieving our goals.
- Journal publication “RAIAP: renewable authentication on isolated anonymous profiles. A GDPR compliant self-sovereign architecture for distributed systems” published in “Peer-to-Peer Networking and Applications” with DOI [www.doi.org/10.1007/s12083-020-00914-5](https://doi.org/10.1007/s12083-020-00914-5). Is the first attempt to build a distributed Self-Sovereign Identities (SS-IDs) framework that is compatible with pseudonymity, break-the-glass and key-management.
- Conference paper “Pseudonymisation with break-the-glass compatibility for health records in federated services” presented at “2019 IEEE 19th International Conference on Bioinformatics and Bioengineering (BIBE)” and awarded with a “**Best Student Paper Award**” with DOI [www.doi.org/10.1109/BIBE.2019.00056](https://doi.org/10.1109/BIBE.2019.00056). An original method based on threshold cryptography is presented as a way to combine pseudonymisation and break-the-glass. This idea is reused in further developments.

- Journal publication “A pseudonymisation protocol with implicit and explicit consent routes for health records in federated ledgers” published in “IEEE Journal of Biomedical and Health Informatics” with DOI [www.doi.org/10.1109/JBHI.2020.3028454](https://doi.org/10.1109/JBHI.2020.3028454). The idea presented in the BIBE conference is extended with an architecture that includes implicit and explicit consent routes.
- Conference paper “A safe architecture for authorisation grant in healthcare ecosystems” as a second author for the conference “ICHI 2020 : IEEE International Conference on Healthcare Informatics” with DOI [www.doi.org/10.1109/ICHI48887.2020.9374380](https://doi.org/10.1109/ICHI48887.2020.9374380). Presenting the idea of managing the patient identity with pseudonymisation compatibility, using the Citizen’s Card.
- Conference paper “Holder-of-key threshold access token for anonymous data resources” presented at “26th IEEE Symposium on Computers and Communications (ISCC 2021)” with DOI [www.doi.org/10.1109/ISCC53001.2021.9631259](https://doi.org/10.1109/ISCC53001.2021.9631259). Describes a method to generate and verify a holder-of-key access token in a (τ, η) -threshold setup.

1.2 Document structure

The remaining thesis chapters are organized in the following way:

- Chapter 2 - Takes an overview of the legal framing that should be taken into account in all phases of the project. A succinct description of the involving technologies and aspects required to connect, unify and share existing healthcare systems and medical datasets. The chapter will end with a discussion about the challenges that those aspects can put in the implementation phase.
- Chapter 3 - Presents the state-of-the-art in security and privacy research applied to healthcare systems. The chapter concludes with a list of real projects dealing with subject identification, authentication and data sharing for health records and datasets.
- Chapter 4 - The motivation and goals pursued by this research and its starting hypothesis are defined in this chapter.

- Chapter 5 - Defines the System and Threat Model that applies to all other chapters of this thesis. Presents the overall architecture, actors, use-cases and possible attack vectors in the limited bounds of the system definition.
- Chapter 6 - Provides the base for self-sovereign identity management, key revocation/renovation and authentication. These core structures are further extended in subsequent chapters.
- Chapter 7 - Introduces the main cryptographic technique of this thesis. A distributed Pseudonym Identifier Derivation (P-ID) function that extends the self-sovereign identity with pseudonymity and break-the-glass functionalities. The P-ID is a deterministic threshold protocol that prevents a single point of attack and generates a pseudonym from the subject's public information.
- Chapter 8 - Reuses the P-ID function for key management and encryption purposes. The presented method can re-generate the encryption key from a quorum of nodes with no single point of attack. The method is then integrated into a distributed PACS architecture.
- Chapter 9 - Presents a new method for generating an access token from a quorum of nodes with no single point of attack. The token is then used to set an anonymous key to access the pseudonym.
- Chapter 10 - Final remarks. Resumes achievements of the presented work and existing challenges to achieving a final product that can be used in production.

Background

This chapter introduces the necessary background and context to understand the legal context, research goals, related work and methods used in this thesis.

2.1 Legal framing

From 25 May 2018 ongoing, any company that stores or processes personal information about EU citizens must comply with the GDPR, even if they do not have a business presence within the EU. Only personal data will be under the GDPR; i.e, any form of data which can lead to the direct or indirect identification of a natural person. The GDPR defines four main roles that are responsible for ensuring compliance: **Data Subject** (the natural person that can be identified, directly or indirectly, from the data), **Controllers** (a legal person, public authority, agency or other body which, alone or jointly with others, determines the purposes and means of processing of personal data), **Processors** (a legal person, public authority, agency or other body which processes personal data on behalf of the controller) and the **Data Protection Officers** (DPO) (authorised personal auditing and enforcing the rules). DPO will be a requirement for large to medium-sized organizations. The GDPR places equal liability on Controllers and Processors, if a third-party Processor is not in compliance it also means the Controller is not in compliance.

The ratification of GDPR achieves greater individual control over personal data. The GDPR requires that personal data be protected against illegitimate processing, accidental loss, destruction or damage. IP addresses, device identification, location and cookies are now considered personal data. The GDPR demands the use of best practices to minimise the risk of a security breach (art 32). This changes the way webmasters and vendors collect, store and use such information. However the regulation is lengthy, the best first approach to avoid traversing all the regulations is to close all data access by default and tackle the issues one by one. The main key points of the regulation are:

- **Breach Notification** is mandatory in the 72 hours of first having become aware of the breach. The solution should implement detection and reporting in two distinct phases. Detection is handled individually by Controllers and Processors and its efficiency is mainly dependent on the internal implementation, where the reporting is easily implemented in a portable module.
- **Data Portability** mandates that the access to data is provided in a “commonly use and machine-readable format”. Our proposal does not intend to define portable data formats for Controllers and Processors, probably these datasets will be delivered in the same formats as they are stored. However, any new formats should be well defined and specified.
- **Right to Access** is the right that data subjects have to obtain their personal data from Controllers and to perceive if this data is being processed by others and the usage purpose. The first access should be given free of charge in an intelligible electronic format, although subsequent requests may be charged with a reasonable fee. Who has the right to access would be better handled with a single source of truth and unique digital identity when using multiple Controllers.
- **Right to Consent** and rescind that consent at any time is a fundamental requirement. However, the rescind property is not extensively possible in a practical sense. For instance, when a dataset is released to an external point, it is not possible to cut access to it because the data is not under the provider’s control anymore. It is only possible to rescind the access to the original data source. Furthermore, it is defined that professionals who are subject to professional secrecy (e.g. doctors and nurses) are able to handle certain datasets without user consent. Alternative consent channels may be required for parental consent, elderly and other technologically impaired, deceased persons or even in an unconscious state that require medical attention.
- **Right to be Forgotten** or to erasure personal data are no longer necessary. It states that data subjects have the right to have their personal data removed from Controllers and Processors under a number of circumstances. But some datasets are impossible or infeasible to remove, e.g. server backups or distributed records. This right also means that important references are also erased, and data access can be lost. In case the solution is not able to erase the records, it

should at least be impossible to link to any personal identification. So, dissecting the personal identifiers from all other types of data is a good approach.

- **Privacy by Design** mandates that the Controller implements appropriate measures to effectively meet the regulation and protect the rights of data subjects. It also mandates that Processors only use the minimum necessary information for the designated task and for a defined period of time. Complex data sharing architectures are required when dealing with several levels of authorizations, or when the same data is shared with multiple interested parties. For instance, how should emitted assertions of criminal records be handled? How private and sensitive data should be managed in the same space as public data for the same entity? The dissection of data from the previous requirement would also be a good solution to this requirement. By using data profiles, it would be conceivable to provide only isolated subsets of data, e.g. releasing medical records without exposing the patient identification elements, if such information is not needed for a correct diagnosis.

2.2 Digital identities

A digital identity is coded information that is able to authenticate and represent a unique entity, be it a person, organization, application or device. The concept of Digital Identities is addressed here because the Universal Patient Identifier (UPI) is utterly important when connecting multiple records of the same patient through different healthcare providers and data curators.

2.2.1 History

The concept of digital identities has evolved over many years. In the blog “The Path to Self-Sovereign Identity”¹, Christopher Allen identifies several phases of this evolution, although our identified phases differ when focusing on personal identities:

- **Dispersed** - The same user identity is fragmented throughout multiple websites and applications, users do not control their digital identity and it is very difficult to track all the digital presences. Passwords and other critical aspects

¹<http://www.lifewithalacrity.com/2016/04/the-path-to-self-sovereign-identity.htm>
1

are troublesome, requiring iteration with all the fragmented services to update the identity information.

- **Federated** - Control is put into multiple federated authorities, allowing the use of the same credentials to login into multiple services. The Liberty Alliance Project² was a major effort on decentralizing and providing control to end-users. This contributed to widely recognized standards, such as the Security Assertion Markup Language (SAML). However the result was an oligarchy, the control is still out of the hands of users, opening the doors to user impersonation, corporate espionage and denial of service.
- **User-Centric** - It was proposed in the Augmented Social Network white paper [21] and, as the term suggests, users have control of their identity elements and processes. However, much confusion exists about the steps necessary to make the process truly user-centric. The idea is to make identities fully portable between providers by using modern standards like OpenID Connect³. Yet, the digital identities are still maintained under the control of third party services.
- **Self-Sovereign** - The concept means truly user-centric (under user control) and portable, not only in the authentication and authorization processes but also in the identity itself. The digital identity should be usable without any provider and be able to maintain integrity in any kind of storage. These requirements can partially be assured with public-key cryptography, however, it is uncertain how to identify missing information that can lead to revoked parts of the identity.

Microsoft Passport [22] was one of the firsts attempts to introduce the concept of a unified online identity, by enabling consumers to log into different sites using a single username and password. This idea is now present in modern Single Sign-On (SSO) protocols, such as OpenID Connect which offers similar features and extensions. The use of digital identities is now so widespread that its definition has shifted to encompass the entire online information of a person. This has profound implications on the history line. Instead of migrating to a Self-Sovereign system, we are actually spreading the user information and reverting to the dispersed phase. In order to escape this divergence, it is necessary to have standards and portable structures that do not introduce more features, but integrate existing ones and place them under the control of the data subject.

²<http://www.projectliberty.org>

³<http://openid.net/connect>

2.2.2 Self sovereign identities

SS-IDs means that individuals or organizations own their personal identity data, and have efficient methods to provide it to others without relying on a central repository. “Sovereign” means that the digital identity cannot be taken away from the owner. It has the goal of replacing all paper documents with digital records. There are three important concepts in SS-IDs: a **claim** is an assertion made by some identity about a piece of information that is not necessarily true, e.g. name, birthday or nationality; a **proof** is some sort of document that provides evidence for the claim, e.g. passports, birth certificates or ownership documents; and finally, **attestations** are emitted by authoritative third parties that can validate the claim according to their records. Proofs can sometimes be faked and extra steps are required to prove the authenticity of the claim. In digital identity management systems, proofs can mostly be discarded, since signed attestations provide better ways of proving claims. Electronic records also have the advantage of being processed easily with computation and presenting faster verifications. Attestations may require access to the internal records of the authority containing sensitive information. Those should be carefully crafted to reveal only the important information, and not more, e.g. attest that a person is “over 18” or that he “can drive cars”, revealing only the necessary information without disclosing the real birthday. The SS-IDs should comply with three fundamental requirements:

- **Sovereignty** - Users must have control over their identities and the data sharing process. Be able to supervise consent and data access without needing a central repository. Be able to find where their identity is being used and communicate with those points for self-governance, e.g. update structures and revoke keys.
- **Integrity** - No other entity, other than the identity owner, shall be able to modify or add information to the digital identity. The data should be tamper-resistant no matter where the structures are stored (be it in private or public environments).
- **Portability** - The digital identity should be self-contained and portable. The export and import formats should be standardized and easily readable by humans.

In figure 2.1 is presented how we see the SS-ID concept, a self-governance portable bubble with pointers towards external services, data structures and other identities,

as opposed to a IdP centric view. Federated nodes can still exist, but they have a relegated role in providing accessibility and redundancy.

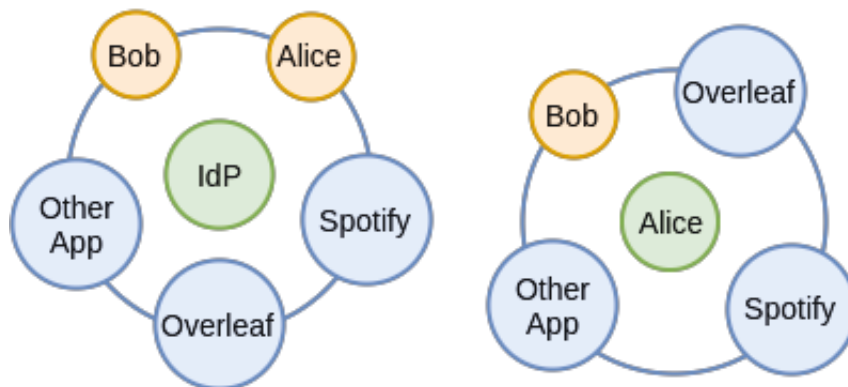


Figure 2.1.: From left to right: the provider-centric view and the identity-centric view, with applications and other identities as external points of communication.

Self-Sovereign principles have already been proposed to control the synchronization of data between health sensors and the respective online account [23]. The UPI debate has been reopened in some countries⁴ in order to solve the problem of correlating patients across data curators, but there are several problems with this. First, once there is a single identifier, multiple activities can be correlated wherever the identifier shows up. Second, the UPI by itself does not solve the integration problem. And third, it does not help with consent since there are no built-in mechanisms to manage patient consent. A Self-Sovereign identity system can present solutions to all of these with the correct use of claims and attestations.

2.2.3 Citizen's card

In 2007, as part of a technological upgrade in the public infrastructure of the public services in Portugal, the Portuguese state started to issue the Portuguese Citizen Card (CC) to its citizens. This document replaces the previous national identity card and aggregates information about other documents like National Health System ID, Tax Payer, Social Security and Electoral Card [24].

The CC is a smart card with multiple legal capabilities. For instance, it is possible to authenticate and perform digital signatures using asymmetric cryptography [25]. The authentication is activated when the citizen receives the card and proves he is the citizen by biometric means. On the other hand, the signature is activated if asked

⁴<http://www.modernhealthcare.com/article/20161006/NEWS/161009950/ahip-blues-push-congress-to-lift-ban-on-patient-identifier>

for and if the citizen has more than 16 years old. Both capabilities are protected by a Personal Identification Number (PIN) and the public keys are defined in X.509 certificates issued by a National Entity with a certificate chain, thus allowing to verify or revoke Citizen IDs. The CC certificate must obey the chain of certificates, including the National Entity's ones and the root certification authority. Therefore, it is possible to determine if the information is correct and accurate [26].

The citizen's private keys are written in the citizen card chip and cannot be extracted. The substitution of those keys depends on the issue of a new citizen card and revoking of the certificates of the previous one. These properties implicate that the operations of authentication and digital signing must be performed by the internal chip of the citizen's card and unlocked upon insertion of the personal PIN. Such characteristics allow the system to authenticate and verify the physician's identity using the terminal.

2.3 Distributed ledger technology

The term Distributed Ledger Technology (DLT) emerged as a more general description for blockchain technology, introduced by Bitcoin⁵. DLT refers to the infrastructure, protocols and mechanisms for reaching consensus on a distributed ledger in a time-sequential and immutable manner. Several consensus solutions and variants are available in historical bibliography [27, 28, 29], that sometimes proved to be difficult to implement then originally anticipated [30], and even harder if byzantine faults [31] are considered. All consensus algorithms in an asynchronous environment are constrained by the Fischer, Lynch and Patterson (FLP) impossibility [32], defining three properties referred to as **termination**, **agreement** and **validity**. Termination defines the final and irreversible decision value for a transaction, where agreement means that the decision value is unanimous for every non-faulty process, relative to the same transaction. The most important achievement of Bitcoin was the simplification of the Byzantine consensus protocol with the use of cryptographic methods in order to solve the "double-spend" problem. The Proof of Work (PoW) method was introduced to minimise Denial of Service (DoS), spam and sybil attacks [33]. However, the provided simplification relaxes the termination property. A number of predefined blocks are required so that the termination is considered with a high probability, but never 100%.

⁵<https://www.bitcoin.com>

DLT excels at the so-called “frenemy” business model, a concept comprised of competing organisations in the same sector trying to share common interests. It provides a single source of truth between parties that do not fully trust each other. Classic distributed consensus protocols fall into two major theoretical categories:

- **Leader-Based** - Is defined when a leader is selected on the network to decide what transactions are to be committed. In this way, concurrent transactions are resolved in one central point. The leader election is an expensive process, but it is supposed to be a rare occurrence. Networks of this type are moderately fast, around 1000 transactions per second with a latency of seconds. Yet, due to the existence of a central decision-maker, pure leader-based byzantine protocols are not possible and the network is susceptible to DoS attacks.
- **Vote-Based** - These have no leader election. In a rough definition, the consensus is determined when votes from a majority (over 50%) or super-majority (over 33% for byzantine protocols) are collected. Since there is no single point of failure, it is resilient to DoS attacks, maintaining the high availability status. When every vote is known and confirmed by every other node, byzantine protection is possible. However, this normally requires $O(n^2)$ messages of n participants and multiple phases (prepare, commit) before reaching an agreement, and thus, pure vote-based consensus systems have (in general) low throughput and are not scalable.

Many practical solutions mix both the theoretical approaches and/or relax some of the consensus requirements. The blockchain and Proof of Work (PoW) solution is a type of DLT that has been born from the necessity of building a byzantine consensus at a planetary scale for financial transactions, solving the double-spend problem. Traditionally, only vote-based protocols can achieve byzantine and DoS protection, but due to scalability problems of traditional consensus algorithms, the PoW opted for a solution based on cryptographic methods, making it viable with gossip protocols and avoiding multiple consensus phases. Yet, because of that, PoW does not guarantee finalization (finalization is based on statistical convergence) and the transaction values are considered final after a number of approved blocks. But, race conditions are still possible and may originate soft forks on the chain, leading to deleted transactions. The most common solution to reduce these consequences is to put a limit on the accepted blocks per minute, one of the main reasons for the 10 minutes interval of Bitcoin blocks, revealing that DLT are not without scalability

limitations. However, DLT has some important features that are difficult to find in other systems, namely:

- **Availability** is defined as the proportion of time a system is in a functioning condition. Distributed environments have a natural capability of re-routing faulty services in order to maintain availability, and since data is distributed and replicated across multiple nodes, there is a probable fall-back node capable of replacing a faulty one. Availability is important in medical systems that many times require immediate access to information in order to assist urgent cases.
- **Reliability** is the capability of a system to repeatedly return the correct result, even if the result is a well-defined error. Basically, the undefined behaviour is what destroys reliability. The most reliable systems work with Byzantine consensus protocols because these are capable of detecting and discarding incorrect results.
- **Integrity** is the capability of the system to maintain data intact and unaltered between transfer, storage and usage. The possible ways of data corruption fall into two categories: technical issues and security flaws. Ledger integrity is maintained by using cryptography, in order to detect incorrect blocks of data (normally related to security flaws), and by decentralization, in order to protect against technical issues.
- **Immutability and Irreversibility** is the assurance that committed transactions cannot be altered or removed by a single node. We can interpret immutability and irreversibility with a slight difference, the former refers to incapacity to change a transaction and the latter to the incapacity to undo a transaction. These properties are capable of extending the integrity property to the whole transaction history. This means that the ledger works in append mode, and corrections can only be performed if some type of compensation transaction exists.

2.3.1 DLT landscape

Blockchain is one form of DLT with a sequential structure of linked blocks, secured using cryptography. Other forms use a Directed Acyclic Graph (DAG) structure (e.g.

Tangle⁶) for the ledger with similar security constraints. The blockchain vs DAG is comparable to the synchronous vs asynchronous execution model. As depicted in figure 2.2, blockchain is essentially a continuously growing sequence of records where the DAG can also grow in parallel. The parallelization increases transaction throughput, but it also incorporates more race conditions that the DAG model must solve. Forks on the blockchain are forbidden, and thus, sequential processing of transactions is the only option.

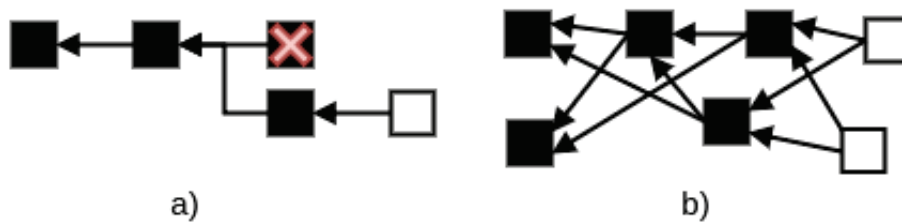


Figure 2.2.: Blockchain sequence *a*) with the next block referencing the previous one; forks are forbidden. DAG *b*) where the next block references two previous blocks. Non filled boxes represent non confirmed transactions.

The landscape is also defined by permissioned vs permissionless ledgers, where permissioned entails a core network with identified and privileged updating nodes. These can be pre-selected by authoritative entities or defined by some rules, as is the case for Proof of Stake (PoS) protocols. Many new DLT (e.g. Hyperledger Fabric⁷, R3 Corda⁸, Tendermint⁹) are exploring permissioned ledgers for enterprise use-cases, offering several advantages in this context:

- **Governance** - having a set of pre-defined nodes can simplify the establishment of policies, coordination and continuous monitoring of development between organizations. Enterprises looking for process automation in a consortium, demand a network capable of moving in different directions and optimising for different business cases. Also, the fact that mining pools are now more centralized than originally anticipated [34], indicates that governance may play a significant role on security concerns.
- **Privacy** - although permissionless networks can also keep privacy, by freely joining the network we can break that premise. In a permissioned network, sensitive data can be closed to a set of nodes without the data owner actually

⁶https://iota.org/IOTA_Whitepaper.pdf

⁷<https://www.hyperledger.org/projects/fabric>

⁸<https://www.r3.com>

⁹<https://tendermint.com>

needing to trust in a single node. Nodes can also be reused to construct private off-ledger storage and protect shared secrets [35].

- **Scalability** - permissioned ledgers can use traditional models to establish consensus (e.g. RAFT, Paxos or PBFT), preventing the waste of computational cycles of PoW algorithms. Also, by trusting in a set of privileged nodes, byzantine algorithms (in general with less performance) can be avoided entirely.
- **Access Control** - permissioned ledgers allows for fine-grained restricted access. For instance, PoW original idea was first proposed by Cynthia Dwork et al [36] as a solution to limit email spam. But, having a set of nodes controlling who can write to the ledger is a cheaper alternative to this issue.

This set of features contributes to a single source of truth without actually having to trust an isolated node on the network. In general, DLT drastically reduce the cost of trust, by providing a common data source between competing organizations interested in a common set of information, maintaining a minimal operating cost. This cost is even more reduced in permissionless ledgers, but the lack of privacy, scalability, resource efficiency from most consensus protocols, verifiable identities and governance make it difficult to be accepted in enterprise environments. Permissioned networks, as already mentioned, have a set of features that fits better in our use case. Governance is utterly important for maintaining compliance with legislation on further developments. Also, these networks normally require a scarce asset for their consensus protocol, be it processing power or a stake currency. This is used to minimize byzantine behaviour, considering that lying to the network will waste this rare asset. Yet, it is not expected that healthcare providers consume large amounts of computing resources in a consensus protocol. Assuming there is a degree of trust between ledger nodes, these are faster and cheaper to maintain. The value provided by the network itself is an incentive to participate, removing the need for a currency as an incentive method to participate.

2.4 Cryptographic tools

Our mathematical constructions are defined in a prime field $(\mathbb{F}_p, +, \cdot, 0, 1)$ where p is a large prime defining the order of the field. (\mathbb{F}_p^*, \cdot) is the multiplicative group of \mathbb{F}_p . $\mathbb{G} = E(\mathbb{F}_p)$ is an additive group defined by an elliptic curve E in \mathbb{F}_p . We assume that elements from \mathbb{F}_p and \mathbb{G} have binary formats that can be used for concatenations, for

instance $(a||P)$. Such transformations are implicit in our constructions. It is assumed the random oracle model, the hardness of the Discrete Logarithm Problem (DLP)¹⁰ and Computational Diffie-Hellman (CDH)¹⁰ for elliptic curves.

Definition 1 Let $H_p : \{0, 1\}^n \mapsto \mathbb{F}_p^*$ define an one-way hash function with preimage and second-preimage resistance, where a stream of n bits is mapped to a positive number in \mathbb{F}_p . For instance, $H_p(D) = h$, given the input data D in a binary format, results in the output $h \in \mathbb{F}_p^*$. It is computationally infeasible to derive D by knowing h .

Definition 2 Let $\times : \mathbb{F}_p \times \mathbb{G} \mapsto \mathbb{G}$ define the one-way scalar multiplication for the additive group \mathbb{G} . For instance, given the input $u \in \mathbb{F}_p$ and a generator point $G \in \mathbb{G}$, $u \times G = P_u$ outputs the point P in the curve. If the hardness of the DLP is assumed, u cannot be derived when P and G are given.

Definition 3 Let $E_k : \{0, 1\}^n \times \{0, 1\}^m \mapsto \{0, 1\}^l$ define a symmetric encryption function accepting a stream of n bits and a symmetric key k with m bits, outputting an encrypted stream of l bits. For instance, using $(s, \text{plaintext})$ as inputs we get $E_s[\text{plaintext}] = \text{ciphertext}$.

Definition 4 Let $e : \mathbb{G}_1 \times \mathbb{G}_2 \mapsto \mathbb{F}_{p^k}^*$ define a type-3 bilinear pairing [37] in Symmetric XDH settings (definition 5 from [38]). $\mathbb{G}_1 \neq \mathbb{G}_2$ are additive groups with no efficiently computable homomorphism $\phi : \mathbb{G}_2 \mapsto \mathbb{G}_1$. $\mathbb{F}_{p^k}^*$ is the extension field for the multiplicative group. Let $P \in \mathbb{G}_1$ and $Q^\dagger \in \mathbb{G}_2$, the following holds for any bilinear pairing:

1. Bilinearity: $\forall a, b \in \mathbb{F}_p : e(a \times P, b \times Q^\dagger) = e(P, Q^\dagger)^{a \cdot b}$
2. Non-degeneracy: $e(P, Q^\dagger) \neq 1$
3. e has to be computable in an efficient manner

We will use the following properties that are derived from bilinearity:

$$e(G, a \times G^\dagger) = e(a \times G, G^\dagger) \quad (2.1)$$

¹⁰<http://www.ecrypt.eu.org/ecrypt2/documents/D.MAYA.6.pdf>

$$e(P + Q, G^\dagger) = e(P, G^\dagger) \cdot e(Q, G^\dagger) \quad (2.2)$$

$$e(c \times P, G^\dagger) = e(P, G^\dagger)^c \quad (2.3)$$

2.4.1 Schnorr's signatures

Given a random secret $u \in \mathbb{F}_p^*$ and a public key generated by $u \times G = P_u$, with $P_u \in \mathbb{G}$. A Schnorr's signature is a direct application of the Fiat-Shamir heuristic [39], as follows:

$Sign(u, B) \mapsto \sigma$: The signing procedure accepts the secret u and a data block B . The signature is derived in four steps:

1. $m = H_p(u||B)$, $M = m \times G$
2. $c_\sigma = H_p(G||P_u||M||B)$
3. $r_\sigma = m - c_\sigma \cdot u$
4. Outputs the compact form $\sigma_u = \langle c_\sigma, r_\sigma \rangle$

$Verify(P_u, B, \sigma_u) \mapsto \{0, 1\}$: The verification procedure accepts the public key P_u , the data block B and the signature σ_u . The verification is done with:

1. $r_\sigma \times G + c_\sigma \times P_u = M$
2. Check if $c_\sigma \stackrel{?}{=} H_p(G||P_u||M||B)$

We assume the unforgeability of Schnorr's signatures under the random oracle conditions. Our variant uses a hash function to generate m , maintaining a deterministic output for the same data block B and at the same time setting a different m for each B that is indistinguishable from a random value, avoiding key leakage from nonce reuse. This minor change does not affect the security assumptions or the output of the Schnorr's signature.

Definition 5 Let σ_u define the output of a digital signature performed with the private key u for some data block. We will use the short notation $\sigma_u\langle B \rangle = \langle \sigma_u, P_u, B \rangle$ to represent a signature performed on the the data block B with the private key u , where the public key P_u is implicit. The default base point is G , but a signature can also be defined using a different base point, such as Y , in this case, the notation is $\sigma_u^Y\langle B \rangle$.

2.4.2 Threshold secret sharing

A (τ, η) -threshold secret sharing scheme [35] provides methods to increase redundancy without exposing the secret. Such a scheme enables $\tau + 1$ nodes to construct a secret among $\eta \geq \tau + 1$, where any subset cannot. Such a scheme has a redundancy of $\eta - \tau + 1$ shares. The scheme is information-theoretically secure, it does not rely on unproven computational hardness assumptions. Any number of τ nodes cannot give valuable information about the underlying secret.

Given a polynomial $L(x)$ of degree τ of the form $a_0 + a_1 \cdot x + \dots + a_t \cdot x^t$, $L(0) = a_0$ is the solution for $x_0 = 0$ and it is considered the secret. Other evaluations $\{L(x_1), \dots, L(x_i), \dots, L(x_\eta)\}$ are part of the set of secret shares $\{y_i \in \mathbb{F}_p : i \in [1, \eta] \cap \mathbb{Z}\}$. The method relies mostly on the Lagrange interpolation over finite fields to reconstruct the secret $L(0)$ from a set of $\tau + 1$ minimum shares. Although $i \in [1, \eta] \cap \mathbb{Z}$, we will simplify our formulations by using the minimal number of required shares, such that $i \in [1, \tau + 1] \cap \mathbb{Z}$. A short notation for a set $\{x_i \in \mathbb{F}_p : i \in [1, \tau + 1] \cap \mathbb{Z}\}$ can also be described as $\{x\}_i$. We will also use the short notation for $\sum_{i=1}^{\tau+1} = \sum_i$.

2.4.3 Lagrange interpolation

Given a polynomial $L(x)$ of degree τ , the Lagrange polynomial interpolation over a finite field is a basic construction that recovers $L(x)$ from a set of points $\{(x_i, y_i) \in \mathbb{F}_p^2 : i \in [1, \tau + 1] \cap \mathbb{Z}\}$. The Lagrange polynomial is derived from:

$$L(x) = \sum_i y_i \cdot l_i(x) \quad (2.4)$$

where,

$$l_i(x) = \prod_{m=1, m \neq i}^{\tau+1} \frac{x - x_m}{x_i - x_m} \quad (2.5)$$

Definition 6 Let $\{(x_i, y_i) \in \mathbb{F}_p^2 : i \in [1, \tau + 1] \cap \mathbb{Z}\}$ be the minimal set of shares to recover $L(x)$, and $\alpha = y_0$ be the secret, where the set of x_i are publicly known and equal to the index of the shareholder $x_i = i$. We define the evaluation of the Lagrange interpolation for $x = 0$ as $L(0) = \mathcal{L}^i(y_i) = \alpha$, where the Lagrange coefficients $l_i(0)$ are implicit in \mathcal{L}^i . \mathcal{L}^i is a mere simplification, or alias for $\sum_i l_i(0)$ that evaluates to $L(0)$ when applied to a set of secret shares. For instance, $\mathcal{L}^i(y_i) = \sum_{i=1}^{\tau+1} l_i(0) \cdot y_i$.

Given three polynomials $L_1(x)$, $L_2(x)$ and $L_3(x) = L_1(x) + L_2(x)$ of the same degree τ , the secret values $a, b, w \in \mathbb{F}_p$ where $L_1(0) = a$ and $L_2(0) = b$, and the corresponding shares $\{a_i, b_i \in \mathbb{F}_p : i \in [1, \tau + 1] \cap \mathbb{Z}\}$. The Lagrange homomorphic properties preserving the polynomial degree are defined as:

$$\mathcal{L}_3^i(a_i + b_i) = \mathcal{L}_1^i(a_i) + \mathcal{L}_2^i(b_i) = a + b \quad (2.6)$$

$$\mathcal{L}^i(w \cdot a_i) = w \cdot \mathcal{L}^i(a_i) = w \cdot a \quad (2.7)$$

$$\mathcal{L}^i(w + a_i) = w + \mathcal{L}^i(a_i) = w + a \quad (2.8)$$

In addition, there is an extra property that works with elliptic curve points (Equation (2.9)). From the distributive property $(a + b) \times G$, we can easily verify that $\sum_i [y_i \cdot l_i(x) \times G] = [\sum_i y_i \cdot l_i(x)] \times G$ is the same as:

$$\mathcal{L}^i(a_i \times G) = \mathcal{L}^i(a_i) \times G = a \times G \quad (2.9)$$

Definition 7 Let $y_i \mapsto {}^i y$ be an index transformation for $x = 0$ and $l_i(0) = l_i$ where $y_i \cdot l_i = {}^i y$.

This interesting property is defined from the equality $\mathcal{L}^i y_i = \sum_i y_i \cdot l_i = \sum_i {}^i y = y$, allowing the index transformation when applying the Lagrange basis polynomials. From the same property, we can cancel indexes when they are diagonally aligned with the operators (\mathcal{L}^i, \sum_i) , a useful property to simplify our formulations.

2.4.4 Verifiable secret sharing

Using previous definitions, the Feldman's Verifiable Secret Sharing (VSS) [40] scheme is able to verify if the shares in S_α can correctly produce a secret α without revealing those shares. When a dealer distributes the shares S_α for each corresponding party P_i , it also publishes all coefficients $A_k = a_k \times G$, $k \in [0, t] \cap \mathbb{Z}$ as auxiliary information to allow parties to check their shares. Each party accepts its share (x_i, y_i) if the following is correct:

$$\mathcal{V}(y_i) \equiv y_i \times G \stackrel{?}{=} \sum_{k=0}^{\tau} x_i^k \times A_k \quad (2.10)$$

This is the public method for evaluating the polynomial $A_0 + A_1 \cdot x + \dots + A_\tau \cdot x^\tau$ for a pre-defined i . Note that the Feldman's scheme leaks additional information $a_0 \times G = A_0$ from the secret when $k = 0$. However, a_0 will be hard to derive under the DLP assumption. Also, in our construction x_i are public parameters and are assigned to the party index such that $x_i = i$, where $i \neq 0$. The index is a number as good as any other. This is a standard procedure when using the VSS, without affecting the security of the scheme.

2.4.5 Joint random VSS

The Joint Random Verifiable Secret Sharing (JR-VSS) is a protocol capable of generating a random key pair α/P_α from a set of $\tau + 1$ parties where α is unknown and can only be recovered from $\sum_i i\alpha$. Each party i selects a random polynomial with $i a_k$ private coefficients and the respective public $i a_k \times G = i A_k$ Feldman's coefficients. Using the polynomial, each party evaluates a minimum set of $\tau + 1$ secrets shares $i\alpha_j$ (with $j \in [1, \tau + 1] \cap \mathbb{Z}$), where each party secret is $i\alpha = i a_0$. The j shares are distributed to each corresponding party j . The entire matrix of secrets $i\alpha_j$ is required to recover α .

The random secret α is recoverable from the sum of the polynomials of all parties (or the shared secrets), $\sum_i i\alpha = \sum_i \mathcal{L}^j(i\alpha_j)$. Applying Equation (2.6), the result is equivalent to $\mathcal{L}^j \sum_i (i\alpha_j) = \mathcal{L}^j \alpha_j = \alpha$. The public key is defined from $\alpha \times G = P_\alpha = \sum_i i A_0$.

Rushing adversary. The Feldman's scheme effectively avoids the rogue key attack [41] for a rushing adversary who is allowed to submit their shares after observing

the shares and Feldman's coefficients of other honest parties. An adversary j can wait for other responses and forge a point ${}^j A'_0$, trying to force the public key $P = {}^j A_0$:

$${}^j A'_0 + \sum_{i=1, i \neq j}^{\tau+1} {}^i A_0 = P \quad (2.11)$$

In this way the adversary knows the private key for the resulting P . However, the adversary j has to produce a set of secret shares such that other parties can accept them when performing the validation:

$$y_i \times G = {}^j A'_0 + \sum_{k=1}^{\tau} x_i^k \times {}^j A_k \quad (2.12)$$

The adversary has to solve the DLP for ${}^j A'_0$ to produce a correct set of shares, otherwise the Feldman's verification (Equation (2.10), similar to Equation (2.12)) will fail. This proves that a forged point has no value when trying to produce a set of secret shares.

2.4.6 Implicit notation

The following implicit notations are used: lowercase letters normally correspond to values in \mathbb{F}_p , and uppercase letters to points or sets of points in \mathbb{G} . $G \in \mathbb{G}$ is the base point (or generator) and $O \in \mathbb{G}$ is the point at infinity. G is a known public parameter of the system.

The short notation for sums is used: $\sum_{i=1}^{\tau+1} = \sum_i$. The following implicit notations are used: lowercase letters are commonly used for scalars in \mathbb{F}_p and uppercase for points in the groups \mathbb{G}_1 or \mathbb{G}_2 . All points in the form P^\dagger belong to \mathbb{G}_2 . $G \in \mathbb{G}_1$ and $G^\dagger \in \mathbb{G}_2$ are the base points (or generators) and $O \in \mathbb{G}_1$ is the point at infinity. Subscripts using i (i.e. y_i) generally represents an element or result of an operation from a node. We assume that elements of \mathbb{F}_p^* and \mathbb{G}_1 have binary formats that can be used for the H_p domain, as also concatenations of these forms, for instance $a||P$. Such transformations are implicit in our constructions.

Related Work

This chapter presents the state-of-the-art research in security and privacy applied to healthcare systems. The chapter concludes with a list of real projects dealing with subject identification, authentication and data sharing for health records and datasets.

The GDPR is already being translated into interoperability standards [42]. Hospitals and other healthcare organizations should be prepared to comply [43, 44]: giving access to data, requiring explicit consent, providing data rectification and portability, and inform about security breaches. Health records (EHR, RIS, PACS and others) are increasingly electronic, but are often still trapped into silos. Medical standards such as Digital Imaging and Communications in Medicine (DICOM), do not have the appropriate safety mechanisms for dealing with the regulation over private network borders. We can extend current solutions [45] adding consent and authorization control. However, DICOM and other protocols are not generally concerned with: linking with a unified patient identity, giving full ownership to data, implementing parental or guardian consent, inter-operate with different types of data sources (medical profiles, debts profiles, curriculum vitae, etc), capturing only the minimal personal data for legitimate use (“data minimization”); or even certifying entities comprised of the “processing of special categories of personal data” [46], e.g. doctors and nurses.

The author of this thesis believes that the future of EHR lies in a good identity management foundation, pseudonymisation of records and exploration of distributed systems and cloud storage to secure EHR.

3.1 GDPR impacts on medical systems

The most known legal constraint of the GDPR is related to consent (art 7) and is distinctively designed for medical data (art 9) where specific types of consent are required. A data protection impact assessment (art 35) is also mandatory for this type of information. Yet, in the healthcare sector, patients’ data is held under a

duty of confidence. Physicians do not require the subject's explicit consent; they are protected by exceptions (art 7) and other bases in the "lawfulness of processing" (art 6), such as the public task basis. However, to invoke this lawfulness, there should be some registry noticing the service request from the patient, meaning that, even implicit consents require some kind of record. Third-party CAD services are not in this sphere of lawfulness and also have to concern with other aspects of the regulation. Any data flow or processing from those entities requires a subject consent registry that is non-repudiable, unforgeable and tamper-resistant.

3.1.1 Identification and data linkage

The concept of a Unique Digital Identity (UDI) is important to unify multiple records between different healthcare organisations. It is critical to match the correct patient to the corresponding data that requires consent from multiple healthcare providers. This connection can be included in the consent record if it follows appropriate security requirements. It is critical to match the correct patient to the corresponding data within multiple healthcare providers. In 2017, the ECRI Institute presented the report analysis "Health IT Safe Practices: Toolkit for the Safe Use of Health IT for Patient Identification"¹ of over 7,600 patient safety events related to patient identification. Such mismatches create safety issues and can lead to adverse events [47]. What exists in the healthcare environment is suboptimal, in our opinion.

Such function has been mainly delivered to the Patient Identifier Cross-referencing (PIX) profile, from the Integrating the Healthcare Enterprise (IHE)² initiative, designed primarily to standardize the identity of patients across multiple affinity domains. The basic principle is for a consumer actor to request a list of patient identifiers from a PIX manager. From the defined profile transaction ITI-45³, a query request is fulfilled by providing one possible identifier; in return receives a list of identifiers for all contributing sites, if there are any. The assumption is that the consumer should already possess one identifier from all of the existing domains. In this sense, PIX is purely an integration profile and does not have any pseudonymisation functionalities, such as retrieving an anonymous identifier from a designated piece of information, which is one of the contributions of this thesis.

¹https://www.ecri.org/Resources/HIT/Patient%20ID/Patient_Identification_Toolkit_final.pdf

²<https://www.ihe.net>

³<https://www.openempi.org/confluence/display/openempi/PIX+v3+Query>

3.1.2 Electronic consent

Electronic consent has some advantages over paper documents. For instance, in the EU eIDAS⁴ legal framing can now be used to recognise a digital signature with the same value as a handwritten signature. This makes granular consent frameworks (such as ones based on check-boxes) more tamper-resistant. The availability of electronic records and consents facilitates its withdrawal (art 7 - 3), personal information rectification (art 16) and status change notifications. Yet, consent requires some security features. Non-repudiation is one of the most important with some attached dependencies, mainly, integrity and authenticity of records. The former is easily achieved with digital signatures, but the latter requires true SS-IDs [48]. SS-IDs is a form of user-centric (and sovereign) digital identity without relying on a centralised organisation. However this comes with some drawbacks, full sovereignty will make any break-the-glass procedure unfeasible and unpractical for any real-world application, especially in medicine where this is utterly important.

3.1.3 Principles of data management

The GDPR creates incentives and relaxes several requirements for controllers who anonymise and encrypt data, such as extending the processing to a different purpose than the one for which it was initially envisioned (art 6 - 4e). And, since pseudonymization reduces personal data exposure, breach notifications can also be diminished (art 33, 34). On the alternative approach, anonymous data is not under the GDPR. However, the threshold for anonymisation is very high. Subject anonymity is more complex than initially anticipated, DNA sequences can be linked to real-world human identities and faces can be reconstructed with 3D MRI [11]. In general, the techniques are difficult [12] and not always possible [13, 14], thus so, constraining the use of specific datasets without proper consent. The recommendation is, to not assume that anonymisation is possible.

In terms of the data retention policy, there is no absolute “right to erasure” (art 17). A subject can invoke this right only when there is no compelling reason for its continued processing. Care providers can always justify the data safekeeping for the purposes of providing continuous medical care. For CAD providers, data retention must be in accordance with “the necessity and proportionality of the processing operations” (art 35 - 7b) and the principle of “data minimisation” (art 5 - 1c). The recommendation is

⁴<https://www.eid.as>

to erase the data once the automated diagnosis is completed, minimising the surface area for subject identification.

3.1.4 Automated decision-making

Art 13 demands transparency and “meaningful information about the logic involved” for automated decisions. This right has become the subject of substantial academic attention [49, 50]. Some authors are already including it in their methodologies, not only important machine learning metrics but also paying attention to the interpretability of the model [51]. Google outlines what they considered the fundamentals of interpretability [52], by “making sense of hidden layers”. However, this always has a price in terms of accuracy. Automation is designed to reduce human error, however, also causes a false perception of the lack of accountability. Black box methodologies can lead to unpredictable results (bias and overfitting) and are prone to attacks. Moreover, when the results of automation extend to the level of experts, the output is viewed as reliable leading to automation complacency [53] and consequently to the paradoxical effect of increasing errors rather than eliminating them [54]. The complacency issue was originally identified in aviation, but from then it has also spread into the medical field [55]. The purpose of art 22 is to protect the data subject against such consequences, by ensuring human control regarding the automated decisions.

3.2 Identity management

Although the PIX profile is defined primarily as patient identification between different data curators, it has no cryptographic features. Nowadays, we mainly rely on Identity Provider (IdP) and use SSO protocols to maintain our digital identity. Google, Facebook, Twitter and others, if properly extended to the healthcare domain, may be serious candidates for identity management. Yet, these do not offer true data ownership or assure the certification mechanisms to identify users or to build a Know Your Customer (KYC) procedure. Users do not really have ownership of their digital identity as it is always “given to them”. These kinds of centralized IdPs have all the essential information to impersonate the identity without requiring the owner’s consent, and thus so, it is not a decent solution to protect sensitive information. There are novel attempts using cryptographic methods to remove the strong trust in IdP such as SlashID [56]. However, this requires a drastic change in

the authentication protocol. There are opportunities for federated architectures [57] to fill the needs of identification and consent registration. Yet, a reworking has to be made to circumvent trusts issues, such as introducing self-sovereign elements, claims and attestations into the mix. Also, the username/password method does not certify that the user is the person he claims to be. Banks, healthcare providers and other institutions spend significant time and money verifying peoples' identities in a process called Customer Identification Program (CIP) or KYC⁵. These programs are in general inefficient, cumbersome and often collect more information than is necessary, invading the customer privacy.

3.2.1 Sovereign identities

For the past several decades, the answer to sovereign identities has been Public Key Infrastructure (PKI) and Certificate Authorities (CA). The fundamental problem with PKI is that it is cumbersome, costly, centralized and, most importantly, does not represent a single source of truth, opening the possibility for counterfeited digital certificates⁶. Identity-Based Cryptography (IBC) is an alternative that discards the need of CA, simplifying public key-management, but, in general, requires personal information such as an email address, a phone number, etc, to generate the required keys, negatively affecting pseudonymity. Current improvements based on anonymous IBC [58] are designed for message transmission and do not provide any insights on how to build a non-repudiable registry. This type of scheme also has time boundaries related to their private keys, a necessity due to the lack of revocation procedures. Tracing old keys is required to maintain the backward integrity of records.

Alternatively, the use of biometrics [59] for identification works well because it is physically tied to a person. However, using immutable biometrics is not the best authentication mechanism, if those are stolen it represents a permanent breach and if revoked a permanent ban. Using SS-IDs mechanisms is preferred in order to certify any individual characteristic. The data portability also implies that data subjects be able to transfer their identity data. Standardization efforts on this domain are underway with the Decentralized Identifiers (DIDs)⁷ and Verifiable Claims Data Model and Representations⁸. Moreover, recent projects on the field [60, 48, 59] do not

⁵<https://www.thomsonreuters.com/en/press-releases/2016/may/thomson-reuters-2016-know-your-customer-surveys.html>

⁶<https://www.cnet.com/news/google-confronts-more-site-certificate-problems>

⁷<https://w3c-ccg.github.io/did-spec/#notification-of-did-document-changes>

⁸<https://www.w3.org/TR/verifiable-claims-data-model>

tackle key management (revoke and recover) and pseudonymisation in simultaneous, user profiles, or any integration with medical standards.

A parallel concept is the web-of-trust [61], where entities define trust metrics from each other and calculated trust paths, creating a graph of trusted peers. KeyChains [62] is an example of a lookup algorithm resorting to the web-of-trust model to build a Distributed-PKI and to remove the dependency of a centralised CA.

This thesis leverages the idea of IBC principles and distributed peers to manage secrets [63], avoiding a single point of attack, improving key availability and recoverability with a small impact on security. We provide a clear distinction between key recovering and identity recovering, being able to recover an identity even if there is no previous backup of the private-key.

3.3 Security and privacy of EHR

Criticisms on the current security and privacy of EHR [64] testify to the lack of protection and auditability on healthcare infrastructures. Proposed Role-Based Access Control (RBAC) models [65] only manage who can access the records, but do not consider the data subject control over privacy, because the control can be jointly managed by the data subject and by the healthcare professionals. By using cryptographic signatures and DLT as a single source of truth, the authorization to role assignment can be uniquely managed by the patient. Also, reusing the network as a message broker will allow briefing patients about who is accessing their EHR; and will also allow storing encryption keys as shared secrets [35] on the ledger and use them to encrypt data on cloud storage. Some authors actually advocate that privacy should be enforced via encryption [66], but encrypting large medical images without consuming overwhelming time appears infeasible. The pseudonymization alternative introduced by [67] does not encrypt EHR, the patient's identification and grants to the records, are instead mediated through a smartcard containing a secret key. Remote patient monitoring is likewise becoming common with sensors placed inside homes. The security and privacy implications of remote monitoring were uniquely identified in [68]. However, as stated by the author, these are not distinct to the medical field. Consolidation of cryptographic methods, distributed systems, RBAC solutions and standardized authentication/authorization protocols [69] may serve as the final solution.

3.3.1 Authorisation grant

As stated in the Integrating the Healthcare Enterprise (IHE) document⁹ section “3.1.2 Common Access Control Models”, RBAC models are best aligned for intra-enterprise access. However, in cross-domain and cloud environments (such as cloud-based PACS [45]), the patient is considered as “the sovereign of his medical data”, where Discretionary Access Control (dac) is best suited. The dac model directly states “to each resource a special property called the owner is assigned, who may exclusively grant or deny any access rights to users or other groups for this resource”, for which our proposal is qualified. Our work also follows the principle of decoupling authorisation from authentication, section “3.2 SOA Security Principles”. By doing so, our dac proposal is not an impediment to work concurrently with different models [70], such as active auditing schemes [71], integration with Electronic Consent Frameworks¹⁰ and context-based access decisions [72].

At the same time, we strive to maintain the best user experience for the subject. As the name suggests, Quick Response (QR) codes have unusual high impacts on interactivity and response [73] and are already wildly accepted to promote social collaboration. QR codes can also be used with cryptographic primitives [74] as a way to identify and certify terminals, preventing phishing attacks [75].

The OAuth2 protocol [76] is not designed to use a mobile device as part of the authorisation Server and a different endpoint as part of the client. In general, the same terminal initiates and receives the authorisation. Although OAuth2 flows can be adapted to wait for a mobile authorisation instead of redirecting the browser, the initiation of the authorisation flow is still an issue for privacy-preserving features. A privacy-preserving OAuth2 based protocol is proposed by Victor Sucasas et al. [77, 78], however, in our use case, the authorisation flow must start at the terminal, forcing the operator to insert the subject’s identity or a pseudonym. The insertion of a pseudonym degrades the system’s usability, and collecting such a pseudonym from a terminal/mobile communication (i.e. using Near Field Communication) adds unnecessary flows to the solution. Besides, in our scenario, starting the flow at the terminal is an open door for uncontrollable flooding of authorisation requests, since the authorisation is not initiated by the subject itself.

⁹https://www.ihe.net/Technical_Framework/upload/IHE_ITI_TF_WhitePaper_AccessControl_2009-09-28.pdf

¹⁰<http://dla.gov.in/sites/default/files/pdf/MeitY-Consent-Tech-Framework%20v1.1.pdf>

Multiple Factor Authentication (MFA) for the subject authentication and authorisation is not practical for the described scenario [79, 80]. Besides a mobile phone, it would require an alternative out-of-band authentication, not usable when a subject only has access to a phone. We remove the MFA requirement with minimal impact on security. Even in the case of a compromised mobile device, the data access requires an explicit request from a certified operator and terminal, minimising the risk of unauthorised access. Moreover, using biometrics [81] as a MFA alternative can be expensive for small clinics and may compromise our pseudonymisation compliance.

3.3.2 Pseudonymised vs anonymised records

Anonymity and pseudonymity have different legal framings in the GDPR. Anonymity means that someone's identity is entirely unknown, impossible to discover by any means, and is not subject to regulation. Pseudonymity implies that a different identifier is being used, rather than the legal name or any information capable of re-identifying the identity. Anonymisation efforts of DICOM files are mainly focused on completely removing patient information from meta-data and pixel data, using Optical Character Recognition (OCR) methods [82]. Although this is an important part for second usage of EHR, removing all the patient identification hinders the usage of this data for diagnosis. Furthermore, anonymised datasets are one-time constructions that cannot be incremented over time. A pseudonym must exist to implement such a feature.

For instance, anonymity can be achieved with specially designed keys using ring signatures [83]. This is a method of authenticating an anonymous message from a group of possible signers without prior cooperation. However, in the original publication, there is no way to trace the signature to its source or to disclose the identity to interested parties. A generalisation for ElGamal signature scheme was proposed in [84] and includes a non-repudiable disclosure process (convertible ring signature). The method proves the ownership of the original message, and our proposal takes some ideas from this construction. However, our non-repudiation feature extends to records (not just messages) to prove the existence of those records that, once persisted, cannot be denied. Both publications served their original intention, but do not serve our non-repudiation, data minimisation and break-the-glass requirements.

Pseudonym derivation. A pseudonym derivation is defined as the process that derives an identifier for a subject from a piece of public or private information of that subject. Such procedure can be performed using hashing [85, 86] or encryption techniques. Both forms demand storing a list of pseudonyms in order to assure a relationship between the input and the pseudonym. However, the former is not suitable for break-the-glass requirements, specifically when the pseudonym is derived from an accessible piece of public information, with the hash result easily derived from an offline procedure. The latter normally need to share encryption keys that hinder break-the-glass procedures and data sharing requirements. For instance, Bernhard Riedl et al. [67] shares the secrets of the patients in order to construct a fallback mechanism for lost, compromised or destroyed smart cards.

Rita Noumeir et al. [87] uses an encryption algorithm to generate the pseudonym; however, the author states “Although we have not studied patient consent management, we think that a consent manager is needed”. Consent management is the cornerstone of this work and is tightly integrated with the pseudonym generation. Also, the process of recovering encryption keys or exposing a break-the-glass bypass is generally provided by centralized RBAC models [88, 89, 90, 91]. In these situations, the authorised personnel must provide a reason with a pre-established time interval to remain accessible. Although access control schemes cannot be entirely avoided, from our point of view it is better to provide a separation of the access control mechanism from the break-the-glass compliant pseudonymisation scheme. In doing so, both schemes can evolve independently. The definition of the roles, who grants access to the data and what kind of information should be provided is entirely defined by external modules.

A centralized system holding sensible data is an interesting target for attackers. Riedl et al. have designed a hull-architecture [92, 67] with multiple layers of authorization mechanisms and cryptographic keys. Nonetheless, pseudonyms are still generated by encrypting the identification data. The architecture is able to share keys using threshold schemes; however, we go further and use the threshold scheme to generate pseudonyms, avoiding the management of keys associated with the patient. In this manner, patient identification is always recoverable, and there are no private keys that can be lost.

Harald Aamot et al. [93] defines a good evaluation model for pseudonymisation techniques and evaluate several existing methods in Table 1. Our method can be classified by this table as non-reversible, reproducible and patient-centric, using

single-pass, with duty-separated regarding de-pseudonymization and configurable in terms of required parties for de-pseudonymization. In fact, the configurable number of parties is a direct consequence of the threshold scheme and the main feature that stands out from other methods. This feature, allows for a custom level of security and high availability, depending on the number of trusted parties and acceptable failures.

3.3.3 Break-the-glass

Break-the-glass is a method of bypassing normal security measures in case of an emergency, following a previously established protocol and imposing audit-trail mechanisms. One approach to deal with emergency situations are Implantable Medical Devices (IMD) such as in [94]. Such devices have storage limitations and are difficult to operate when performing EHR updates. Those solutions use biometrics and internal processing units to authenticate the data-subject. Our point of view is that a Radio-Frequency IDentification (RFID) should only be used for the purpose of subject identification, deploying EHR in cloud services. However, cloud services have their own set of challenges [95], single points of failure, and insider attacks.

Most healthcare systems already allow to bypass normal flows in emergencies [8], but the concept also extends to more classical use-cases such as circumventing authentication and account problems (forgotten username/password, locked password, smart card or biometrics reader failure, etc). In general, bypass policies [96] are defined in the RBAC models [88, 89, 91, 97] and Privileged Access Management (PAM) [98]. Such demand introduces weak points in the system that should be regulated [99, 100] and audited. However, the regulation does not restrain attackers from bypassing centralized services.

This thesis defines a distributed and open pseudonymisation protocol that is compatible with break-the-glass procedures. By “open“ we mean that it is decoupled from any authentication, authorization method or query engine. The distributed nature of the proposed architecture protects encryption keys and data against localized cyber-attacks, such as insider attacks, ransomware, denial-of-service, virus and worms.

3.3.4 Encrypted records

Medical image encryption is a crucial method to improve privacy, integrity, and authentication of private data [101]. The most common method to protect private data in cloud environments rely on searchable encryption schemes [102], being CryptDB [103] one of the most used databases for medical data. However, data leakage is an unavoidable byproduct of efficient forms of searchable encryption, for instance, Kellaris et al. in [104] and subsequent improvements by Grubbs et al. [105] shows a devastating attack on the approximate database reconstruction (ϵ -ADR) problem, being able to reconstruct complete records in nearly any database size. Although there are innovations on Witness-Based Searchable Encryption (WBSE) [106, 107], it is necessary to evaluate if those schemes hold in such an aggressive environment. Furthermore, in a PACS cloud scenario, searchable encryption is not the best solution. DICOM files are considered immutable, most of the object fields could be anonymised for search compliance [108], and pixel data could be encrypted with standard symmetric encryption. The issue with such approach is to deploy a safe key escrow/management with break-the-glass compliance, which is one of the contributions of this thesis.

Key management. One of the biggest encryption headaches is key management¹¹. The main reason is the lack of clear ownership, skilled personnel and trusted key escrow provider. Key escrow is a method of safely storing important cryptographic keys; an important foundation of key management, data recovery and break-the-glass implementations, but not without associated risks [109], and in particular “insider abuse”.

Some key management architectures [110] have complex and very specific requirements and tight bounds to specific authorization protocols, such as attribute-based encryption [111]. It is possible to use biometrics [112] to generate new keys. However, biometrics is strongly tied to an identity and may represent a privacy risk. Also, those can easily be stolen [113] and cannot be re-credentialed. If compromised by spoofing [114] or forgery [115] the consequences would be unprecedented, representing a permanent breach. Moreover, direct control of encryption keys [116] does not provide reliable break-the-glass mechanisms, and some proposals [117] rely on a centralized trusted server. Furthermore, the issue of key revocation and data re-encryption [118] is many times overstated. The fact is, there is no revocation

¹¹<https://www.hipaajournal.com/study-reveals-health-information-the-least-likely-data-type-to-be-encrypted>

scheme when someone already has the ciphertext and the key. Based on this fact, we ignore the issue completely. A better approach, followed in our proposal, is to isolate DICOM sessions in different datasets using different encryption keys, protecting data that has not yet been accessed. The proposed approach of this thesis aims to be a simplistic architecture with minimal dependencies on predefined access control mechanisms.

3.4 DLT adoption

Healthcare is evolving from traditional to more robust and distributed solutions with people encouraged to provide their personal healthcare data into cloud applications [119, 120]. In parallel to this, DLT is being increasingly adopted for exchanging patient data [121]; building Healthcare Data Gateways [122] preventing scattered information throughout multiple healthcare systems, going towards securing patient records [123] and turning the ledger into an automated access-control manager [124]. Conducted studies [125, 126] show the potential of using DLT not just for finance, but also for SS-IDs, as an integration layer [127], and additionally for key management and privacy [128] in a system that does not need a centralized trusted party.

Due to the recognized hype cycle model [129], there is an over-use of this technology to solve problems that could be easily accomplished in a centralized platform. A unique set of features must be set in place to correctly decide what makes the DLT desirable for our particular problem in relation to other traditional solutions, and answer the question “Do you need a Blockchain?” [130]. Some desirable features for healthcare applications have already been mentioned in other works [131]: decentralized management, immutable audit trail, data provenance, robustness and availability, improved security and privacy. Some of the identified features of DLT are important in order for other properties to emerge. These emerging properties are part of the requirements of our work and are defined in the following set:

- **Censorship-Resistant** assures that everyone can transact in the network on the same terms and can not be excluded from participating. Censorship is a sort of DoS, but differs from the lack of availability because the service availability is intentionally refused. Resistance to this is generally only possible in decentralized or distributed environments and depends on the particular consensus protocol, Hashgraph [132] is one that advocates this feature. This

is more a nice-to-have feature than a critical one, but it is important to build trust.

- **Security** is in general improved with decentralization. Data redundancy and DoS protection is increased, removing any single point of attack. Secrets and encryption keys can be divided into parts and shared across several nodes. This will protect sensitive information against rogue employees, requiring multiple colluded parties to recover the secrets.
- **Trust** is a firm belief in the truth and consistent result of the system. This is maintained by the constant availability, reliability, censorship-resistant and enforcement of specified security policies. High levels of trust are needed for mass adoption and for the system to be accepted as a standard layer of integration.
- **Sovereignty** is an entity's full right and control over themselves or of any other owned data. This is only possible if the personal data is under direct control (in the possession of the data) or in a highly trusted environment. Total sovereignty is probably the only way to guarantee authorization and consent without external influence.
- **Non-Repudiation** is when someone cannot deny an activity or contract (e.g. a debt contract). It is achievable by applying the irreversibility, integrity and auditability properties of the ledger. This is important in order to avoid unreasonable legal actions, e.g. if consent was given, it should not be possible to deny it and file a legal action against the Processor for the improper use of the data.

These features are beneficial when building anonymity, key-management, non-repudiation and self-sovereignty. For instance, by slicing a private key into shared secrets [35], we can improve collusion resistance within key-management. Availability and reliability improve censorship resistance, one of the most important requirements for self-sovereignty. Integrity and immutability make any changes in the registries, auditable, and enhances the requirements for non-repudiation. Ledger nodes can also be used as an overlay network [133] in a similar fashion to the Tor network [134], improving the requirements to anonymity. This concept is already applied in the Verge¹² project.

¹²<https://vergecurrency.com>

However, the general lack of scalability, verifiable identities and governance makes DLT difficult to be accepted in enterprise environments. The fact that mining pools are now more centralised than originally predicted [34], makes governance especially crucial for controlling majority and supermajority attacks. Yet, governance is a characteristic of permissioned ledgers [135], a category that entails a network of identified and privileged mining nodes that can control access. Some recurrent Ethereum incidents¹³ remind us of why permissioned ledgers are essential. When an incident occurs, it gives power to the participants to act and move forward with fixes. This category of ledgers can also improve the overall privacy and anonymity of the data-subject, by restricting public access to data that may lead to statistical attacks for re-identification [136]. However, these type of ledgers also introduces some drawbacks. It requires trust in the participating nodes and would considerably reduce the censorship-resistance capabilities.

3.5 Metadata frameworks

Studying Metadata Frameworks is important to our work in order to understand how to store and exchange information about EHR in an optimized and interoperable way between different healthcare providers. This is a method of achieving interoperability by means of controlled terminologies and precise meaning [137]. As defined in the book "The Semantic Web: Semantics for Data and Services on the Web" [138], Metadata Frameworks are defined in four specifications: **data models**, **semantic constraints**, **serialization formats** and **query languages**. Metadata Frameworks have been used in many contrasting fields [139, 140, 141], but we will focus here on existent standards and practices applied to EHR. An EHR is a digital version of the patient's paper records, making information available instantly and securely to authorised users. Digital formats offer higher efficiency, reduced error rate and better search facilities compared with paper records. EHR may contain the patient's medical history, diagnoses, medications, treatment plans, immunization dates, allergies, radiology images, and laboratory and test results.

¹³<https://medium.freecodecamp.org/a-hacker-stole-31m-of-ether-how-it-happened-and-what-it-means-for-ethereum-9e5dc29e33ce>

3.5.1 Medical standards

Creating interfaces between vendors to share information is a key challenge faced by many healthcare IT departments. It is imperative to understand the existing interoperability standards in order to assess the best methodologies that align with our goals. Standards, industry organizations and communities such as openEHR, IHE, Personal Connected Health Alliance (PCHA) and DICOM, working on the development, promulgation and giving guidance on their use.

OpenEHR¹⁴ is an initiative with the intent to formalize and standardize all of the EHR concepts and models in an interoperable way. In terms of metadata definitions, the openEHR architecture¹⁵ defines two levels of interoperability: the first level is the Reference Model, predefining the minimal schema of an EHR for ensuring data interoperability; and the second one is the Archetype Model, that creates a common knowledge vocabulary for data interpretation, ensuring semantic interoperability. These two levels define the **data models** and **semantic constraints** of the framework. Archetypes allow for clinicians and domain experts to be involved in the design and specification process of EHR. Once an archetype is defined, it is submitted to a Clinical Review Board for validation, ensuring compatibility with existing ones and, at the same time, targeting stability and longevity. In the end, it is open-source and published in the public archetype repository¹⁶. The archetype definition can be expressed in XML or Archetype Definition Language (ADL)¹⁷, an abstract language based on classic Frame Logic [142]. Frame-based systems use entities as frames, their properties as modelling primitives and, types and cardinality to define constraints. These definitions deliver the content structures and specifications that are both understandable by domain specialists and by information system developers. In terms of **serialization formats**, the openEHR has, at least, the Object Data Instance Notation (ODIN) specification¹⁸, with the intent of being a human-readable and a computer-processable data representation syntax that can be edited using a normal text editor. Finally, the Archetype Query Language (AQL)¹⁹ is a declarative **query language** developed specifically for expressing queries used for searching and retrieving the clinical data found in archetypes. The syntax uses some

¹⁴<https://www.openehr.org>

¹⁵http://www.openehr.org/releases/BASE/Release-1.0.3/docs/architecture_overview/architecture_overview.html

¹⁶<https://www.openehr.org/ckm>

¹⁷<http://www.openehr.org/releases/trunk/architecture/am/adl2.pdf>

¹⁸<http://www.openehr.org/releases/BASE/latest/docs/odin/odin.html>

¹⁹<http://www.openehr.org/releases/QUERY/latest/docs/AQL/AQL.html>

clauses from SQL and is projected to be independent of applications, programming languages, system environment or storage models.

IHE²⁰ is a group of healthcare industry representatives with similar objectives as the openEHR community. IHE Profiles²¹ are at the core of the specification, and can describe models using the Cross-enterprise Document Sharing (XDS) integration profile. However, XDS is more concerned with using metadata to facilitate document discovery rather than describing consensual knowledge between clinical affinity domains. This can lead to difficulties in locating a document in different affinity domains. Dogac et al. [143], proposes the use of ontologies to solve this issue. Additionally, those profiles can also describe workflows for day-to-day clinical activities. IHE encourages the integration of established interoperability standards such as HL7 and DICOM into their profiles, allowing seamless integration with existing infrastructure. IHE Profiles extend to a range of domains other than pure clinical activities. For instance, the Import Reconciliation Workflow (IRWF) manages image imports and reconciliation of identifiers to match local values. Furthermore, some profiles already address parts of the GDPR, i.e. mechanisms to record the patient privacy consent using Basic Patient Privacy Consents (BPPC) and Advanced Patient Privacy Consents (APPC), and methods of exchanging claims about an identity across enterprise boundaries using Cross-Enterprise User Assertion (XUA). However, profiles such as Audit Trail and Node Authentication (ATNA), only offer perimeter protection and mutual node authentication, not empowering the patient with control over their own data. As stated in the white paper²², GDPR can be an effective catalyst to extend the reach of IHE Profiles. However, those will require further evaluation to see if they fully comply with the GDPR requirements.

DICOM²³ is a standard defined by the National Electrical Manufacturers Association (NEMA)²⁴ with a universal level of acceptance amongst imaging equipment vendors. It is used to store and transmit medical images and includes simple workflow capabilities, such as the Modality Worklist [144]. Apart from the pixel data, DICOM files have a rich metadata structure with patient demographics and a detailed description of how the image was produced. DICOM was essentially designed for communication between image stations, archives and acquisition equipment, with simple query capabilities and no concerns about privacy and security.

²⁰<http://www.ihe.net>

²¹<https://ihe.net/Profiles>

²²https://www.ihe-europe.net/sites/default/files/GDPR_WEB_00.pdf

²³<https://www.dicomstandard.org>

²⁴<https://www.nema.org>

3.5.2 Adopting semantic web

Studies on the feasibility of the openEHR methodology have been widely carried out [145, 146, 147] showing some positive results in using semantic models. However, there are other general-purpose languages that may serve as alternatives, RDF has been used to form medical ontologies [148, 149] and SPARQL²⁵ has emerged as the standard RDF query language. Ontologies are useful for computers to reason about information, e.g. a computer can alert a physician about a possible drug interaction for a given set of patient prescriptions, improving the quality of care. However, building ontologies is a labour-intensive task with a high learning curve involving domain experts, and query engines are expensive to develop when binding to new database engines. In the meantime, existing data is being leveraged by applying simple heuristics and machine learning. Some argue that semantic web [150] can improve the search capabilities, however, the Google search engine does not need semantics to return meaningful results. Moreover, new techniques are being studied in order to tackle the presence of errors in ontologies [151]. All of these undermine the massive adoption of semantic web. Further studies should be conducted in order to determine if the semantic web will be useful to our approach or if a simple data model definition language, similar to GraphQL²⁶, and a simple query language such as Lucene²⁷ will suffice.

3.6 Related projects

In general, the fundamentals of this work are very close to a set of representative projects, falling into our sharing models: **Science Gateways** (representing the **Anonymous Dataset** sharing model) and **Electronic Health Records** (representing the **Direct Consent** sharing model). Our proposal is a balanced integration of these concepts, where **Identity and Sovereignty** solutions are essential to accomplish this integration. Unification is important because the data sources of the two sharing models are basically the same, but striving for different use-cases and security requirements. In order to perceive how the final solution should be accomplished, detailed studies of these platforms are needed. In general, technology silos exist that solve for each of these particular points, but there is no current solution that gathers all the GDPR and technical requirements in one solid, distributed architecture,

²⁵<https://www.w3.org/TR/sparql11-query>

²⁶<https://graphql.org>

²⁷<https://lucene.apache.org>

with all the associated advantages. In the following sections, we will describe some relevant projects demonstrating each of these aspects. The goal is to show a slightly different approach or feature (in each project description) that may be important to this work.

3.6.1 Electronic health records

EHR were never designed to handle the complexities and business lock-in of the many institutional medical records. Patient data becomes scattered across different organizations, losing connection and easy access to past records. There are many efforts to solve the “Data Portability” challenges and lock-in business models using different approaches, e.g: the openEHR and ISO EN 13606 archetypes [152] with a heavy focus on the persistence of data, HL7-FHIR [153] more on data exchange formats and APIs and HL7 Clinical Document Architecture (CDA) which is primarily focused on documents and document exchange [154]. However, in general, there is a lack of a unified and distributed layer required for some key security aspects of the GDPR: managing the “right to consent”, “right to access”, “right to be forgotten” and “data minimization”.

OpenEMR²⁸ is a free and open-source EHR software. The set of features includes: management of patient demographics and medical records, appointment scheduling using a calendar, clinical decisions and prescriptions, billing and reports. It is certified by the ONC²⁹ and is HIPAA³⁰ compliant, implying a “privacy by design” architecture. The “Data Portability” is sustained by the internal structured records and from the HL7 (for exchanging patient data) and EDI standards (mainly used for billing and invoicing). It also provides a native portal for external access that supports both ACLs and RBAC [155], allowing patients to view their medical history and partially solving the “Right to Access”. Encryption can be used for external documents, but there is no support for digital signatures. The DICOM standard is also not supported natively, nevertheless, it is possible to achieve DICOM interoperability using Mirth Connect³¹.

²⁸<https://www.open-emr.org>

²⁹<https://www.healthit.gov/topic/about-onc>

³⁰<https://www.hhs.gov/hipaa/for-professionals/security/laws-regulations/index.html>

³¹http://www.visolve.com/uploads/resources/vicareplus/OpenEMR_DICOM_Interoperability_using_Mirth.pdf

EtherCIS³² is also free and open-source, but is more of a data repository than a full-fledged EHR software. The PostgreSQL database is extensively used to support the openEHR AOM 1.4³³, where the AQL is used for querying those models. However, the AQL syntax is a synthesis of the SQL language, adding a rigid dependency on relational databases. The support for GraphQL queries is still in an experimental phase, yet, merging the openEHR persistent models with the flexibility of GraphQL [156], seems to be a good alternative to the query system and the “Data Portability” aspect. Authentication services are provided by the Apache Shiro³⁴ framework, but the “Security Features” chapter on the project documentation is mostly unfinished, as stated in their documents: “At the time of this writing, EtherCIS security and confidentiality has been left open since is strongly web site dependent. Nevertheless, authentication and authorization are implemented to support various schemes”.

Chino.io³⁵ is a very recent project primarily concerning in storing and protecting data records in the cloud. It has an explicit focus on the GDPR and HIPAA concerns, designed with “Privacy by Design” in mind and certified for medical sensitive data. Some worth mentioning features from the white paper [157] are: natively implemented Consent Tracking by keeping records of collected consents in a legally valid manner and informing users about the processing of their data; data encryption both in transfer and at rest (at record level) with encryption keys stored in separated environments; immutable audit logs useful for “Breach Notification” and legal disputes; index engine with the ability to search and retrieve records in milliseconds. Chino.io complies with most of our objectives, but there is still room for improvements: it is a full solution, but lacks integration with existing data sources and medical standards; as mentioned in the white paper “accessing the data requires to have in hand master keys, which are set in a special VMs and that only the CTO and CEO have access”, meaning that, encryption keys are not under the data subject control and it is technically possible to access the data without consent; there are no mentions on how to apply the “data minimization” concepts.

Medrec³⁶ is an experimental decentralized record management system, currently under development at MIT Media Lab³⁷. Medrec uses DLT for authentication, permission management, confidentiality, accountability and data sharing [158]. Patient data is not stored directly in the ledger, rather a signature of the data and metadata

³²<http://ethercis.org>

³³<http://www.openehr.org/releases/AM/latest/docs/AOM1.4/AOM1.4.html>

³⁴<http://shiro.apache.org>

³⁵<https://www.chino.io>

³⁶<https://medrec.media.mit.edu>

³⁷<https://www.media.mit.edu>

is encoded, allowing access control to the medical records; the signature assures that an unaltered copy of the record is obtained. Medrec is built on the principle of interoperability, using the Ethereum blockchain³⁸ and Smart Contracts [159] as an integration layer for existing data providers, with the flexibility to support open standards for health data exchange. Off-chain requests are handled by the Database Gatekeeper, which implements an access interface to the patient local database. However, there are limitations on the realm of privacy, pseudonymous identities are easily traceable in the public ledger, allowing for data forensics. Important features such as “data minimization”, “non-repudiation” and “key-management”, are not properly addressed in the architecture.

3.6.2 Science gateways

Science Gateways are a community-developed set of tools to access shared data, software, computing services, instruments, educational materials and other resources for the needs of a specific scientific community. These platforms are available in a broad range of scientific knowledge, from food security [160] to astrophysics [161]. These gateways deliver the datasets and requirements to create reproducible science [162], by offering sharing possibilities within a community. But also face serious challenges, in relation to: privacy, security, data isolation and the existence of multiple data formats. Our goal is not to replace the existing Science Gateways, but, to integrate them into a secure and unified system by leveraging standards and APIs, and by applying the necessary layers of regulation to the already existing security recommendations [163]. In this way, researchers can focus on their scientific goals and less on assembling the appropriate infrastructure. Although a Science Gateway software normally has job submissions and workflow tools, without discarding future developments in these fields, we are essentially interested in the data sharing, privacy and security methodologies applied to medical data. Following is a representative list of Science Gateway tools.

XNAT³⁹ [164] is a dedicated open source project for research in medical imaging. The core features include: project management, image importing, archiving, organising, searching, processing and sharing. These features can be extended into other systems similar to XNAT-TraiT [165]. XNAT has been present in datasets provided by the Open Access Series of Imaging Studies (OASIS) [166, 167] and other databases [168]. Access control is managed by project roles, and by default, it defines the following:

³⁸<https://www.ethereum.org>

³⁹<https://www.xnat.org>

Owners (having all the permissions on the project), **Members** (cannot modify the project users and data types) and **Collaborators** (cannot insert or modify data, but can download and use the project data). However due to the age of the software, it has not been designed with the GDPR in mind, subjects are anonymized and linked throughout the database using a Generic ID, serving only the anonymous dataset sharing model. Using “Pseudonymity” and direct consent will require many changes to the underlying software to be GDPR compliant.

MONTRA is a catalogue system that is part of the EMIF project⁴⁰ and can be considered a modern vision of the XNAT concepts. Besides, it is integrated with TASK [169], an in-house workflow engine. The catalogue intends to comprise different vertical projects and databases through the creation of communities, such as EMIF-Electronic Health Record Data (EMIF-EHR) and EMIF-Alzheimer’s Disease (EMIF-AD), fulfilling distinct research requirements. The major goal is to integrate heterogeneous biomedical databases into a unified platform and User Interface (UI), with the main ability to build such web platforms almost on the fly. Access control is achieved with a regular RBAC. Although the project has a UI plugin-based architecture, capable of linking with other data sources, the database core is limited to questionnaires. MONTRA project also cites projects such as “Bridge-To-Data”⁴¹ and “Healthcare Cost and Utilization Project”⁴², as the best-known initiatives for similar purposes [170].

Galaxy Project [171] consists of the **Galaxy Software Framework**, where the goal is to develop and maintain a system that enables researchers without informatics expertise to perform computational analyses through the Web. **Public Galaxy Service** provides CPU and disk space for datasets and job submissions from a set of public servers⁴³ from contributing institutions. IRMACS⁴⁴ deployed, in 2012, the first instance of the software that, enabled researchers to perform complex genomic analyses and visualizations. It was a relevant achievement due to the aggregated nature of multiple decentralized services, demonstrating the viability of such architecture for Cloud and Grid computing [172]. However, the service is no longer in operation.

⁴⁰<http://www.emif.eu>

⁴¹<https://www.bridgetodata.org>

⁴²<https://www.ahrq.gov/research/data/hcup/index.html>

⁴³<https://galaxyproject.org/public-galaxy-servers>

⁴⁴<http://www.irmacs.sfu.ca/infrastructure/gateways>

3.6.3 Identity and sovereignty

In this topic, most of the projects are very recent contributions to the field of Identity and Sovereignty Management. Most of these projects are part of the Decentralized Identity Foundation⁴⁵, a consortium of companies building the ecosystem for decentralized identity management. However, the information is normally provided in white papers that do not always have the relevant information for a correct third party evaluation, and as so, it is difficult to accept them as valid contributions to the scientific knowledge. Nevertheless, those projects added relevant value and different perspectives to the state-of-the-art.

Sovrin⁴⁶ implements “Privacy by Design” on a public permissioned ledger. No private data is stored on the ledger, even in encrypted form. All sensitive information is protected in a parallel network of distributed private agents separated from the public ledger. However, its white paper [48] does not state that their solution will provide “Integration” with existing databases or that the Anonymous Dataset sharing model is possible. It uses the new W3C standard DID⁴⁷ in order to implement the “Sovereignty”, “Data Minimization” and “Pseudonymity” requirements. DIDs are the first globally unique, verifiable identifiers that require no registration authority. However, DIDs “Key-Management” and “Non-Repudiation” features seems only vague concepts. It specifies operations to revoke and recover keys from a quorum of trusted parties but does not provide any specific solution to do it. The Sovrin project only shows intentions to apply Shamir’s secret sharing [35] as a possible method. The reported “Non-Repudiation” feature is not strongly specified over the “Right to Consent” framework, implying that it is not possible to identify if the data subject does not want to give access to a set of data or if there is no information available at all for that set.

Civic⁴⁸ aims to tackle the problem of consumer identity theft and online identity fraud by using MFA. It offers more than a technical solution with monitoring, alerts and identity theft insurance services. It is an excellent example on how the “Breach Notification” requirement should be implemented. In the project white paper [59], it is stated that the ecosystem incentivizes the participation of trustworthy Validators; these are financial institutions, government and other utility companies. Validators will be able to verify the identity of individuals and organizations. “Data Minimization”

⁴⁵<http://identity.foundation>

⁴⁶<https://sovrin.org>

⁴⁷<https://w3c-ccg.github.io/did-spec>

⁴⁸<https://www.civic.com>

is implemented using Merkle Tree techniques [173] composed by the hashes of the Validators' attestations. Parts of the tree and the corresponding content can be selectively revealed with integrity assurance. The presence of the hashes proves the existence of unrevealed information, providing "Non-Repudiation" features. However, this methodology requires that all the attestations' content be off-ledger, in order to protect the hashes against attacks. The identity and attestations are encrypted and stored in the Civic App. Data is revoked on the blockchain by the authenticating authority, although it is not completely explained how this authority works. In practice the Merkle Tree solution cannot automatically disclose revocations or any subsequent alterations, it needs the user's consent every time changes are inserted into the ledger. This makes the "Non-Repudiation" feature somehow limited because parts of an important block of information can be hidden from external readers.

uPort⁴⁹ is built on top of Ethereum ⁵⁰ (a public permissionless ledger) where the identifier is the address of a smart contract. "Key-Management" is addressed by the Controller and Proxy contracts. The Proxy preserves a fixed identifier while the Controller contains a recovery address that can change the Proxy ownership. The recovery address can actually be a Quorum contract, providing in this way a distributed model to recover keys; and is flexible enough to implement other key recovering methods. It is not clear how "Data Minimization" and "Pseudonymity" can be done, or if it is possible at all. The "Integration" with external databases and the dataset sharing model may be easily supported with Ethereum smart contracts, but this is not specified directly by the project. uPort is a simple identity framework, no support for claims, attestations, "Non-Repudiation", or "Right to Consent" features were found. Although the Ethereum network is working on the Ethereum Claims Registry (ECR) ⁵¹, the issue was still open at the time of this writing. Also, the dedicated Ethereum smart contracts are not compliant with the "Data Portability" requirement. We cannot accept these contracts as "Sovereign" because they are not self-contained data blocks, and cannot be used without the Ethereum network.

⁴⁹<https://www.uport.me>

⁵⁰<https://www.ethereum.org>

⁵¹<https://github.com/ethereum/EIPs/issues/780>

Hypothesis

The motivation and goals pursued by this research and its starting hypothesis are defined in this chapter.

4.1 Motivation

Any given information system in the context of healthcare faces two major conflicts related to the patient's data privacy right:

- The importance of the availability of massive datasets when research is leaning towards machine learning techniques [10], as well as identifying connections between those datasets.
- Restricting access to a patient EHR will likely reduce the quality of the delivered care. A study conducted by Tierney et al [174] shows that 63% of the clinicians point out that issue.

The main motivation for this thesis is to study and propose a technological solution to put current datasets under the patient's control, aggregate them in a unique identifier and make them available for **primary** and **secondary** usages (research and diagnosis). From what was already stated we identified several important aspects that drove decisions for this thesis:

- Protecting healthcare data is mandatory under the GDPR context. Any proposed solution must be restrained in the context of such legality.
- Aggregation of EHR data for each patient is important for diagnosis.
- It is important to connect conglomerates of anonymous and pseudonymous data in one identifier that represents the data-subject.

- Tracing back all aggregated data to the correct pseudonym is important for future research.
- Break-the-glass access policies for emergencies are extremely important in a medical context.
- Availability and data durability are also extremely important in a medical context.

4.2 Research goals

The main research goal of this thesis is to provide a foundation to **connect** different data curators in a interoperable and secure way; **unify** all the existing scattered datasets under the same UPI; to safely **store** and **share** medical records between those data curators. The main purpose is to provide a unified model for primary and secondary uses of healthcare data. The model accomplishes two distinct use-cases for the same data: the need for physicians, nurses, or other authorised personnel to directly access medical records; and the usefulness of massive anonymised datasets for research and training, being for machine learning algorithms or step-by-step learning protocols [175].

This thesis will research through existing anonymisation, pseudonymisation and encryption techniques, SS-IDs [176], Distributed File System (DFS) and DLT [177] ideas and concepts. We will explore new methods of encryption and key-management to achieve implicit consent (break-the-glass) and identity unification. Explore distributed systems and threshold cryptography to accomplish data availability and durability. This thesis will be focused on medical imaging use-cases, but the presented techniques should be extendable to general EHR.

4.3 Hypothesis

This Doctoral Thesis researches the following hypothesis:

Distributed Ledger Technologies and Self-Sovereign Identities are natural symbioses to solve GDPR challenges in the context of healthcare. Namely, providing means for direct

access to the patients' records, for diagnosis and therapeutic processes, and bulk access to anonymous datasets or aggregated data for research purposes.

Proposed System

This section provides an overview of the proposed architecture, compliant with GDPR rules, in the demanding context of medical imaging records. It provides an integrated solution for several problems like anonymity/pseudonymity, encryption, key management, break-the-glass and shareable records.

Ideally, the architecture and threat modelling takes place in the inception and design phase of a project. The goal is to identify the actors, uses-cases and possible attack vectors. What is expected to be secure and what secure limitations exist. This chapter defines the system and threat model that is applicable to all other chapters of this thesis.

5.1 Architecture overview

A reasonable description of an overall architecture is depicted in Figure 5.1. In *a)*, a unique piece of public information (the identifier or *id*) is used to derive a pseudonym π . The pseudonym is used as the identifier for DICOM files and stored via a standardized PACS interface *b)*. DICOM files are sliced into meta-data and data blocks and stored in DLT *c.1)* and DFS *c.2)*, respectively, encrypted and anonymised. The same records are accessible via PACS with implicit *d.1)* and explicit *d.2)* consent.

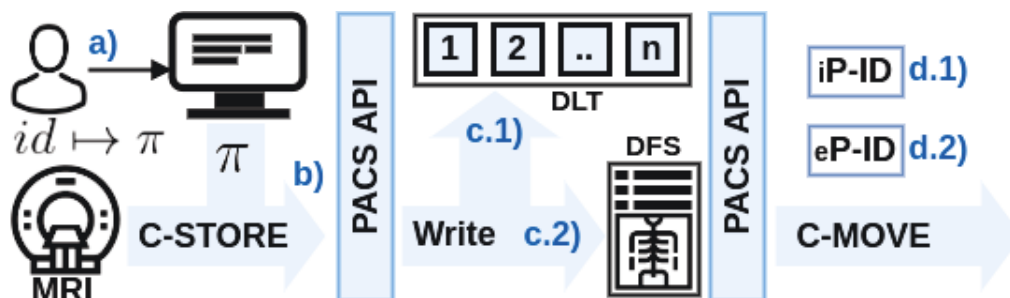


Figure 5.1.: Overall architecture the proposed thesis. The storage process *a)*, *b)*, *c)* is nonetheless standardized as a DICOM service. The goal is to retrieve anonymised records *d)* from a federation of PACS nodes, supporting both implicit *d.1)* and explicit *d.2)* consent.

The **identity** structure (a primitive for other components) and **access** control mechanisms are part of Chapter 6. Generation of pseudonyms π , **Explicit Pseudonym Identifier Derivation (eP-ID)** and **Implicit Pseudonym Identifier Derivation (iP-ID)** consent protocols are part of Chapter 7. Storage of **encrypted** records is part of Chapter 8. Finally, the **anonymous token** protocol, required to connect a pseudonym to a real identity (patient), is defined in Chapter 9.

Chapter 9 has an extension to this system and threat model. Although it has a compatible model, it uses more complex cryptographic tools (pairing-based cryptography) that require further context. Introducing this model here would confuse the reader for the other chapters.

5.2 System components

All components, structures and protocols for this thesis are depicted in Figure 5.2.

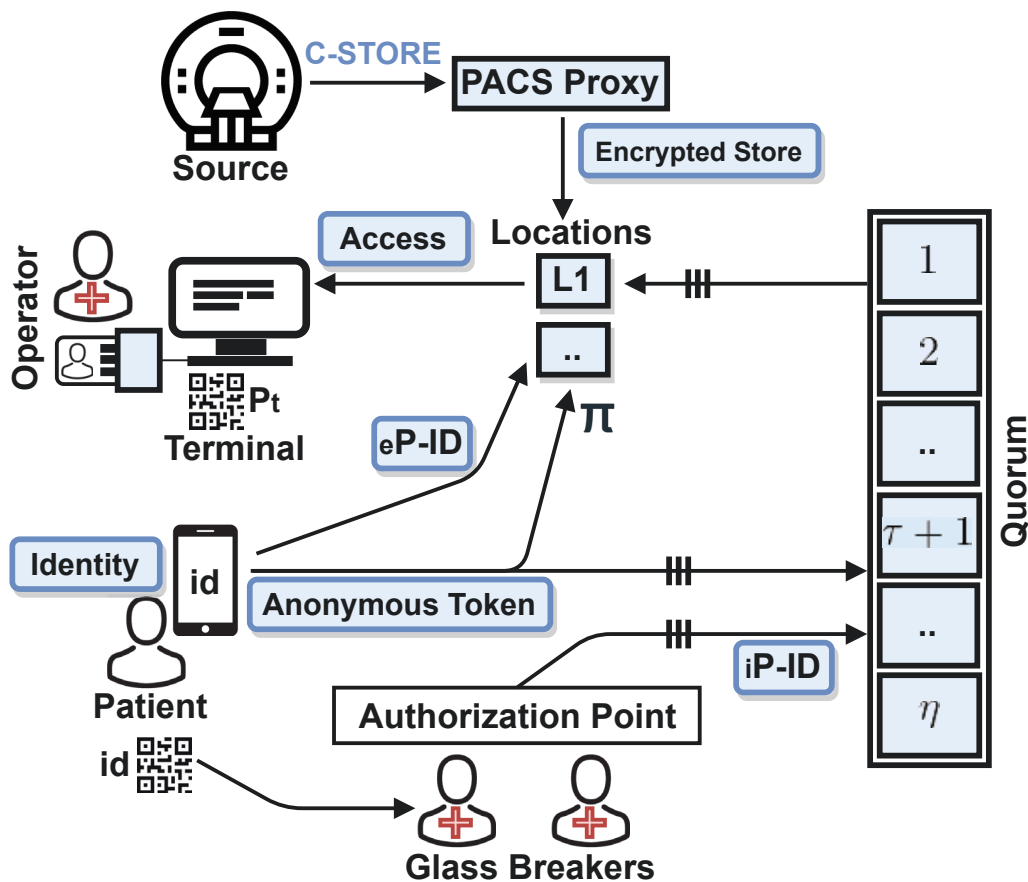


Figure 5.2.: This figure identifies all components and protocols that are proposed in this thesis.

The **Operator** or physician is the interested party in reading the patient's EHR. The operator has a smart card with the respective key pair $o \times G = P_o$, used for signatures.

The **Terminal** is the endpoint that receives patient's records that are then displayed to the operator. A key pair $t \times G = P_t$ is defined for this point, used for signatures.

Location are isolated storage points for anonymised EHR. Locations are data servers for terminals. Each location has a fixed address L_{addr} (i.e. an HTTP URL) and their own pair of keys $l \times G = L$, where $L \mapsto L_{addr}$ is registered in the quorum. L should be also certified by the quorum or manually installed in each terminal.

The **Patient** or data-subject is the legal owner of EHR. A key pair $s \times G = P_s$ is defined for the subject. The private key is stored in the subject's mobile device and is used to initiate the explicit consent protocol (eP-ID).

Glass Breakers are groups of physicians that provide authorization to read patients' records without explicit consent. A key pair $a \times G = P_a$ is defined for each glass-breaker. The private key can be stored in a mobile device or smart card (the device) and is used to initiate the implicit consent protocol (iP-ID).

The **Authorization Point** is the software or device that collects consent from authorised parties (glass breakers), the subject's public information id and starts the implicit consent protocol.

A **Source** is a system that is authorized to write on patient's EHR, generally in encrypted form. The source key can be the same as the subject P_s , or other authorized key.

The **Quorum** provides distributed threshold keys and protocols for deriving pseudonyms and encryption keys. These form a federation of $\eta \geq \tau + 1$ parties in a pre-configured (τ, η) -threshold, with corresponding key pairs $n_i \times G = N_i$, where $i \in [1, \eta] \cap \mathbb{Z}$ and pre-configured parameters:

- A defined federation of η parties in total, resilient to τ Byzantine failures.
- A set of secret shares $\{(y_i, e_i) \in \mathbb{F}_p\}$ derived from a dealerless share distribution protocol [178, 179, 180].

- A pair of master private keys (y, e) derived from Equation (2.9), where $\mathcal{L}^i(y_i) = y$ and $\mathcal{L}^i(e_i) = e$.
- Corresponding public keys from $y \times G = Y$ and $e \times G = P_e$. These are actually derived from $\mathcal{L}^i(y_i \times G) = Y$ and $\mathcal{L}^i(e_i \times G) = P_e$, and stored as public parameters for the quorum.

5.3 Threat model

We assume the random oracle model, the hardness of the DLP and CDH¹ for elliptic curves. All elliptic curve points and scalars of the form $s \times P = P_s$ are constrained within $P_s \neq G$ and $P_s \neq O$ assuring that $s \neq 1$ and $s \neq 0$.

Key properties. It is assumed that the terminal public key P_t is known and trusted by the authorization point. This is a reasonable assumption since both systems are normally in the same affinity domain. P_a and P_s public keys are also trusted and managed by the quorum via a distributed storage.

Protection of the secret keys s and a are under the responsibility of the patient and healthcare organisation, respectively (not part of the protocol responsibilities). A compromised s key corresponds to a single patient pseudonym compromise, however, the a key corresponds to a compromise of the explicit consent protocol.

Distributed model. It is assumed that the adversary is able to compromise a maximum of τ shares and can run a rogue client (without a valid P_a or P_s key), running any kind of procedures and tampering with the protocol inputs. Attacks on $\tau + 1$ secret shares are considered catastrophic failures. It is assumed that honest parties verify inputs, such that if the elliptic points are in the correct curve and if scalars are in the correct range, avoiding an entire class of issues [181].

We assume a quorum with a distributed public database that can attest for these keys as belonging to the quorum, and a public channel or storage providing a way to publish information between parties. Such requirement could be provided by a DLT. We presume that the underlying distributed storage technology is safe under

¹<http://www.ecrypt.eu.org/ecrypt2/documents/D.MAYA.6.pdf>

the same (η, τ) -threshold security. This is a normal assumption for any Byzantine distributed ledger for a $\eta > 3\tau + 1$ configuration.

We assume that locations are honest-but-curious and will not maliciously delete patients' records.

Identity Management

This chapter presents the data structure of a master digital identity and respective anonymous profiles. The relation between the master digital identity and any of the profiles is protected with an identity key pair. Such relation can only be disclosed with proper consent from the profile owner (the identity). The master digital identity serves as a foundation for other data structures and methods through this thesis.

Implementing pseudonymity, key-management, data minimisation, authentication and authorization as isolated features is trivial. However, integrating all of them in one consistent architecture has several challenges to tackle. This chapter presents the foundation and data structures to represent SS-IDs and to handle those features in a consolidated architecture.

The master digital identity is identified by a UDI related to a cryptographic key pair. The UDI is just the subject's top identifier of a tree data structure. Pseudonymity and data minimisation is established using anonymous profiles, showing different views of the same identity. Profiles are sliced views of personal information connected to the UDI. The correlation between the UDI and any of the profiles is protected with anchors (elliptic curve points) and multiparty computations. Such correlation can only be disclosed with proper consent from the owner of the identity.

This master identity and profile structure fits directly into the proposed architecture (Chapter 5), where the structure of the master identity can be managed by the federated Quorum, while profiles can be managed by Locations.

6.1 Master digital identity

The master digital identity needs an associated asymmetric key pair $s \times G = P_s$ to authenticate with third-parties and sign data. Only one key pair is active at any given time; however, these keys can be lost or compromised. In order to maintain the chain of custody when such keys are lost any subsequently renewed keys should be

traceable and verifiable to the same identifier. The identity owner should also do this without depending on any centralised external service. The data-subject should be sovereign when managing his identity. These requirements are reflected in the identity structure depicted in Figure 6.1.

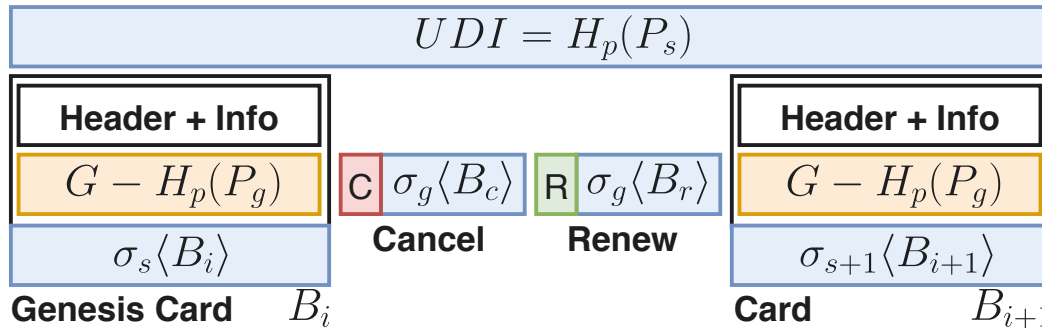


Figure 6.1.: An identity structure identified by the UDI evolving from the P_s key to P_{s+1} . The process is done via cancellation of the old key and renovation to a new one. P_s gets invalid as soon as the cancel block is registered. The last card in the chain defines the active card and the current asymmetric key pair in use. A set of T-Link groups (G) containing a hash of a public-key P_g are used in the evolution process.

The most important feature of the identity's structure is to provide key-management by tracking the evolution of the active public-key P_s , maintaining a unique identifier throughout this process. The fundamental identity blocks (card, cancel, renew) contains data for traceability and verifiability. Several cards are linked together via the evolution process, each one corresponding to a new pair of keys $s \times G = P_s$. The identity is identified by the UDI, corresponding to the hash of the first generated public-key $H_p(P_s)$ (in the genesis card). All cards contain the content signature and also any public information that the identity owner wants to make public. The header defines the chosen cryptographic functions, elliptic curves, signatures, hashes and key derivation functions; or any other necessary public parameters. The genesis card has a unique **name** field that serves as an alias, for human-readable identification. The master identity does not provide native anonymity features. Although the UDI says nothing about the identity, the amount of anonymity depends on the registered information.

6.1.1 Trusted link groups

Trusted-Link groups (T-Link groups), identified as G blocks in Figure 6.1 are pre-defined when the card is initially created. A pair of keys $g \times G = P_g$ is selected for the commitment scheme [182], where $H_p(P_g)$ hides the public key. P_g is the key that validates σ_g signatures from evolution blocks, such as **cancel** and **renew**. Different

groups and configurations can be combined in multiple scenarios. For instance, a group can be used to evolve the identity key and give full control to a legal entity in a litigious process. Configurations can define if the group is fixed and cannot be removed from the identity; if it is only used for the cancellation procedure, renovation or both; if it is authoritative over others, locking the possibility of changing the T-Link groups only to this set. In a sense, it is even possible to disable the evolution process if all the T-Link groups are removed, making this a more general approach for an identity management system. Transferring ownership is also possible and useful in some contexts, by evolving the identity to a card where the T-Link groups are not controlled by the same data-subject anymore. Many of these options can be considered in future specifications, and not all are elaborated on here.

The g private key is normally used to recover the identity's ownership. For instance, in case s is lost or compromised. For this purpose, it is logical that the group key g is not stored in the same place as the identity key s . There are multiple options to derive g . From a strong password and a key derivation function $H_p(\text{password}||UDI) = g$; or even from a multi-party computation $H_p(\mathcal{L}^i(y_i \times K)) = g$, where the elliptic curve point K may serve as the password. The main goal is that the derivation of g is not specifically bound to a protocol.

6.1.2 Card evolution

Details of the card evolution structure are depicted in Figure 6.2. The evolution forms an immutable chain link of hash codes and signatures, effectively transforming the block sequence into a verifiable chain. Both **cancel** and **renew** actions have to be signed with the corresponding private-key g of the T-Link group that is evolving the identity.

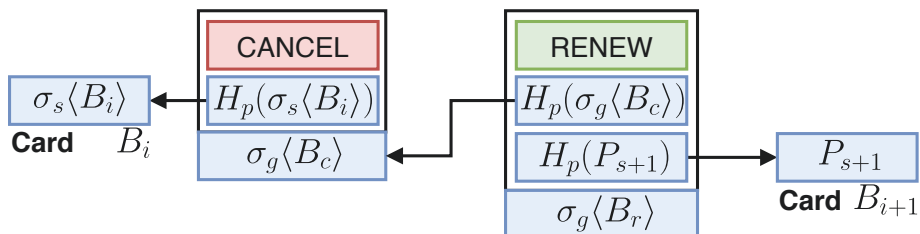


Figure 6.2.: A detailed view of the evolution structure, the respective evolve blocks and cards linked in a sequence of digital signatures. Additionally, the renew block needs the $H_p(P_{s+1})$ reference to next valid card. Any card with a different P_{s+1} key is invalid.

The cancellation requires a reference to the signature of the last active card, defined as $H_p(\sigma_s\langle B_i \rangle)$. When a correctly signed and verified cancellation is registered, the cancel status is established, and the identity evolves to the inactive state.

The renovation contains the hash of the new public key for the identity, fixing P_{s+1} as the only acceptable key for the next card block, avoiding any attacker to reuse the renew block on a different card. The renew block can be inserted into the new card instead of being a separated block. However, the standalone renew block can be used to commit the P_{s+1} key for the next block without such block being present in the database (defined as the next candidate block). This can be useful in situations where the renew and candidate blocks are generated in different machines. In this way, compromised machines would not be able to compromise the new identity key and group key at the same time. The renew block can also exist without a previous cancel action. When a correctly signed and verified renovation is registered, the active status is established, and the identity evolves to the next public key P_{s+1} .

An alternative evolution path is to consider $P_s = P_g$ and self-sign the evolution blocks. In that regard, the card itself is considered an implicit T-Link group. However, the self-signed evolution cannot change T-Link groups for the next card, preventing an identity takeover from a compromised s key. This simplification is useful to upgrade cryptographic functions without requiring the use of T-Links. The group can also emit a **close** block to permanently close the identity. No more linked cards would be accepted after this procedure. This is the last resource for a compromised identity.

6.1.3 Registry

Identity owners can emit registries with information that can be useful for different use-cases, such as claims, attestations or other general information. All records follow a tabular format in the *info* section, where interpretation of the contained fields is specified for each domain and type. Domains can be reserved for specific schemes and specifications. Records also have a valid content signature, using the current identity key. Note that, the index of the key can be used to identify the exact key used since the public key P_s is already stored in the identity card.

Figure 6.3 shows both types of records that can be inserted in the database (*SET* and *DEL*). The *SET* and *DEL* flags follows a substitution and deletion rule. This rule offers a standardised method to precisely define the actual state of the database

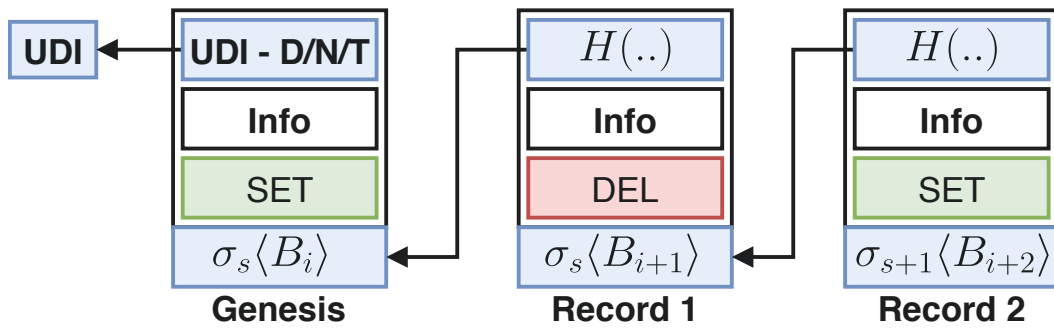


Figure 6.3.: An example of an identity registry. Contains new entries and substitutions (SET), and deletions (DEL). The D/N/T setting and association to the UDI are registered in the genesis block. Record blocks form a linked list by referencing the signature of the previous one. The info structure contains the actual data of the block.

fields, overriding previous field states. This is not only important to build a resume of the states but also makes it easier to implement state-tree-pruning¹ methods in future developments, by providing drop-out points in the chain. The genesis record has a reference to the UDI ownership and a reference to a schema defined by the **Domain** and **Type**, respectively. The domain is a string format in the form of a namespace, i.e. “sector.organization.pt”, while the type indicates the specified schema structure of the registry in that domain. A registry instance is identified by the **Name**. Any combination of (UDI, *Domain*, *Name*) is unique and must be assured by ledger constraints.

6.1.4 Security analysis

We assume the immutability of the ledger, as also the security of the private keys. We assume that the genesis card registration is done under secure conditions. The goal of an adversary is to break the chain of ownership of identity cards or registries.

Theorem 1 *The identity structure preserves traceable ownership in any direction within key evolutions.*

Proof: Due to the commitment scheme, $H_p(P_g)$ on T-Link groups, only a restricted set of public keys are allowed to evolve the identity. The chain of hash codes and signatures (on cancel and renew blocks) forms an immutable and tamper-proof linked list that traces backward ownership. The forward commitment scheme $H_p(P_{s+1})$ on

¹<https://blog.ethereum.org/2015/06/26/state-tree-pruning>

the renew block produces a double link, forcing the next key and preventing the formation of any chain forks.

Theorem 2 *A registry forms a tamper-proof linked list that is associated with only one identity and ownership.*

Proof: The chain of hash codes ($H(\dots)$) and signatures form the tamper-proof linked list. The genesis record ties the registry to the UDI. Signatures $\sigma_{s+k}\langle B_{i+l} \rangle$ use the current card key, and from Theorem 1, those keys have a unique ownership.

Public Key Exposure. There is a rising concern about EC cryptography related to quantum computing and compromised signature algorithms. Let us suppose a used EC suffered a flaw that allowed somebody to derive a private key from a public key in a short period. This is not entirely unfeasible, and current cryptographic primitives may need to be replaced in the quantum computer generation [183]. The situation is less critical for hash functions and symmetric cyphers [184]. Bitcoin applies SHA256 and RIPEMD-160 hashes on the public key to generate the wallet address as a fail-safe backup. The public key is only revealed when some coins are spent. In the emergency of a broken EC, users can move their coins to a fresh address and use the backup protection. This is why the recommendation is to “never reuse the same address”. The same concept is applied here to the T-Link groups; a hash is applied over the public key of the group. P_g is only revealed when the cancellation process is executed. The new group must have a different key pair. We apply this concept in all structures whenever possible. However, this protection cannot be replicated for the identity keys P_s because they are necessary to validate the card signature at any moment, and consequently, to validate the card chain. The identity card chain is public and may be replicated to different systems. The chain should be represented in a portable format and verifiable without requiring any additional contact with the UDI owner. For instance, previous knowledge of the active P_s is required if a private notification is to be sent to the UDI; otherwise, a key-exchange negotiation would be previously necessary. Nonetheless, securing the group key provides a mechanism to evolve the identity to a safer cryptographic primitive in case the EC has been compromised. Furthermore, current research on hash-based signatures [185] can provide alternative options to post-quantum cryptography.

Delegation. The delegation of full control (read/write) is sometimes needed. For instance, for a father to control the digital identity of a minor child. A delegated

account is managed in the same way as any other, but it should be noticeable by external identities. These accounts have additional requirements: the real owner should be able to take full control of the identity at any moment and, all account activities should be notified to the owner (if possible). The procedure to take full control can be provided by government entities. One can configure an authoritative master T-Link group (at the registration step) controlled by the government, and capable of removing the parental control of the identity.

6.1.5 Results

The integrity of the structures was tested and is available at [github](#)². These results are in the form of unit tests to check the most important structure constraints. To run those tests execute “cargo test”. Identity, cards and evolutions are checked in the “identity” tests. Streams and chains are checked in the “stream” tests. This demonstration does not prove that the proposed cryptographic scheme is safe. The intention of these checks is to prove that the proposed structures and constraints are possible in a real scenario.

6.2 Anonymous profiles

Why are identity profiles important? Let us assume that the same identity is used for a private health profile and a public curriculum vitae profile. The UDI will link both profiles; however, if those were under a single monolithic block of information, it would not be possible to have different privacy settings for each profile. Even if the health registries were anonymous, when authorising a third-party to access those registries the process will inevitably disclose all the data. Not doing so can imply a failure in data integrity, since interrelated data may have references to each other that may not be identifiable by computers. Data isolation can be provided with a multi-tenant architecture; however, many of these data silos are under the control of the database owner. Our intention is to achieve such similar data isolation but also maintain data ownership under the control of the data-subject. Here, we distinguish two extremes of data privacy. One can be useful as public information, while the other has sensible healthcare information that cannot be publicly available. For any of those profiles to handle the authorisation procedure without disclosing too much information, they must be managed as if they were independent identities. These

²<https://github.com/shumy-tools/raiap-test>

independent blocks can only be linked to the original identity through authorised disclosure. This principle provides just the minimum amount of data (data minimisation) for the usage context, and it also provides a method to reduce the impacts of inferring sensitive information [186, 187] between profiles. This means, that even if a profile is compromised with an inference attack, other profiles are isolated from this security breach. Moreover, inferring multi-profile linkage based on their content should be difficult since their information generally applies to a completely different domain.

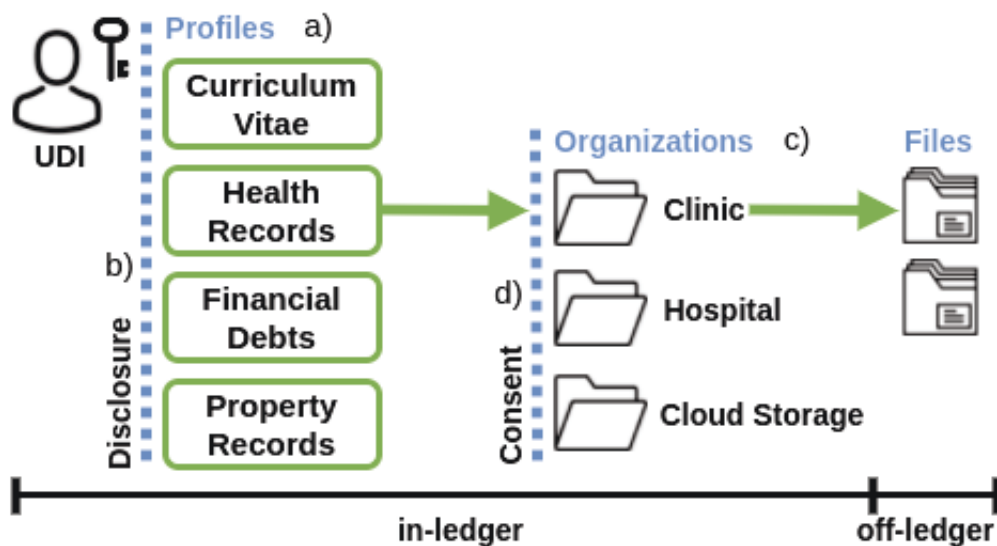


Figure 6.4.: The figure shows the integration of the identity and associated profiles *a)* (in-ledger) with existing infrastructure (off-ledger). The in-ledger structure has two levels of isolation: the disclosure *b)* reveals the pseudonymous profile related to the data-subject UDI and, consent *d)* provides access to the profile files stored in external organisations *c)*.

As depicted in Figure 6.4, profiles *a)* aggregate pseudonymised data structures linked to the same UDI. The disclosure *b)* of a profile as a whole will make all of the respective registries visible and ensure data integrity. Consequently, this visibility associated with the ledger immutability will also provide the technical requirements for non-repudiation. When a profile is disclosed to interested parties, these can identify external systems and file addresses *c)* where the data resides. This makes the ledger an integration/authorisation layer for foreign data curators, such as cloud providers. A final consent from the data-subject *d)* may be required to access the external content. The master identity is normally located at the federated Quorum, while profiles can be set and spread into multiple Locations, as defined in Figure 5.2.

6.2.1 Anchors

A profile requires the presence of an **anchor** in the public registries of the master identity. The purpose of the anchor is to have minimal information for a one-to-one non-repudiable link to an existing anonymous profile without actually revealing it. The overall schematic is represented in Figure 6.5.

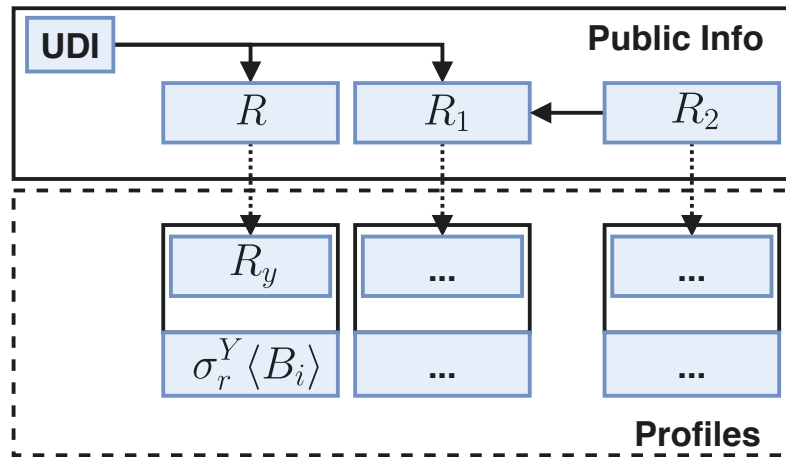


Figure 6.5.: The UDI and associated profile anchors (R , R_1 and R_2). For each anchor R there is an associated anonymous profile R_y that can only be disclosed with the authorization of the identity owner.

The identity owner selects a random key pair $r \times G = R$, with $r \in \mathbb{F}_p^*$, when registering an anchor in the public registries of the master identity. A profile can have multiple anchors R_i for the purpose of renewing the r_i key. A set of anchors $\langle R_0, \dots, R_i \rangle$ connects to a set of profile identifiers $\langle R_{y/0}, \dots, R_{y/i} \rangle$. For simplification we will use a profile with a single anchor (a single $R \mapsto R_y$ mapping), where $r \times Y = R_y$ is connected to multiple blocks of information B_i . Each block B_i must be signed with the r private key using Y as the base point, resulting in $\sigma_r^Y \langle B_i \rangle$. The signature can be verified using Y (the base point) and R_y (the public key) as defined in Section 2.4.1.

Profile disclosure. Note that, the $R \mapsto R_y$ correlation can be disclosed with one of the two equivalent paths; $R_y = r \times Y = \mathcal{L}^i(y_i \times R)$. The r secret is mainly used to sign profile records; however, the multi-party computation $\mathcal{L}^i(y_i \times R)$ (as defined in Equation (2.9)) is useful to disclose the $R \mapsto R_y$ connection without involving a direct action from the data-subject.

6.2.2 Security Analysis

The profiles' architecture is an overview of what is further presented in subsequent chapters of this thesis. Details of the method and security analysis will be given in Chapter 7. However, even without a detailed explanation of the method we can still conclude the following:

Theorem 3 *The anchor assures a unique path from the identity to the respective anonymous profile.*

Proof: The $\mathcal{L}^i(y_i \times R) = R_y$ defines a path from the identity to the profile that cannot be used by other identities due to the strong link between the UDI and the anchor R . If we assume that r and y_i secrets are safely guarded; any system that collects a set of $\tau + 1$ shares $(y_i \times R)$ can derive the correct profile R_y . Also, due to the DLP and from the equivalence of $\mathcal{L}^i(y_i \times R) = y \times R$, will be very hard for any node in the Quorum (without communicating with each other) to produce a share that could result in an existing R_y . However, this method still allows for the rushing adversary attack described in Section 2.4.5.

Theorem 4 *The profile anonymity is preserved. There is no direct evidence from public information that connects the identity to the profile.*

Proof: The $\mathcal{L}^i(y_i \times R) = y \times R$ computation is considered irreversible due to the DLP. r or a set of $\tau + 1$ shares y_i are required to obtain the $R \mapsto R_y$ correlation. We discard any correlations that can be obtained from side channels attacks, such as timing requests that reach Quorum nodes and the respective responses from Locations.

6.3 Authorisation

The goal of this section is to provide a grant mechanism for third-parties to access the identity records, in a manner that is safe and compliant with GDPR principles. Furthermore, our proposal will explore the described demanding scenario of medical imaging archives.

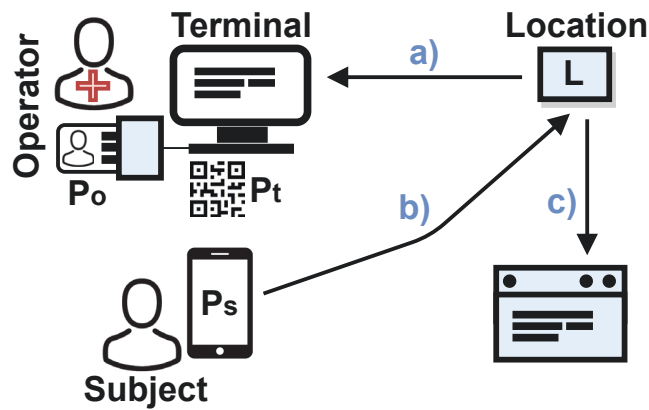


Figure 6.6.: The terminal/operator is the processor for the personal data, where the location is the controller. The data-subject is able to provide explicit consent for any data concerning him in the controller by using his mobile phone.

Depicted in Figure 6.6 is the overall schematics for our use-case. The purpose of such a scenario is to provide a method of transferring medical data from the controller (location) to an authorised operator and terminal *a)*, providing that there is explicit consent from the data-subject *b)*. This preliminary work is focused on the protocol between the operator/terminal and the location *a)*. The advantages of the proposed method are:

- Both the terminal and operator can be **physically identified**, increasing the security impression that the data is being delivered to the correct endpoint. Such information is provided in the consent protocol and can be visually confirmed by the subject on his device.
- The operator can reuse existing **hardware protection** for the authorisation keys and certification procedures, such as the one that already exists for the Citizen's Card.
- The data controller (location) is able to deliver the data in a regulated manner, knowing where the data is and who is using it, as the requester of the data had already proven his identity in the authentication process. Such information is reflected in **audits, logs and data tracers** that are available to the subject (required under GDPR legal constraints), provided through a user interface *c)*.
- The consent protocol flow is inverted from what is normally used in other protocols (i.e. OAuth2). The consent protocol starts from the subject's mobile device, and the data that reaches the terminal does not require to be identifiable. Since the terminal does not select the subject and does not collect the subject's information from the received data, the protocol is **compliant with pseudonymisation schemes**.

- The consent protocol flowing through *b*) communication channel is performed in two rounds. The consent is finalised in the second round after receiving the information about the terminal and who is currently using the terminal. However, the **data is pre-loaded** (with an unknown encryption key) as soon as the subject initiates the first round.

6.3.1 Method

As depicted in Figure 6.6, let's assume the existence of a terminal, a location and an operator with a smart card. The respective key pairs are ($t \times G = P_t, l \times G = L, o \times G = P_o$). We also assume that P_t and P_o are pre-registered and certified in the location, and the existence of an access control model to manage authorisations for those keys. The sequence diagram containing the data exchange between entities in the process described above is depicted in Figure 6.7. The protocol follows:

Start Session. The terminal establishes a session to the location each time the operator's smart card is activated. The terminal generates a random value e_t , a timestamp *time* for this session. Derives the ephemeral public key $e_t \times G = P_{et}$ and signature $\sigma_t \langle P_{et}, time \rangle$. With the operator's authorisation the smart card signs the terminal signature $\sigma^\dagger = \sigma_o \langle P_t, \sigma_t \rangle$, sending it to the location and binding P_{et} to the operator and terminal. These operations are represented in steps 1.1 to 1.3.

Bind Session. The location receives σ^\dagger , validating both signatures and recovering P_{et} and *time* (step 1.4). The location checks if *time* is in an acceptable range and if the operator and terminal keys (P_o and P_t) are authorised to establish the channel. The location generates a random value e_n , the ephemeral public key $e_n \times G = P_{en}$ and responds with $\sigma_n \langle P_{en} \rangle$. The same symmetric key for the session can be derived from the terminal and from the location via $H_p(e_t \times P_{en}) = H_p(e_n \times P_{et}) = k$ (step 1.4). The connection *a*) for the session is kept alive and is now ready to receive blocks of encrypted data.

Read Consent. The consent protocol starts with the subject reading the terminal key P_t with his mobile device from an out-of-band [188] channel (i.e. from a QR-Code, represented in step 2). Existing authentication mechanisms can be used to start the protocol *b*), for instance, using the assigned subject's key P_s in a Challenge Handshake Authentication Protocol (CHAP). Assuming the subject is correctly authenticated, the location randomly selects a symmetric key d for this specific consent session, this

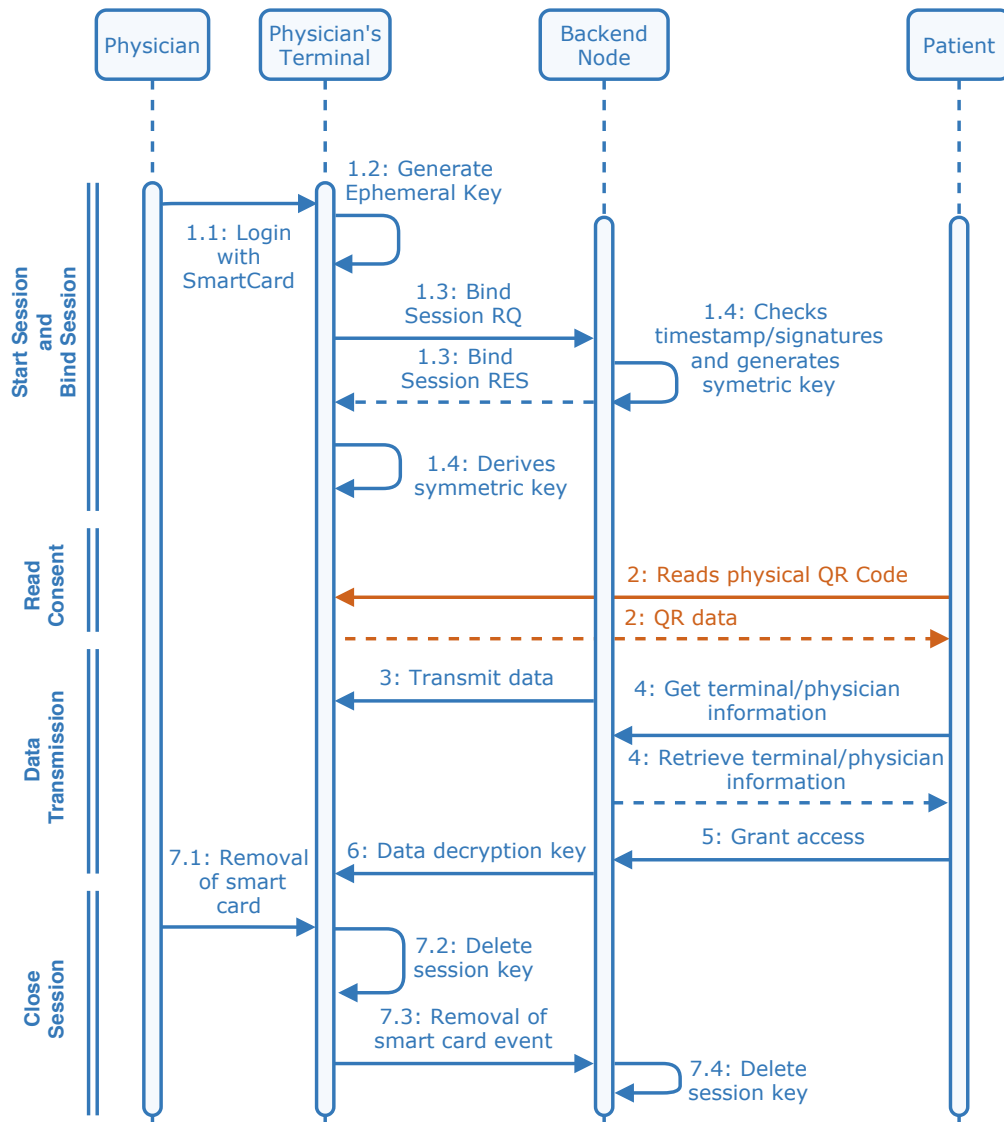


Figure 6.7.: Sequence diagram of the interactions between actors. There are four entities depicted: Physician, the Terminal where the Physician logs in with the smart card, the Backend node containing the desired information and the Patient.

key is kept secret in the location. Additional information about the terminal and the operator that is currently using the terminal is sent to the subject to finalise the consent (step 4).

Data Transmission. The moment that the subject starts the consent protocol in step 2, data can immediately be transmitted to the terminal via the session channel in the encrypted form $E_d[data]$ (step 3). The d key is only sent to the terminal (encrypted via $E_k[d]$) when the consent is finalised (step 5 and 6). In this manner, the required data may already be present on the terminal as soon the consent is finalised, minimising the perceived latency from the operator. The transmitted data

does not identify the patient at any moment as the data is pseudonymised before the encryption process in the node.

Close Session. If the smart card is removed from the reader, all session keys (e_t , k and all d keys for each consent) should be immediately deleted from the terminal, as also all the corresponding session keys in the location (represented in sub-steps 7.1, 7.2, 7.3 and 7.4). Although encrypted data may be persisted in the terminal's disk, such data blocks are unusable without the respective keys.

6.3.2 Security analysis

Forward Secrecy is maintained due to the use of ephemeral keys (P_{et} and P_{en}) for the key exchange and Diffie-Hellman construction $H_p(e_t \times P_{en}) = H(e_n \times P_{et}) = k$. Both keys are authenticated from each source with a digital signature (σ_t and σ_n). Man-in-the-Middle attacks are prevented, assuming P_t and P_n are known from each endpoint. Replay attacks are useless. Besides having the timestamp limiting the re-usability of sessions, k can only be derived from P_{et} and P_{en} key owners. These techniques are well known from Transport Layer Security (TLS) protocols.

Key Bond (Operator/Terminal). The P_{et} ephemeral key is bonded to the operator and terminal via authenticated signatures. The method assures that P_{et} is assigned to the terminal in the presence and with the authorisation of the respective operator P_o . The operator cannot assign the authorisation to a different terminal without the respective σ_t signature. The terminal, even if compromised, cannot receive data without the operator's authorisation. However, such authorisation is not linked to any specific subject. The key binding merely corroborates that the operator is using the terminal and that both are certified by the location. The operator/terminal information that is sent to the mobile device, complemented with visual cues (i.e. physician's name in a badge and terminal id), increases the subject's confidence that data is being delivered to the correct endpoint.

Pseudonymisation. There is nothing in the protocol that identifies the subject. The consent protocol starts at the mobile device without sending any information that is able to identify the subject. If the location data is pseudonymised, the terminal has no way of receiving the subject's identity. Furthermore, with the use of mixed networks (i.e. TOR [134]), statistical analysis of network traffic should be hard to implement. However, Global Positioning System (GPS) information from mobile devices should

be considered to protect pseudonyms, since terminals are geographically located and can be associated with the subject's consent.

6.3.3 Results and discussion

We used the Citizen's Card for the underlying smart card procedures. The Citizen's Card has a public Application Programming Interface (API)³ allowing the cryptographic operations like authentication and signature of data. Our resulting system makes use of this API to authenticate the actor accessing the terminal, leaving no doubts about the identity of the data requesting actor. Furthermore, the signature certificate allows to encrypt and verify the issuer of the messages between the terminal and the network nodes. Together with the session keys, this method prevents the tampering and leaking of the data being transferred.

These methods were implemented and simulated in a controlled virtual environment⁴ using the developed library⁵. Referencing the Figure 6.7 representing the communication between parties in the system, the flows between the physician's terminal and back-end nodes were successfully tested and validated. However, further improvements are needed relating to the development of mechanisms that give the patient the authority of granting or denying access to personal data.

Discussion. The physician terminal unlocks after the unequivocal authentication using the Citizen's Card and, therefore, all the following operations are registered. This method allows the patient to verify the actor accessing his data and check the requesting terminal. Once the patient allows access, information about the physician, terminal and resources accessed is provided. Moreover, the patient is assured that his entity is not disclosed during the session due to the personal details pseudonymisation. The proposed architecture is being deployed in the demanding medical imaging scenario. The integration is being conducted in an open-source PACS, the Dicoogle Open Source project [189, 190].

The method only describes the path for explicit consent. No paths for emergency situations (such as accessing data from unconscious patients) are available. However, both paths (explicit and implicit consents) are handled in the next chapter.

³<https://github.com/amagovpt/autenticacao.gov>

⁴<https://github.com/rlebre/SafeClinic>

⁵<https://github.com/shumy-tools/aot>

Pseudonymity

This chapter describes a distributed P-ID mapping function used to enhance protocols such as PIX with pseudonymity and break-the-glass functionalities. The P-ID is then extended with **implicit** and **explicit** consent routes in a more practical approach.

Pseudonymity is a major requirement in recent data protection regulations, and of special importance when sharing healthcare data outside of the boundaries of the affinity domain. However, healthcare systems require important break-the-glass procedures, such as accessing records of patients in unconscious states. Pseudonymisation is achieved by separating the subject identification from the anamnesis data, with the possibility of recovering the identity by reconnecting both of these data blocks. The essential part that connects both blocks is the pseudonym, which we will refer to from now on as π .

This section presents an original breakthrough that maps a unique fragment of public information id to a pseudonym π (mapping $id \mapsto \pi$), established on a (η, τ) -threshold secret sharing scheme and public key cryptography that is compliant with break-the-glass procedures, and named as the P-ID function. The P-ID is a deterministic threshold map function. The id value is public, unique and immutable for the lifetime of the data-subject. It should not contain any sensitive personal data or rely on any cryptographic method that could be broken over time.

The goal is to retrieve the same set of pseudonymised records from a location, using the id without revealing the original $id \mapsto \pi$ mapping to any of the federated nodes. Note that, the use of the id is important because this is the only piece of public information that can be used at any moment without restrictions. For instance, if the patient has an accident and is unconscious, an RFID chip or QR code tattoo can be used to get the id . Our method has two modes of providing consent (both as part of the GDPR legislation): **implicit** for emergency events and **explicit**, directly given by the patient.

The novelty of the method lies in an unusual application of the Lagrange interpolation, using elliptic curve points as inputs (equation 2.9). It is imperative that such a map cannot be performed without $\tau + 1$ parties, and without an unauthorised client. The distributed nature of the protocol makes it resistant to some cybersecurity attacks such as: insider attacks [191], ransomware, virus and worms, and denial-of-service. Other important properties are:

- **Break-the-glass compliance.** Pseudonyms can be deterministically derived from fragments of public information related to the data-subject (citizen-id, name, etc). No secrets are required to derive a pseudonym.
- **Offline attacks.** No τ number of parties are able to derive a data-subject pseudonym π , even when having access to the personal information that is used to derive such pseudonym. Only a set of predefined $\tau + 1$ parties should be able to derive π .
- **Statistical attacks.** Parties cannot identify any public information about the data-subject (discover $id \mapsto \pi$) even when clients perform a set of sequential reads and writes to the federated parties.

7.1 Master key setup

The existence of a master threshold secret key y , shared by the Quorum, is essential for all of the proposed methods in this thesis. Despite the existence of methods to setup a set of shares via a dealerless protocol, we describe one that is simple to understand and implement, although most probably not the best in terms of performance and efficiency.

Our master key setup method is an extension of the JR-VSS protocol, which provides a way to exchange the required shares in a public verifiable manner. Any party is able to verify that other parties have the correct information to derive the secret y , without requiring to access the private secrets of other parties. The protocol follows the steps:

Matrix Setup. Each party $i \in [1, \eta]$ derives $H_p(m_s || n_i \times P_j) \mapsto {}^i p_j$ as a secret between parties i and j , resulting in a Diffie-Hellman secret matrix. m_s is the master key session number, resulting in a different p for each session. The public matrix is

derived from ${}^i p_j \times G \mapsto {}^i P_j$ and is published to all parties. All parties should verify that the public matrix is symmetric, if $\forall_{i,j} ({}^i P_j = {}^j P_i)$.

Step 1. Every party i derives a random polynomial of degree τ with the local secret ${}^i y$ and the corresponding Feldman's coefficients ${}^i A_k$, where ${}^i A_0 \neq G$ and ${}^i A_0 \neq O$, being O the point at infinity. These coefficients are published, committing all parties to a pre-defined polynomial. The step is completed when a pre-defined number of parties commit to the polynomial.

Step 2. Every party i then constructs a set of shared secrets ${}^i y_j$ from the i party polynomial to be shared with j . The shares are encrypted and published in the form ${}^i p_j + {}^i y_j$.

Step 3. Every party now has the required public information to verify if all shares are correct. ${}^i Y_j = ({}^i p_j + {}^i y_j) \times G - {}^i P_j$ is derived in order to perform the verification $\mathcal{V}({}^i y_j)$ (from equation 2.10) without needing to know ${}^i y_j$, since ${}^i Y_j = {}^i y_j \times G$. If all shares from a party i conform to \mathcal{V} , the set ${}^i S_y$ is accepted. Each party is able to recover the corresponding shared secret from ${}^i p_j + {}^i y_j$, since ${}^i p_j = {}^j p_i$. All parties sum their shares from multiple sources to get $\sum_i {}^i y_j = y_j$, where $\mathcal{L}^j(y_j) = y$ and $\sum_i {}^i A_0 = Y$.

No one actually knows y , because there is no dealer. Our scheme requires memory resources in the order of $\mathcal{O}(\eta^2)$ due to matrix setups and shares. Using the proposed method is not mandatory for our main contribution. One can select in a range of available alternatives from other works [178, 179, 180, 192]. However, this is a simplistic protocol for a dealerless distributed key generation and therefore straightforward to implement and prove its security, an acceptable trade-off for small networks.

7.1.1 Master key setup security proof

The setup is resistant to rushing adversaries due to the pre-commitment of Feldman's coefficients (step 1) and proof in section 2.4.5. Rogue parties can still lie about the public matrix ${}^i P_j$. A non-symmetric matrix automatically shows the presence of a rogue party. However, a pair of rogue parties can generate equivalent rogue keys ${}^i P'_j = {}^j P'_i$, and can escape undetected at the matrix setup phase. Nonetheless, the

corresponding ${}^i Y'_j$ still needs to pass the verification \mathcal{V} . The fact that we need all ${}^i Y'_j$ values to be valid assures $\tau + 1$ shares to reconstruct the y_j secret anyway.

Key exposure. Even if an adversary derives $p + y = k$ from the encrypted matrix, k yields perfect secrecy [193] under finite field arithmetic if both values are indistinguishable from random. Defining y_2 as the key evolution of y_1 , then $y_2 = z_1 \cdot y_1$ is the transformation for the evolution process.

Frequency analysis and pattern matching does not work on results such as $(p + y_1) - (p + y_2) = (y_1 - y_2)$ because it is still indistinguishable from random. k is of no use when trying to obtain a pseudonym from $k \times P_{id} = p \times P_{id} + \pi$ since $p \times P_{id}$ can only be known if π is already known. The discovery of $p \times P_{id}$ is not useful when deriving other pseudonyms since the generator point P_{id} is different. Moreover, p values should be different for each master key session m_s to avoid solving the system of linear equations, where (p, y_1, y_2) are the unknowns:

$$\begin{cases} z_1 \cdot (p + y_1) = x_1 \\ z_2 \cdot (p + y_2) = x_2 \\ y_2 = z_1 \cdot y_1 \end{cases} \quad (7.1)$$

7.1.2 Results and discussion

The tool used to perform measurements is available in the fedpi github¹ branch. The master key setup simulations (in Table 7.1) measure different steps such as: time and memory consumption for the setup of secret matrices, commitment of Feldman's coefficients and encryption of shares.

The master key setup is a process with high computational cost. However, such a procedure should be uncommon in the lifecycle of the quorum and is asynchronous to other services.

¹https://github.com/shumy-tools/pseudo_break_glass/tree/fedpi

Table 7.1.: The average time (in milliseconds) and memory consumption (in Kb) for the master key setup for different threshold values τ , where $\eta = 3\tau + 1$. Rows 1, 2, 3 and 4 show: (m. setup) - the time to setup the secret and public matrices, (c. & e.) - the time to compute local secrets, the Feldman's coefficients and to encrypt the shares, (verif.) - the time to publicly verify the shares, (total) - total time of the master key setup procedure. Row 5 displays the memory consumption of the public matrix (in Kb).

τ	2	4	8	16	32	48	64
m. setup	29	21	101	344	1,156	2,627	4,251
c. & e.	1	4	14	62	298	688	1,156
verif.	8	55	300	2,272	20,607	65,585	127,431
total	38	80	415	2,678	22,061	68,900	132,838
mem.	1.5	5.3	20	75	294	657	1,164

7.2 P-ID method

Assuming that an authorised client starts a session to derive π , the procedure follows the steps:

Step 1. A point in an elliptic curve is derived from an hash-to-curve² function $HtC(\cdot)$, using the public information id , where $HtC(id) = P_{id}$. From the properties of elliptic curves we know that for any point in the curve, there is a scalar α_{id} such that $\alpha_{id} \times G = P_{id}$. Yet, α_{id} cannot be know under the DLP hardness assumption. A random $r \in \mathbb{F}_p$ value is selected such that $r \times P_{id} = R_{id}$ is derived. R_{id} is then sent at least to $\tau + 1$ parties in the quorum. r is kept as a secret for the request-response session.

Step 2. Each party i uses its secret share y_i and derives $y_i \times R_{id} = \pi_{r/i}$. The public point of the share is then $\pi_{r/i}$. The set of $\pi_{r/i}$ public shares from all parties is defined as π_r .

Step 3. The client collects all the necessary shares and calculates the Lagrange interpolation for the public shares defined as $\mathcal{L}^i(\pi_{r/i}) = \pi_r$ the same as $\mathcal{L}^i(y_i \cdot r \cdot \alpha_{id} \times G)$. From Equation (2.9) this is also equivalent to $\mathcal{L}^i(y_i) \cdot r \cdot \alpha_{id} \times G = y \cdot r \cdot \alpha_{id} \times G = \pi_r$. The client removes r from the result by performing the inverse operation $r^{-1} \times \pi_r = \pi$, resulting in a deterministic map $id \mapsto \pi$ that only $\tau + 1$ parties can compute.

²<https://eprint.iacr.org/2009/226.pdf>

The short resume of the threshold map function follows:

$$r \times \text{HtC}(id) = R_{id}, \mathcal{L}^i(y_i \times R_{id}) = \pi_r, r^{-1} \times \pi_r = \pi$$

Any extra information I_k can be attached to the pseudonym (including aliases), as long it conforms with anamnesis data. Any given map $\pi \mapsto I_k$ should not give hints about personal information.

7.2.1 Security analysis

The ultimate goal of an adversary is to discover any $id \mapsto \pi$ mapping, by exploring weaknesses in the protocol or attacking the master key directly. Both passive and active attacks are open to the adversary. This security analysis is performed under the following P-ID properties:

Break-the-glass compliance. The P-ID function is an open scheme, meaning that, the client doesn't require persistent secrets to derive π . The r secret is derived randomly for each new session. Since there are no client secrets, anyone can execute the protocol just by having access to the id information. However, the open scheme format requires additional countermeasures to defend against oracle attacks. The P-ID function cannot be openly accessible, since it can be used to perform queries and guess any $id \mapsto \pi$.

Offline attacks. From the P-ID function, a single party cannot get the map $id \mapsto \pi$, even when having a database of all possible values of id or P_{id} . Note that $y \cdot \alpha_{id} \times G = \pi$, where y and α_{id} cannot be known without resolving the DLP. Moreover, y can only be plugged into the result with a $\tau + 1$ multiparty computation. Even knowing $(G, \alpha_{id} \times G, y \times G)$, the direct application of the CDH hardness assumption states that it is not possible to get $y \cdot \alpha_{id} \times G$.

Statistical attacks. The P-ID function has a strong resistance to statistical attacks in the event of at most τ compromised parties. For instance, it is easy to map $id \mapsto P_{id}$ by computing $\text{HtC}(id) = P_{id}$ and searching the result in the database. If P_{id} was directly provided to a party (in *Step 1*) instead of R_{id} , it would be possible to extend the map $id \mapsto P_{id} \mapsto \pi$ by correlating the first request with subsequent requests that used π . Yet, by sending $r \times P_{id} = R_{id}$, the only possible correlation

$R_{id} \mapsto \pi$ has no useful information, and R_{id} cannot be reverted to P_{id} due to the DLP hardness assumption. R_{id} is indistinguishable from random if r and P_{id} are uniformly distributed. This is a reasonable assumption if we also assume that $HtC(id)$ gives an uniform distribution, and in general hash-to-curve function are designed for such cases.

7.2.2 Results and discussion

The tool used to perform measurements is available in the master github³ branch. We measured the scalability of the P-ID function by taking run times for a different number of parties and threshold values. The results in Table 7.2 show a linear progression of the running times.

Table 7.2.: The average of a run (in milliseconds) for different numbers of parties and threshold values.

η	10	20	40	80	160	320
$\tau = \eta/2$	1.072	2.010	4.141	7.533	15.857	38.858
$\tau = \eta/3$	0.929	1.732	3.701	7.139	13.735	32.224
$\tau = \eta/4$	0.865	1.612	3.112	6.748	12.739	28.729

The P-ID function addresses limitations of PIX profile, when pseudonymisation is an important feature under break-the-glass constraints. Our method decouples the query engine, authentication and authorization, which may be a source of security issues when not strictly necessary.

7.3 Explicit and implicit consent

The P-ID is the foundation for the implicit consent route. However, a practical application requires two modes of operation (both as part of the GDPR legislation), or consent routes. As defines in the architecture, Figure 5.1: **implicit** consent *d.1*) for emergency events and **explicit** *d.2*), directly given by the patient. This section applies some minor changes to the P-ID method in order to fulfil both modes. The flow of execution is similar in both modes, starting at the authorization point and ending at the authorised terminal P_t . P_t is both a terminal identifier that can be used for routing protocols and for public key encryption schemes such as the Elliptic Curve Integrated Encryption Scheme (ECIES) [194]. Note that verifying an authorization can be done by an external system or locally at each party.

³https://github.com/shumy-tools/pseudo_break_glass

7.3.1 Implicit consent (iP-ID)

The iP-ID mode provides a protocol for implicit consent. A break-the-glass authorization is required before the pseudonym is disclosed. The protocol is defined under the scenario depicted in Figure 7.1.

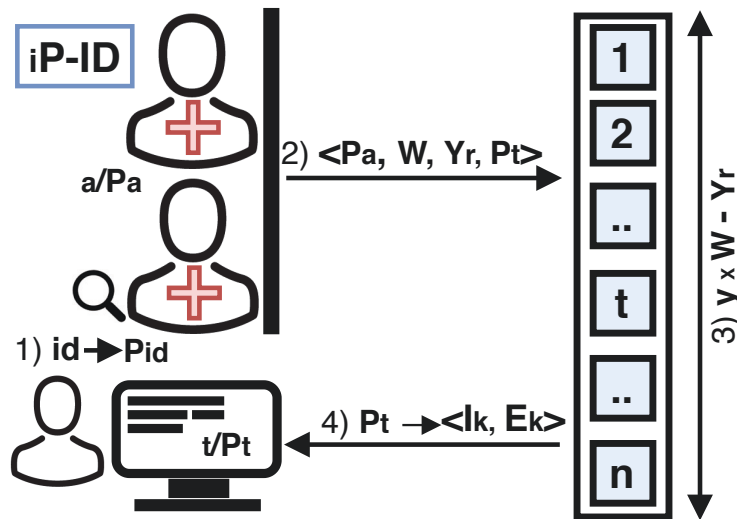


Figure 7.1.: The implicit consent mode (iP-ID) execution flow with the possibility of using multi-factor authorizations.

Step 1. The protocol starts with a physician collecting the public information id from the patient. id can be directly provided from an authorization point or from a terminal that can feed the authorization point. It is then used in a hash-to-curve function to get $HtC(id) = P_{id}$, where there is an unknown α_{id} such that $\alpha_{id} \times G = P_{id}$. The identifier of this request session is $u_i = H_p(P_a || \sigma_a)$.

Step 2. The authorization point selects a random value r , where $r \times G = R$, $W = (P_{id} + R)$ and $r \times Y = Y_r$. The proof of authorization $\sigma_a \langle W, Y_r, P_t \rangle$ for a specific terminal P_t is sent to at least $\tau + 1$ parties. The main difference from the original P-ID function is the sum of the random nonce R with P_{id} instead of $r \times P_{id}$, where W replaces the goal of π_r .

Step 3. Each party verifies if the signature is valid and if P_a is an authorised break-the-glass client. Each party i uses its secret share y_i and derives $y_i \times W = W_i$. Each W_i is authenticated and distributed through $\sigma_i \langle W_i, u_i \rangle$, signed with the party's key n_i .

Step 4. The terminal collects at least τ shares for the corresponding session u_i . u_i can be used to check the origin of the authorization request. The terminal is now able to recover π from the resulting multiparty computation $\mathcal{L}^i(W_i - Y_r) = \pi$. From the Lagrange homomorphic properties and Equation (2.9) this is equivalent to $\mathcal{L}^i(y_i) \cdot (\alpha_{id} + r) \times G - r \times Y$, resulting in $y \cdot \alpha_{id} \times G = \pi$.

Step 5. Using π , the corresponding dataset can now be retrieved by the authorised terminal. The terminal public key P_t can be used in an ECIES protocol to encrypt the transmission stream. The final result is retrieved as an ephemeral mapping $m_i = I_k$ for the current client session. The terminal must receive at least $\tau + 1$ correct and authenticated results for a consistent read.

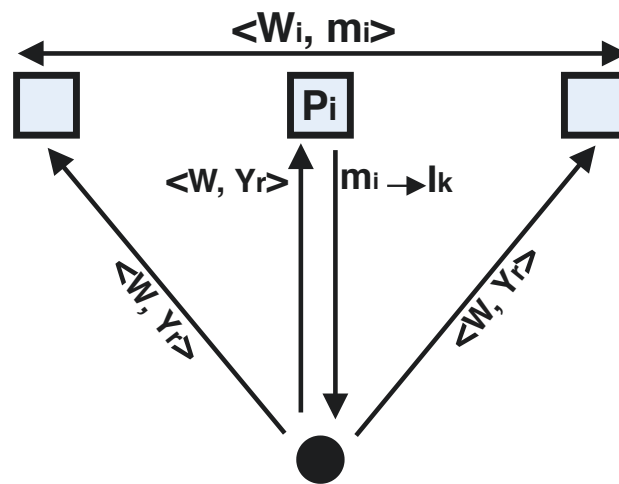


Figure 7.2.: Resume of the multiparty computation. Only the selected party P_i provides the dataset result for the terminal.

Any extra information I_k (clear text fields) or E_k (encrypted fields) can be attached to the pseudonym, as long it conforms with anamnesis data. Any given map $\pi \mapsto I_k$ should not give hints about personal information. The resume for the flow of messages in the multiparty computation is depicted in Figure 7.2. The example is for a $\tau + 1 = 3$ configuration, where the client and terminal are the same (for simplification). Note that, only the selected party P_i retrieves the dataset, others can retrieve a hash fingerprint of the result for integrity checking. We leave the protocol open for other optimizations, such as selecting additional redundant parties, or having an active terminal selecting and resuming the dataset result from a different party when a timeout occurs.

7.3.2 Explicit consent (eP-ID)

The explicit mode assumes a pre-registered mapping $P_s \mapsto \pi$ between the data-subject's authorization key and the pseudonym. The authorization key should not disclose anything about the owner's identity. The mapping does not necessarily need to be set in the same system where pseudonyms are accessible, and may even be part of an external access control service. We assume this simplification for the moment, although the method on how to set this mapping properly is proposed in Chapter 9.

The eP-ID flow in Figure 7.3 describes a possible method using a cell phone as an authorization point. The necessary modifications for each step follows:

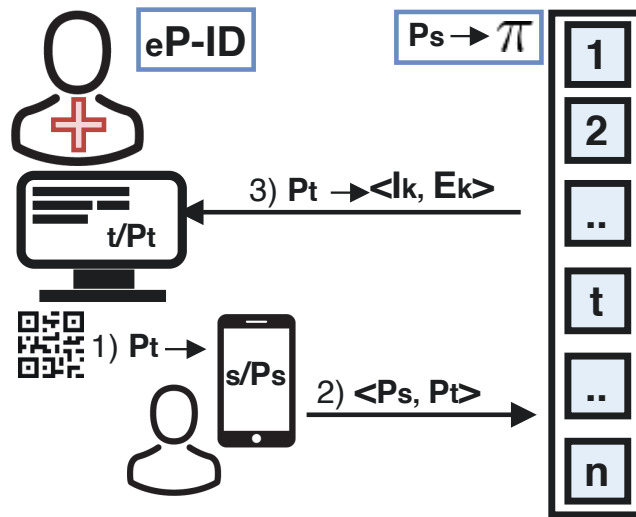


Figure 7.3.: The explicit consent mode (eP-ID) execution flow with a pre-registered map of $P_s \mapsto \pi$. No multiparty computation is performed in this mode.

Step 1. The data-subject uses a cell phone to collect the terminal identifier P_t via QR code or other accessible method.

Step 2. The authorization point just needs to send the authorization proof $\sigma_s \langle P_t \rangle$ with the terminal identifier. P_s is implicit for the σ_s message.

Step 3. Each party verifies if the signature is valid and searches for the $P_s \mapsto \pi$ mapping. The dataset and fingerprints are returned to the terminal in the same way as in iP-ID. In this case, the session is derived from $m_i = H_p(\pi || \sigma_s)$.

7.3.3 Impacts on the DICOM standard

From a PACS perspective, only the patient identification is affected by the P-ID method. Some additional protocols are required for the authorization procedure; however, those can work in parallel with the DICOM standard. Note that, in steps iP-ID 4) and eP-ID 3), we can ignore retrieving data via a secure protocol if a private network is being used. Data retrieving can still be done via the DICOM standard, as long as the patient identification is not revealed. The work to process and authorise the π does not require any changes in the DICOM standard, only internal changes in the PACS server.

7.3.4 Verifiable iP-ID

A verifiable iP-ID function should be able to confirm if the derived pseudonym from the multiparty computation is correct. We assume the identification of dishonest parties as a limitation of this thesis. However, there are some approaches that can be explored.

Checking the polynomial degree. Solutions such as the one proposed by Harn et al. [195], with Ghodosi's refinements [196], can be used to detect the polynomial degree from equation Equation (2.9), even if the shares are elliptic curve points. However, if j represents the number of shares used to reconstruct the polynomial, it requires $j \geq 2\tau + 1$ to detect τ colluding parties and an impractical number of runs $\mathcal{O}(j!)$ to identify those parties. Nonetheless, detection has acceptable requirements and more so if our work is to be integrated with Byzantine consensus protocols [197] (since the requirements are the same).

Checking equality of a discrete log. Note that, we can prove the equality of a discrete log between $y_i \times G$ and $y_i \times HtC(id)$ from "Proof Systems for General Statements about Discrete Logarithms" [198] using Schnorr's signatures. The normal Schnorr's signature is replaced with a double interlaced signature. The protocol could be extended with the following procedure to check individual shares:

SignInterlaced. Each node selects a random nonce $m_i \in \mathbb{F}_p^*$ and derives $m_i \times G = M_i$ and $m_i \times R = M_{w/i}$. Calculates $c_i = H_p(Y_i || W_i || M_i || M_{w/i} || u_i)$ and signs via $p_i = m_i - c_i \cdot y_i$. This is now the output for the signature $\sigma_i(W_i, u_i)$ as described in Step 3 for the iP-ID.

CheckShare. Each node is able to verify the share by calculating $p_i \times G + c_i \times Y_i = M_i$ and $p_i \times W + c_i \times W_i = M_{w/i}$. And then checking that $c_i \stackrel{?}{=} H_p(Y_i || W_i || M_i || M_{w/i} || u_i)$. This scheme is able to confirm that the discrete log y_i is the same for Y_i and W_i . It assures that the operation $y_i \times W$ was performed by the respective party. If it fails, then the party is not following the protocol.

This method is not part of the original proposal in the submitted article. For this reason, is not officially part of this thesis, but is nonetheless important to mention. Since individual shares can be checked as correct, the method can be used to detect rushing adversary attacks, as described in Section 2.4.5. Although, further research is required for the proof of security; checking the equality of a discrete log is a well-known method, and should be straightforward to provide such proof based on already existing work.

7.3.5 iP-ID security proof

The security proof is performed by playing a game where the adversary wins if the $id \mapsto \pi$ mapping is found.

Passive attacks. A single party cannot get the $id \mapsto \pi$ mapping, even when having a database of all possible values of id and P_{id} . Note that $y \cdot \alpha_{id} \times G = \pi$, where y and α_{id} cannot be known without resolving the DLP. y can only be plugged into the result with a $\tau + 1$ multiparty computation. Also, even knowing $(G, \alpha_{id} \times G, y \times G)$, a direct application of the CDH hardness assumption states that it is not possible to get $y \cdot \alpha_{id} \times G$.

When passively listening for the π result from the multiparty computation, the adversary needs to know the P_{id} value embedded in W to construct the mapping. However, W is indistinguishable from random if R is uniformly distributed. $P_{id} = W - r \times G$ cannot be obtained without resolving the DLP for r . Knowing the pair $\langle W, Y_r \rangle$, then $P_{id} = W - y^{-1} \times Y_r$, another DLP for y^{-1} . And even knowing the evolution key $z = e \cdot y^{-1}$ we can only get $W - e \times R = P_{id} + (1 - e) \times R$.

Active attacks. A rogue client cannot force honest parties to participate in a multiparty session without an authenticated request $\sigma_a \langle W, Y_r, P_t \rangle$. Furthermore, no party can receive more than τ compromised W_i values due to the previous requirement. Also, W_i values cannot be provided by non-participants due to the authenticated

message $\sigma_i \langle W_i, u_i \rangle$, or from different sessions due to the strong link between the authenticated client request σ_a and the multiparty session u_i . From the security assumptions, a $\tau + 1$ subset is not enough to derive the pseudonym π and get the $id \mapsto \pi$ mapping.

Verifiable iP-ID. The Feldman's verification cannot be directly applied here since the generator point P_{id} is unknown at the setup phase. The verification would need to be changed such that

$$y_i \times P_{id} \stackrel{?}{=} \sum_{k=0}^t x_i^k \cdot a_k \times P_{id} \quad (7.2)$$

exposing all a_k coefficients as also the secret y .

From Harn et al. [195] we can prove that τ invalid shares can be detected from $2\tau + 1$. The $y(x)$ polynomial with τ compromised shares is defined by equation 7.3, where $l_i^{[n]}$ represents the basis coefficients in the $[1, \eta]$ range. The result should be a polynomial of degree τ with terms of higher degree canceling in each part of the equation.

$$\sum_{i=1}^t y_i \cdot l_i^{[n]} + \sum_{i=t+1}^n y_i \cdot l_i^{[n]} = y(x) \quad (7.3)$$

For instance, by defining $\tau = 2$ and $\eta = 5$ the short version for the compromised shares is a polynomial of degree $\tau - 1$:

$$y_1 \cdot (a_1 \cdot x^4 + a_2 \cdot x^3 + \dots) + y_2 \cdot (b_1 \cdot x^4 + b_2 \cdot x^3 + \dots) \quad (7.4)$$

where the terms x^4 and x^3 must be cancelled to result in a correct polynomial degree. The cancellation process must solve a system of linear equations with the same number of variables (y_1, y_2) where only one solution exists (the correct one). Any attempt to introduce a different solution results in a polynomial of degree $> \eta - 1$. The proof can be easily generalized for (τ, η) .

7.3.6 Other security considerations

This section has important security considerations when implementing and integrating our work with production systems.

Attacks on authorised clients. Such attacks are still possible with less than $\tau + 1$ parties if such a client colludes with one party to get the $id \mapsto \pi$ mapping. However, such an attack is redundant if the client authorization key is already compromised. Since it is not possible to effectively defend against compromised clients without blocking existing functionality, the best countermeasure is to have a distributed audit and notification system. It would be difficult for unauthorised access to evade the distributed nature of the notification system. Such functionality would also be useful for GDPR notifications, but that is out of the scope of our paper.

Statistical attacks. Any public information used to construct aliases for pseudonyms should be as unique as possible for the results to be uniformly distributed. Any extra information attached to the pseudonyms should also be resistant to frequency analysis. It is not advisable to construct searchable encryption schemes from encrypted fields. This is a different class of attacks that we do not intend to directly tackle in this work.

Demographic queries. Any query mechanism should be decoupled from the federation of parties which maintains the pseudonyms. An alternative is to have an external system (query engine) mapping queries to public identifiers or aliases, in encrypted or clear text, depending on the requirements.

Registration procedures. These are not part of this thesis; however, they are important when registering the $P_s \mapsto \pi$ mapping for the first time, or attaching new information fields $\pi \mapsto I_k$. We assume that these mappings are correctly provided by external certified systems. External databases should not have any information that directly links to the P_s or π fields. An essential feature of the registration procedure is to verify if P_{id} is unique in the database. Further considerations should be taken when designing such solutions in order to maintain the security requirements.

7.3.7 Results and discussion

The tool used to perform measurements is available in the `fedpi github4` branch. The scalability simulations for iP-ID multiparty computations are reproduced in Table 7.3. Runtimes are evaluated for 100 runs with different number of parties η and threshold values τ , where $\eta = 3\tau + 1$.

Table 7.3.: The average time (in milliseconds) of a multiparty computation for different threshold values τ , where $\eta = 3\tau + 1$. Measurements are defined for: (recover) - the time to recover π , (detect) - the time to detect wrong shares and (total) - the total time of both procedures.

τ	2	4	8	16	32	48	64
recover	1	1	3	6	11	19	26
detect	1	4	18	73	256	664	1,195
total	2	5	21	79	267	683	1,221

The iP-ID multiparty simulation verifies input parameters (such as elliptic curve points) and message authenticity. Invalid shares are detected from a randomly selected set of $2\tau + 1$ shares. The critical component is the iP-ID multiparty computation (from Table 7.3), with the secret share verification (detection of wrong shares) consuming most of the time. However, from a practical perspective, this verification can be replaced with a lighter step, such as checking if a pseudonym exists in the database and only invoking full verification when required. The number of possible pseudonyms is around 2^{252} for the basepoint in the ristretto group. With the provided implementation and by the birthday bound [199], the chance of a collision is approximated by:

$$p(x) = 1 - e^{-\frac{k^2}{2 \cdot 2^{252}}} \quad (7.5)$$

Negligible when the number of registered pseudonyms k is below 2^{100} .

Optimal threshold setup. Table 7.4 is a simulation of the expected throughput for different τ values. The results show a throughput of 4558 multiparty computations per second (when ignoring wrong shares) at ($\tau = 16, \eta = 49$), a reasonable (η, τ) size for a EU federation. Shares are collected from multiple parties; however, the process is done in one asynchronous round with the pseudonym derivation being performed just in one party. The real iP-ID throughput should account for network

⁴https://github.com/shumy-tools/pseudo_break_glass/tree/fedpi

latency, multi-threaded context and all nodes actively processing requests (if clients hit different nodes per request); which, from a practical point of view, should be $\eta \geq 3\tau + 1$ for Byzantine networks.

Table 7.4.: The expected throughput for the iP-ID function in multiparty computations per second (mcps), without (mcps. no) and with (mcps. yes) wrong share detection. Assuming a network latency of 80ms between nodes and 8 dedicated threads per node. The formula for the interpolation is $8 \cdot (3\tau + 1)/(d + 80ms)$, where d is the time that the iP-ID function takes to recover π , as defined in table 7.3.

τ	2	4	8	16	32	48	64
mcps. no	691	1,284	2,410	4,558	8,527	11,717	14,566
mcps. yes	691	1,238	2,041	2,562	2,310	1,559	1,211

The respective interpolation is visualized in Figure 7.4. The goal here is not to be exact in the threshold calculation, but to predict what is the optimal setup for τ . Real values will be different depending on the deployed setup and; even if the number of dedicated threads is changed, this is a linear change that will not affect the curve of the graph. We can identify that the optimal setup for the threshold value, with wrong shares detection, is in the range $\tau \in [8, 32]$. However, the expected throughput grows without a boundary (in our domain) if we discard wrong shares detection. The η value is not considered since the multiparty calculation is not affected by the total number of parties.

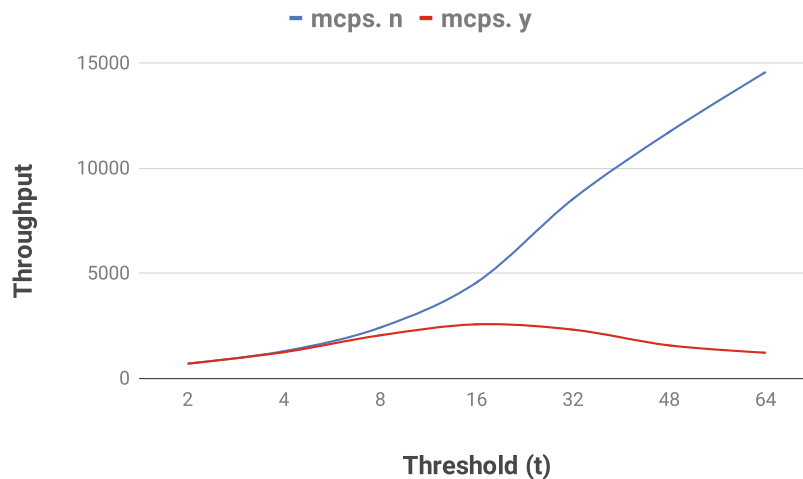


Figure 7.4.: Interpolation for the results of Table 7.4 with the identification of the optimal setup for the threshold.

Data reads. Our method adds a delay in the retrieve process because the calculation of a threshold secret is required. However, the data retrieve throughput is not affected. Once the pseudonym is calculated the data can flow normally from one

of the sources without any imposed restrictions. Knowing this, and from Table 7.3 results, we can expect an average delay (on normal conditions) for the π calculation of $(d + 80ms) = (6 + 80) = 86ms$ for $\tau = 16$. We believe that the overhead is acceptable for the gained security features.

Data writes. The work presented here is mainly focused on retrieving data. This is because the heavy part of our method is always done in a workstation, when data is being retrieved. However, data sources are an important part of real data flows, and these may even have restricted environments (such as embedded sensors). The only restrictions we put on existing data sources is that those must write data in an anonymous way (assigned to a π identifier) and probably via a secure channel. Such restrictions should not impose more overhead on the resources that are already being used. If a data source is trusted, in the end, is just the same as linking the source data to a pseudo identifier.

7.4 Experimental setup

All experiments for this chapter were carried out in a single machine running Linux (Ubuntu 18.04.1 LTS) with an Intel i7-7700HQ CPU @ 2.80GHz with 4 physical cores and 16GB of physical memory. All tools were implemented implemented in Rust⁵ with the help of `curve25519-dalek`⁶ crate. The `ristretto` group⁷ abstraction is used.

⁵<https://www.rust-lang.org>

⁶<https://github.com/dalek-cryptography/curve25519-dalek>

⁷<https://ristretto.group/ristretto.html>

Data Encryption

This chapter proposes a secure architecture for medical imaging storage and management in open and distributed environments. The P-ID construction is reused here to enhance key-management with break-the-glass functionalities.

GDPR constraints (such as encryption and anonymisation) push data storage architectures to a new level. To satisfy those constraints, our proposal describes a method that encrypts data but still authorises free circulation of that data between controllers, even between controllers that are restricted to the data. In this way, the architecture delivers backup services and promotes collaboration between controllers. Our proposal is focused on the management of cryptographic keys without providing a full-fledged integration with external systems. Nonetheless, it is important to describe how image acquisition and retrieval integrates with our proposed architecture. By leveraging current distributed technologies, such as Tendermint¹ and distributed file systems, such as InterPlanetary File System (IPFS)², our proposal can provide a secure and highly available federation of data curators that can be implemented for any PACS system.

8.1 Architecture

The overall architecture depicted in Figure 8.1 defines the entire flow of data. The first step is to identify the patient at the acquisition terminal *a*). The terminal represents a GDPR data controller. The pseudo-identifier π is provided and authenticated via a smart-card, cell phone or other pre-established method. The terminal uses this identification for the DICOM Modality Worklist, containing the list of requested examinations in clinical practice. The acquisition is performed and sent to a PACS archive proxy *b*) (i.e. Dicoogle³), mapping DICOM storage commands (*C-STORE*) to encrypted records and files *c*). Encrypted records R_n have useful information to

¹<https://tendermint.com>

²<https://ipfs.io>

³<http://www.dicoogle.com>

recover encryption keys and references to DICOM files; these are sent to a distributed ledger. DICOM information is encrypted and sent into files F_n to a distributed file system. The granularity of DICOM blocks depends on the retrieve requirements. It is possible to pack entire studies in a zip format or send files one by one. Workstations perform read operation $d)$ by retrieving and recovering F_n files, using encryption keys from R_n .

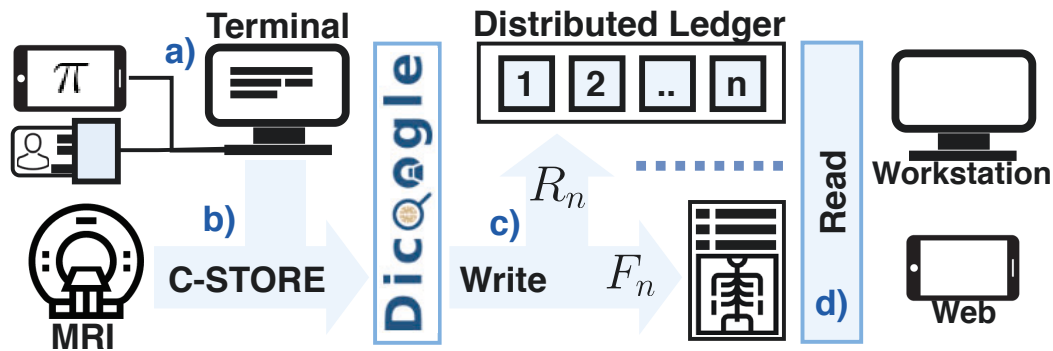


Figure 8.1.: Overall architecture of the proposed method, defining the data flow from acquisition $a)$ to retrieve $d)$. In between, DICOM files are encrypted and stored in two separated blocks, R_n and F_n with cryptographic and pixel-data information, respectively.

Since the ledger and the file system are distributed, any data curator that is subscribed to the network would be able to access any DICOM files in an encrypted format. However, the cleartext is only accessible to some authorised parties. Encryption keys can only be recovered with the collaboration of a minimum number of ledger nodes, defined as the threshold τ . The actors who are authorised to access those keys (access control information) are also included in R_n records, inserted by the data controller and managed by ledger nodes. The way how R_n records and F_n files are used is the focus of this chapter.

8.1.1 Distributed ledger

There are different types of DLT. For our use case, we advise the use of solutions where privacy and governance is easily achievable, and where currency mining is not a requirement. Governance is utterly important to maintain compliance with legislation and further developments. Mining procedures are not important for federated networks and are a waste of resources for such a scenario. Through DLT we can expose and distribute a single source of truth without having to trust an isolated node on the network. Tendermint is suitable for a private ledger, easy to use and customise via Application BlockChain Interface (ABCI). R_n records can

be synchronized automatically between nodes with support for ACID transactions (depending on the selected database backend).

8.1.2 Distributed file system

The file system architecture is depicted in Figure 8.2 where IPFS integration is proposed. Distributed file systems such as IPFS are able to pin a file (permanently persisted) on the original controller’s storage. Pinning actions are managed manually. *C-STORE* commands from acquisition stations are converted to IPFS pinning actions via a PACS proxy. IPFS also maintains a cache of recently accessed files, managed by a garbage collector.

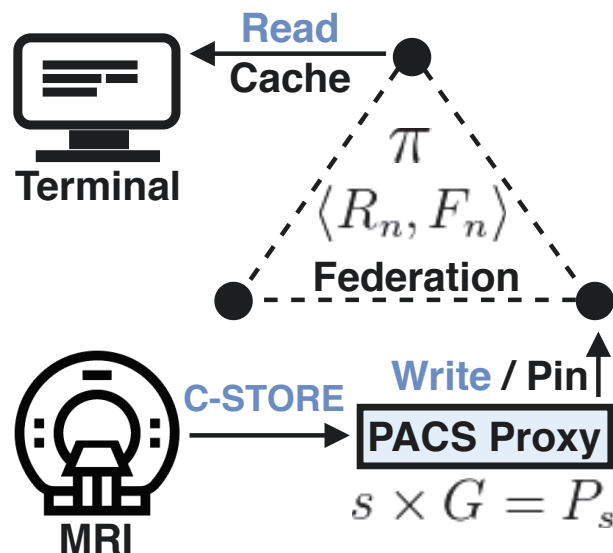


Figure 8.2.: The overall architecture representing a federation of data curators where data sources and terminal workstations (viewers/workstations) are connected via IPFS network.

An authorised terminal can read files from any point in the network. Depending on where the results are stored, those reads can be retrieved from the cache or directly from the source (pinned files). In order to have better data redundancy and performance, several nodes can collaborate by offering to pin each other’s files, exposing multiple source alternatives. Furthermore, files are authenticated with the digital signature of the source key P_s , ensuring the origin and integrity of data. P_s can be the subject’s key or any other key authorized to write data in the EHR.

8.2 Storage methodology

The proposed method defines a cryptographic structure to manage encryption keys and file references. As depicted in Figure 8.3, a data-subject identified by π can have multiple datasets S_j with associated access control lists ACL_j . Those datasets are in general associated with a type of data, but such metadata tags are left open for future specifications. Assuming an implicit dataset index j , encryption keys are managed by chains of records R_n , where $n \geq 0$ is the index of the infinite set, defined by:

$$R_n = \sigma_s \langle prev, K_n, E_{\lambda_n}[\lambda_{n-1}, d_n, h_{F_n}] \rangle$$

where $prev = \pi || S$ for the first record R_0 , and $prev = h_{prev}$ for all other records $\{R_n : n > 0\}$. Those records are digitally signed by the source using the s private key. The chain is constructed with h_{prev} references, where the hash $H_p(R_{n-1}) = h_{prev}$. $K_n \in \mathbb{G}$ is a point derived from a randomly selected $k_n \in \mathbb{F}_p^*$, where $k_n \times G = K_n$. The encryption key of the key chain $H_p(\alpha_n || \pi || S) = \lambda_n$ can be derived from α_n with two different procedures: directly by knowing k_n from $k_n \times P_e = \alpha_n$ or indirectly via $\mathcal{L}^i(e_i \times K_n) = \alpha_n$. Other fields are: λ_{n-1} is the previous key, d_n is a randomly selected data encryption key and $H_p(F_n) = h_{F_n}$ is the file reference. $(\lambda_{n-1}, d_n, h_{F_n})$ are encrypted with the key chain λ_n . The F_n structure is defined as:

$$F_n = E_{d_n}[\sigma_s \langle d_n \rangle, data_n]$$

where d_n is authenticated with the data source private key s . The actual data $data_n$ for the respective record number n is encrypted using the d_n key.

8.2.1 Key integrity check

In a scenario where τ nodes can be compromised, the $\mathcal{L}^i(e_i \times K_n) = \alpha_n$ result may be incorrect. The Feldman's VSS scheme is commonly used to check if e_i shares are correct; however, the procedure cannot be used to check $\alpha_{n/i}$ points. For our construction, Feldman's coefficients would be of the form $a_k \times K_n$, where a_k are secrets and cannot be used to compute such a result. Moreover, the corresponding polynomial $\alpha_n(x)$ cannot be submitted by the data source, since it can easily compute the secret $\alpha_n(0) = \alpha_n$. Even when using checksum techniques for $\alpha_n(x)$ coefficients (i.e. Merkle tree), the procedure still poses challenges for proactive security and

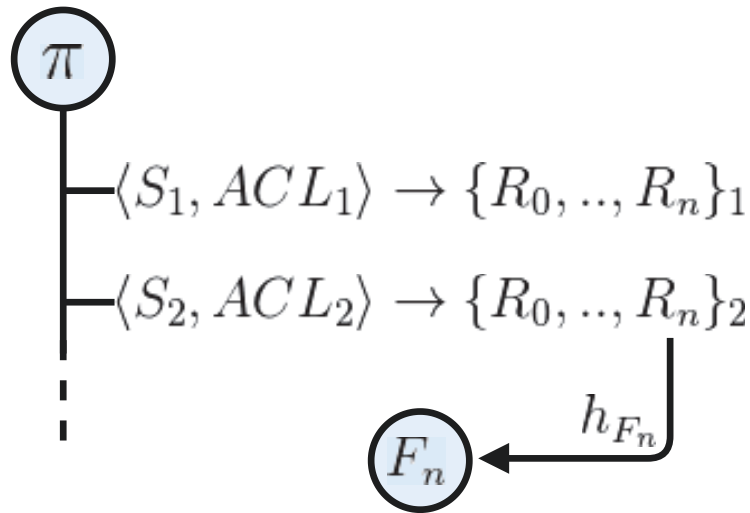


Figure 8.3.: Overall data structure for records and files (R_n, F_n) associated with an identity or pseudonym π .

committee management. If the coefficients change all checksums would be invalid. Updating the checksum is an expensive multiparty computation, since k_n is not known.

A different solution from Harn et al. [195] with Ghodosi's refinements [196], can be used; however, it requires at least $2\tau + 1$ shares to detect τ colluding nodes and an impractical number of runs to identify those nodes. If the goal is to detect a wrong result, an hash $H_p(\lambda_n)$ may suffice. Bilinear pairings [37] can also be used efficiently, for instance, testing for $e(G, \alpha_{n/i}) = e(K_n, P_{e/i})$, but this introduces another possible point of failure in the cryptographic suite. What should be the best approach, is left open for future work.

8.2.2 Read/Write procedures

When a data-subject enters a radiology centre for image acquisition, his pseudo-identification π and authorization $\sigma_a(\cdot)$ can be delivered to the acquisition station via an application on a mobile device, or even with a digital signature from the citizen card [25]. Both will require a public key mapping $\pi \mapsto P_a$ for verification. For instance, Password Authenticated Key Exchange (PAKE) [200] methods can be used for the authentication procedure. The authenticated proof can be submitted and linked with $\langle R_n, F_n \rangle$. For read procedures, PAKE methods can also be used to authenticate the terminal/workstation. The terminal can be authorised from the

access control model or via explicitly consent from the data-subject. Such explicit consent may also be delivered via a mobile device.

Write. Assuming π is known, the write procedure requires 2 rounds. The first round is requested from at least $\tau + 1$ nodes to obtain the last key from the dataset chain, $\mathcal{L}^i(e_i \times K_{n-1}) = \alpha_{n-1}$ and $H_p(\alpha_{n-1} || \pi || S) = \lambda_{n-1}$. The integrity of λ_{n-1} is verified. The second round submits $\langle R_n, F_n \rangle$ records with encrypted fields $(\lambda_{n-1}, d_n, h_{F_n})$. The encryption key for the new record is easily derived from $k_n \times P_e = \alpha_n$. Note that, only authorised clients are able to execute the multiparty computation $\mathcal{L}^i(e_i \times K_{n-1})$.

Read. An authorised terminal collects at least $\tau + 1$ shares to recover $\mathcal{L}^i(e_i \times K_n) = \alpha_n$ and $H_p(\alpha_n || \pi || S) = \lambda_n$, the most recent key for the chain. The integrity of λ_n is verified. The access is given to all R_n records in the dataset. Previous $\{\lambda_{n-i} : i > 0\}$ keys are recovered from the chain. All corresponding $H_p(F_n) = h_{F_n}$ files are retrieved from the file storage and recovered using d_n keys.

8.3 Security Analysis

We assume an “honest-but-curious” semi-trusted threat. More specifically, each controller acts in an “honest” fashion, will not maliciously delete patients’ records and correctly follows the designated protocols and computations. We assume, however, that controllers may insert corrupt records R_n sporadically and unintentionally.

The goal of the attacker \mathcal{A} is to recover the plaintext owned by a controller that is not being compromised. This means, if \mathcal{A} has administration access to a controller C_1 , it should not be able to access the plain text of C_2 without proper consent. However, we consider the plaintext data from C_1 compromised. Note also, consent management is not part of this work and we do not provide security analysis for this.

8.3.1 Security proof

This section should prove that emulation of data fences between controllers are protected against a maximum of τ attackers. The proof will be reduced to the DLP, Shamir’s Secret Sharing perfect secrecy (SSS-PS) and preimage/second-preimage resistance of the hash function (PSH).

Lemma 1 *The master key e is always protected against a maximum of τ number of attackers.*

Proof: Our method saves e_i shares in local storage for every node, making the master secret e protected against a maximum of τ compromised shares. e_i scalar values are never exposed directly in the protocol. Each share e_i is always exposed from $e_i \times K_n$, resulting in an elliptic curve point, protected via the DLP. We call these encrypted shares. Any \mathcal{A} needs to compromise $\tau + 1$ to recover e .

Lemma 2 *α_n keys can only be recovered from $\tau + 1$ encrypted shares.*

Proof: There are two paths to derive α_n . One via $k_n \times P_e$ and other via $\mathcal{L}^i(e_i \times K_n)$. Both require solving the DLP. k_n only gives access to data from the locally compromised controller. Furthermore, k_n keys are discarded in the writing process, exposing local plaintext data to a limited timeline. The only other way to access α_n is with the \mathcal{L} operator, protected via SSS-PS that requires $\tau + 1$ encrypted shares.

Lemma 3 *A unique λ_n key is derived from the secret α_n and public π, S values.*

Proof: The derivation process via $H(\alpha_n || \pi || S) = \lambda_n$ outputs an elliptic curve point that can be easily converted to a high entropy bit field. From PSH, the H function derives a unique λ_n that requires the secret α_n .

Theorem 5 *A can only be recover the F_n plaintext data $data_n$ of a non compromised controller using $\tau + 1$ encrypted shares.*

Proof: Assuming a strong symmetric encryption function E , $data_n$ can only be recovered by knowing d_n . d_n keys are locally stored for each controller. A controller C_1 cannot access d_n keys of C_2 . The only way for C_1 to access those keys is from $E_{\lambda_n}[\lambda_{n-1}, d_n, h_{F_n}]$. From Lemmas 1, 2 and 3 we known that the unique λ_n value can only be recovered from $\tau + 1$ encrypted shares.

From Theorem 5 we conclude that a compromised controller C_1 cannot access the plaintext data from C_2 without executing \mathcal{L} with $\tau + 1$ shares.

8.3.2 Informal security analysis

From $\langle R_n, F_n \rangle$ structures we can identify several security features:

Forward secrecy. When someone has a set of keys $\{\lambda_0, \dots, \lambda_n\}$ does not automatically give access to future keys, even when knowing $\{R_{n+i} : i > 0\}$, resulting in forwarding key protection. Any data controller can maintain their access rights, knowing that λ_n is initially derived from a secret that the controller has produced, such that $k_n \times P_e = \alpha_n$.

Insider attack. Insider attackers can only access local d_n keys and the controller's private share e_i . Local d_n keys give access to the controller's F_n files. Assuming a maximum of τ compromised shares, \mathcal{A} can only compromise the controller's files.

Rogue key attack. The rogue key attack [41] for a rushing adversary is effective in controlling the output of the master secret e . This is mostly a danger in the setup phase; however, we leave the setup of e_i and e for existing literature. Nodes can also control the output of $\mathcal{L}^i(e_i \times K_{n-1}) = \alpha_{n-1}$ or $\mathcal{L}^i(e_i \times K_n) = \alpha_n$, but these cannot influence the existing master key e . However, corrupting the α_n output can corrupt the chain integrity.

Chain integrity. In case α_n are corrupted, any data encryption key d_n and file reference h_{F_n} can be recovered by knowing the respective λ_n . Any previous keys in $\{\lambda_{n-i} : i > 0\}$ can be recovered from an healthy key chain, starting at λ_n . If the integrity of the key chain fails, λ_n values can be recovered from a $\tau + 1$ minimal set of $e_i \times K_n$ shares, achieving key-management with a single distributed master key e .

Data isolation. A controller is able to isolate and use owned data with local encryption keys d_n . Having d_n keys, there is no need to invoke the threshold protocol \mathcal{L} . The result of $\mathcal{L}^i(e_i \times K_{n-1}) = \alpha_{n-1}$ is only required to write the next record R_n if R_{n-1} is not owned by the same controller. R_{n-1} itself is not required. Access control policies can be used to limit the access of R_n records that are not owned by a controller.

Reference integrity. The same F_n item cannot be referenced in multiple R_n records of different sources due to the inclusion of the $\sigma_s(d_n)$ signature. Since d_n is locally randomly derived on each controller, C_1 cannot know the d_n key that C_2 used in the

writing process. However, it can recover these keys from the threshold protocol. If C_1 tries to link a R'_n to the same F_n it will fail in the verification due to different signatures σ'_s and σ_s from R'_n and F_n respectively.

Data confidentiality. Confidentiality is achieved with common symmetric encryption schemes such as Advanced Encryption Standard (AES). The symmetric keys are λ_i , protected via the threshold protocol, and d_n locally owned for each controller.

Proactive security. Having a single master key e it is easier to apply proactive security policies. e_i shares can be updated without changing e via a 0-sharing scheme.

8.4 Results and discussion

The tool used to perform measurements is available in the master github⁴ branch. Hardware instruction sets are used for Advanced Encryption Standard - Block Cipher Mode (AES-CBC) from the rust-crypto⁵ crate. Existing DICOM storage models [201, 202] were used to simulate storage/retrieve overhead. Based on these numbers we performed measurements on 100MB, 1000MB and 3000MB file sizes and compared the results with “openssl”. Note that, the content of such files has not impact on encryption/decryption methods, and so, there is no need to use a prepared DICOM dataset.

8.4.1 File encryption/decryption performance

The encryption and decryption of F_n structures can be tested via “./f-pacs Fn -s <size in MB>”. The test includes $\sigma_s(d_n)$ signature construction and verification. The average output is similar in all file sizes, around 430MB/s, both for encryption and decryption. Since the results are consistent for different file sizes, we can conclude that the signature process has an irrelevant impact on input/output performance.

The results for “openssl speed -elapsed -evp aes-128-cbc” and for the “aes-128-gcm” flag on packet sizes of 8KB were 1.34 GB/s and 5.44GB/s, correspondingly, showing

⁴<https://github.com/shumy-tools/f-pacs>

⁵<https://crates.io/crates/rust-crypto>

some performance gap from our results. Nonetheless, it is shown that the signature process has no major impact, leaving space for bulk AES improvements.

8.4.2 Meta-data encryption/decryption performance

Results for the creation and recovering of R_n records can be tested via “./f-pacs Rn -s <chain size> -t <threshold>”, and are presented in Table 8.1. The values are the average of 10 measurements with a standard deviation below 1. α derivation times are presented in Table 8.2, showing acceptable values throughout the threshold range. Note that, these calculations are performed on the client with minimal performance impact on the server side (only the additional retrieval of small α values).

Table 8.1.: The throughput results (in record per second, R_n/s) for different sizes of R_n chains and τ values. The create procedure is using $\tau = 16$.

size	1k	10k	100k	1,000k
create	3,158	3,285	3,077	3,082
recover-16	200k	490k	595k	604k
recover-32	100k	416k	606k	637k
recover-64	47k	270k	476k	598k
recover-128	19k	153k	471k	587k

Different threshold values were used, however, τ has no significant impact on the throughput of the “create” procedure. Note also, the “recover” throughput is mostly stable at high chain sizes, as long as the integrity of the chain is maintained. Moreover, the nature of the data requires more reads than writes, which is where the throughput is higher.

Table 8.2.: α derivation times (in milliseconds) for different threshold values.

τ	16	32	64	128	256	512
α times	5	13	23	56	147	439

8.4.3 Discussion

Although there are additional overheads for encryption/decryption of files and records, such information can be retrieved in parallel from multiple sites. The impact of recovering R_n records is insignificant compared to the effort of recovering an entire DICOM file, meaning, the throughput of large datasets is essentially bounded to the performance of the selected encryption method. For small files and datasets,

the overhead of the α calculation is non negligible for threshold ranges above 256. However, the threshold represents the level of redundancy and security, not the total number of nodes, that in actual Byzantine networks are around $3\tau + 1$ total nodes. In general $\tau > 128$ should be considered oversized. Furthermore, image pre-loading [203] is a standard procedure in modern DICOM visualizers. In such a scenario the perceived impact by the end-user is accounted for in the first image, where the α calculation is the only important measure to take into account. Assuming that $\tau = 64$ is more than enough for threshold security improvements, and knowing that the required shares are retrieved in parallel, our proposed method should only add a 23ms overhead to existing retrieving times. Finally, we should mention that federation networks are mainly composed of trusted entities, which requires a low number of τ nodes for acceptable security. Also, our results on AES-CBC are not very promising compared to years of “openssl” optimisations; however, as we stated, the encryption algorithm is not coupled to our method and can be replaced or improved.

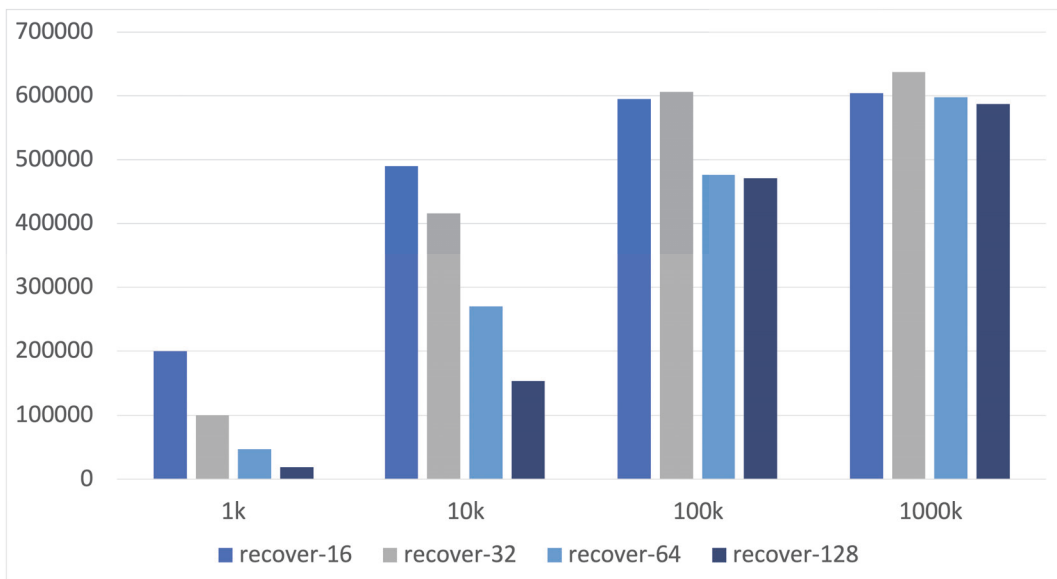


Figure 8.4.: The throughput results (in record per second, R_n/s) for different sizes of R_n chains and τ values. Values from Table 8.1.

From Figure 8.4, we can expect a stable R_n recovering method, even when using $\tau = 128$. R_n chains as long as one million records can be recovered around 600k R_n per second. The figure shows that the calculation of the first λ value for the chain has an impact on small chains, but it is dissipated for longer chains.

8.5 Experimental setup

All experiments for this chapter were carried out in a single machine running Linux (Ubuntu 18.04.1 LTS) with an Intel i7-7700HQ CPU @ 2.80GHz with 4 physical cores and 16GB of physical memory. All tools were implemented implemented in Rust⁶ with the help of `curve25519-dalek`⁷ crate. The `ristretto` group⁸ abstraction is used.

⁶<https://www.rust-lang.org>

⁷<https://github.com/dalek-cryptography/curve25519-dalek>

⁸<https://ristretto.group/ristretto.html>

Anonymous Access Token

This chapter describes the generation and verification of a holder-of-key access token in a (τ, η) -threshold setup, with extended break-the-glass and pseudonymisation features. The token is associated with a random k that can be used for authorization mechanisms.

From Chapter 6 we have a digital identity with a respective P_s identifying the subject and multiple data profiles I . Such identity needs to connect to the pseudonym π in each location L and corresponding associated data $\pi \mapsto data$. However, there is no method for an authorized subject to update/insert data records into π . The subject's identity key P_s should not be used directly, because the mapping $P_s \mapsto \pi$ can easily disclose the connection between the identity and the pseudonym. In Section 7.3.2 such mapping is used for the eP-ID procedure, assuming that P_s does not reveal the connection to the identity. Here, we define a method to set a key pair $k \times M = M_k$, where M is an alternative generator point for G , that can replace P_s without disclosing the original identity.

Nonetheless, we need a way to grant authorization to set the key mapping $M_k \mapsto \pi$. How should such authorization be granted while maintaining already defined security features?

9.0.1 Architecture

The main challenge is to construct a trustful architecture capable of generating access control tokens with break-the-glass and pseudonymisation features. An access token could be created by one node in the quorum. However, such centralized authorization may be circumvented by internal attackers. To minimise the lack of trust in a single authorization controller we propose a distributed architecture (depicted in Figure 9.1) which uses our threshold access token construction. The token is created using a minimal set of nodes $\tau + 1$ and has an associated private key k .

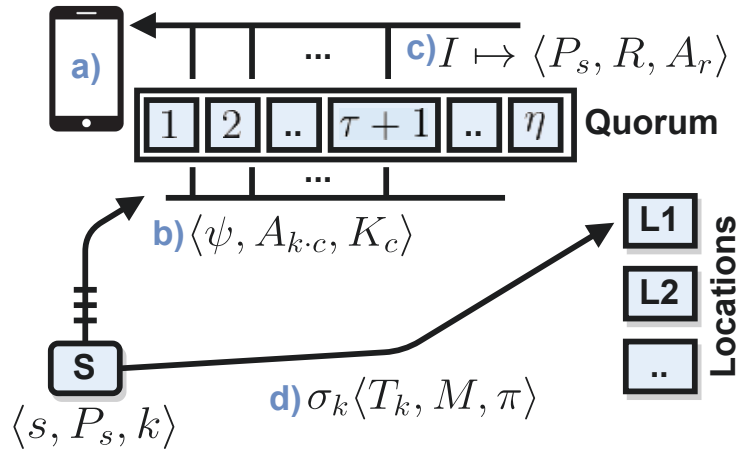


Figure 9.1.: Overview of the proposed architecture, access token generation and usage. A detailed explanation of the protocol flows is provided in section Section 9.2.

Considering that a subject’s profile I (subject id and data profile) is available at all federated nodes as public data c). A profile I is a set of public parameters registered in the committee and owned by a **subject** or resource owner. Profiles (identified by an anchor point R) are useful to decouple the resource owner from the respective anonymous resource, identified by π . The corresponding profile records (i.e. electronic health records) are stored at the location L_i and identified with the pseudonym π . The mapping $y \times R = \pi$ is not directly known because y is a secret shared by all nodes. Only by executing our proposed method such mapping can be disclosed. A third-party client C is able to request an access token b), where $\tau + 1$ nodes are required to construct the token. The token can then be used to read data d) from a pseudonym π at a designated data location L_i . Several authorizations and authentication schemes may be deployed here a), using a cell phone for multi-factor authorization or using similar techniques as in FIDO-U2F¹; however, it is not in the scope of this publication to describe those schemes and how to integrate them in our proposal. A detailed explanation of the protocol is presented in section Section 9.2.

9.0.2 Contribution

The resulting token is bonded to a pseudonym π that identifies a data resource. The resource is associated with a data profile I (that also has an owner) such that a one-to-one $I \mapsto \pi$ mapping exists, re-identifying the owner of π . Our proposal extends the concept of anonymous credentials by hiding the requester and the resource owner. The requester (or third-party client) is masked by the token key k , where the resource owner is masked by the pseudonym π .

¹Universal 2nd Factor authentication: <https://www.yubico.com/solutions/fido-u2f>

The distributed nature of the protocol makes it resistant to localized cybersecurity attacks such as insider attacks, ransomware, denial-of-service, virus and worms. Other important properties are:

- **Break-the-glass.** Pseudonyms are deterministically derived from an anchor R , related to a subject's profile. There is no need to contact the subject to generate the access token, as long as prior authorizations are set.
- **Offline attacks.** No subset of τ number of nodes are able to derive a pseudonym π , even when knowing I . Only a set of predefined $\tau + 1$ nodes can derive π .
- **Unlinkability.** The access token does not have any embedded information that can disclose the $I \mapsto \pi$ mapping. When intercepted by an adversary, only π is exposed by the token.
- **Accountability.** Refers to the ability to connect different streams of data with the same pseudonym. Abusive behaviour from a pseudonym can be stopped without revealing the true identity.
- **Proactive security.** The secret shares of a master key can be renewed with a 0-sharing scheme [204], maintaining the same secret. A server can start fresh and change the shares if there is a suspicion of being compromised.

Other use-cases. An access token that binds a control key k with a pseudonym π can have a multitude of use cases, including some that are not directly related to authorization and access control. The token payload can have useful information, implementing different use-cases. A few of these are listed here:

- Verifiable claims can be dynamically attached to the token, such as replying to a question “is a pseudonym correlated with a financial profile?”. These types of assertions are essential to add non-repudiation features to anonymous data. Claims can be attested by federated nodes without revealing the resource owner.
- Sometimes, users may want to be anonymous in certain online interactions (i.e. online discussion groups). However, when such users are blocked, the community must have a way to prevent users from using different pseudonyms.

If the identity/profile I is unique and verifiable, our scheme can guarantee that the pseudonym π is also unique and verifiable.

- The token can be used to invoke authorized commands and to bind certified information to anonymous resources without revealing who is the source of such commands. The resource owner can emit authorizations to many different entities, without revealing the authorized entities or the resource owner.
- Identities can have many different data profiles scattered throughout financial institutions, healthcare providers, social networks and governments. These providers (or data locations) can maintain the role of data custodians (or custodians of anonymous data) and yield liability for the services they provide. At the same time, all these profiles can be anonymously connected to a real and verifiable identity.
- Management of authorizations is shared between resource owners and federated nodes. Since the resource owner does not need to be online to give consent, break-the-glass paths can be constructed within our architecture in a (τ, η) -threshold security setup. Such alternative paths can be used in legal warrants or urgent medical situations.

9.1 Setup and threat model (extension)

This section is an extension of the original architecture (defined in Chapter 5). For the most part, the original architecture is still valid; however, additional parameters, cryptographic tools and extensions to the threat model are required to make the token work with the proposed properties. The extensions are:

Quorum. Besides y_i we also have a set of shares a_i derived from a dealerless share distribution protocol. Applying equation 2.9, a set of public values $\{Y \in \mathbb{G}_1, A \in \mathbb{G}_1, A^\dagger \in \mathbb{G}_2\}$ are pre-computed via $\mathcal{L}^i(y_i \times G) = Y$, $\mathcal{L}^i(a_i \times G) = A$ and $\mathcal{L}^i(a_i \times G^\dagger) = A^\dagger$. The correlation between (A, A^\dagger) is publicly verified as correct with the constraint $e(A, G^\dagger) \stackrel{?}{=} e(G, A^\dagger)$. Also $A \neq G$ and $A \neq O$ should check that $a \neq 1$ and $a \neq 0$, and $Y \neq G$ and $Y \neq O$ that $y \neq 1$ and $y \neq 0$. All a_i shares are discarded after the procedure. Ideally A^\dagger would be produced from an hash-to-curve function, and A from $\phi : \mathbb{G}_2 \mapsto \mathbb{G}_1$, but there is no such homomorphism in Symmetric XDH settings.

Profile. Each resource owner can register a profile I , where a set of shares r_i are randomly selected from a dealerless share distribution protocol. Two associated points $\{R \in \mathbb{G}_1, A_r \in \mathbb{G}_1\}$ are derived from $\mathcal{L}^i(r_i \times G) = R$ and $\mathcal{L}^i(r_i \times A) = A_r$ and registered in $I = \langle R, A_r, L_{addr} \rangle$. These are the extensions anchors as defined in Section 6.2.1. Each node in the committee is able to verify the parameters' correlation via $e(A_r, G^\dagger) \stackrel{?}{=} e(R, A^\dagger)$. Also $R \neq G$ and $R \neq O$ should check that $r \neq 1$ and $r \neq 0$. All r_i shares are discarded after the procedure.

Threat Model. Bilinear pairing is defined in Symmetric XDH settings. We assume that the master key y cannot be compromised via the setup procedure or reconstructed with less than $\tau + 1$ shares. We assume that ephemeral keys such as (a, r) and respective (a_i, r_i) shares cannot be compromised in the setup procedure. We assume that an adversary is able to compromise τ nodes, the location keys l , and use a rogue client to attack the protocol and inject crafted input parameters. We do not account for statistical correlations of requests between nodes and locations that could disclose the $I \mapsto \pi$ mapping.

9.2 Method

This section describes the details of the access token generation, verification and usage. Details of the request/response flows are illustrated in Figure 9.2. We assume the existence of secure channels for the information exchange. Although we use a simple authentication mechanism for the subject, via a digital signature σ_s , this is not a rigid requirement and may be replaced with other authentication standards.

Assuming the following definitions of *ThrGenToken* and *VerifToken* that will be used in our construction:

$ThrGenToken(y_i, m_i, k, R, A_r) \mapsto \langle T_k, M_k, M, \pi \rangle$: Produces an access token where π is exposed to the requesting client. The output is produced from:

1. $\mathcal{L}^i(m_i \times G) = M$ where $k \times M = M_k$
2. $\mathcal{L}^i(y_i \times R) = \pi$
3. $H_p(M || M_k || \pi) = c$ where $c \cdot k \times A = A_{k \cdot c}$

$$4. \mathcal{L}^i(y_i \times A_r + m_i \times A_{k \cdot c}) = T_k$$

$VerifToken(\sigma_k, T_k, M_k, M, \pi) \mapsto \{0, 1\}$: Checks if the token is valid. The signature $\sigma_k = (c_\sigma, r_\sigma)$ is verified using M as the base point instead of G . Knowing that $T_k = a \times (\pi + M_{k \cdot c})$ and $B = T_k || \pi$, both conditions should pass:

1. $r_\sigma \times M + c_\sigma \times M_k = M_\sigma$
2. Verify if $c_\sigma \stackrel{?}{=} H_p(M || M_k || M_\sigma || B)$
3. $H_p(M || M_k || \pi) = c$
4. Verify if $e(T_k, G^\dagger) \stackrel{?}{=} e(\pi, A^\dagger) \cdot e(M_k, A^\dagger)^c$

When a subject S wants to access a subject's profile I , it needs to request a disclosure of the $I \mapsto \pi$ mapping, to know what is the anonymous resource that belongs to that profile. To access the resource π (stored in a location L_{addr}), the client also requires a holder-of-key access token $T_k \in \mathbb{G}_1$, where $k \in \mathbb{F}_p$ is selected randomly by the client and authorized by the committee. As depicted in Figure 9.2, the procedure to get both of these requirements follows:

Start Session. The client C creates a session ψ to initialize the request from a strictly increasing sequence seq and timestamp tms , where $H_p(seq || tms) = \psi$. The profile I is selected and $\sigma_s(I, seq, tms)$ is sent to at least $\tau + 1$ nodes 1). Each node verifies if the client seq is above the last request and if the tms is in an acceptable range. The protocol proceeds if the client is properly authenticated and authorized to access the profile, 2) and 3). This requires a correct subject's key P_s associated to the profile. Each node produces a random secret share from $H_p(n_i || \psi || P_s || L || A_r) = m_i$. The anchor R is selected from the requested profile. Each node replies with the shares $\langle m_i \times G = M_i, y_i \times R = \pi_i \rangle$ that correspond to the session 4). The client collects the results of exactly $\tau + 1$ nodes and derives $\mathcal{L}^i(m_i \times G) = M$ and $\mathcal{L}^i(y_i \times R) = \pi$.

Token Request. The client generates a random value $k \in \mathbb{F}_p$, derives $k \times M = M_k$ and $H_p(M || M_k || \pi) = c$. Values $\langle c \cdot k \times A = A_{k \cdot c}, c \cdot k \times G = K_c \rangle$ are sent to the same selected $\tau + 1$ nodes 5), referencing the same session ψ . The client authentication for this second round may be achieved via a digital signature, or with the help of

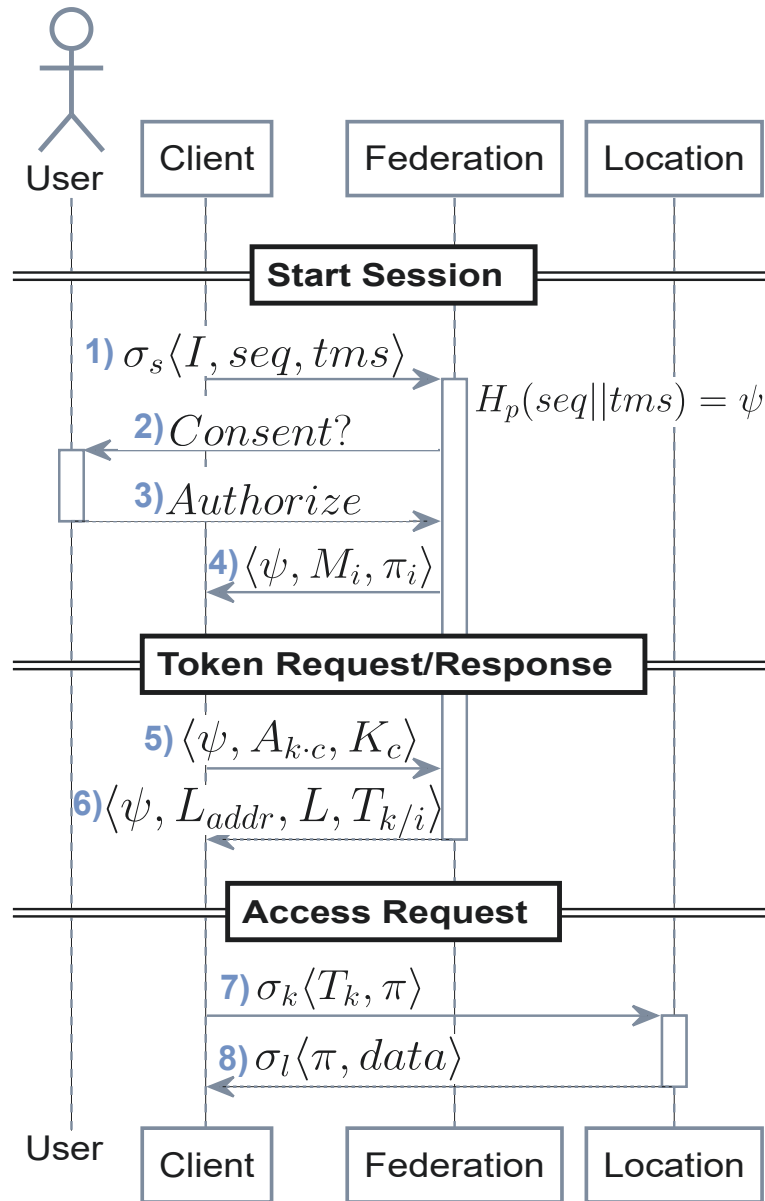


Figure 9.2.: Details of the protocol flows and interactions between clients, federation nodes, locations and data-subjects for the access token generation and usage. Requests to the federation are always to $\tau + 1$ nodes.

a pre-established secure channel associated to the session ψ . Using the property of equation Equation (2.1), each node is able to verify that $e(A_{k \cdot c}, G^\dagger) \stackrel{?}{=} e(K_c, A^\dagger)$, assuring that A is the base point of $A_{k \cdot c}$. Also $K_c \neq G$ and $K_c \neq O$ should check that $k \cdot c \neq 1$ and $k \cdot c \neq 0$.

Token Response. Each node captures the corresponding A_r from the selected profile I and previously inputs and parameters from the session ψ , replying with a token share $T_{k/i} = (y_i \times A_r + m_i \times A_{k.c})$. The location address L_{addr} and key L is also returned from each node 6). $T_k = a \times (\pi + M_{k.c})$ is reconstructed on the client from $\mathcal{L}^i(T_{k/i}) = T_k$. The *Request* \mapsto *Response* flow is resumed in the *ThrGenToken* function, resulting in the total output $\langle T_k, M_k, M, \pi \rangle$. The client now has the access token information for the respective pseudonym π .

Access Request. From all of the collected information the client is able to request access to π by sending $\sigma_k \langle T_k, \pi \rangle = \langle \sigma_k, M, M_k, T_k, \pi \rangle$ to the respective location 7). The token is verified for correctness in the location by executing the *VerifToken* function. The data attached to π is signed and returned if the token is valid 8), via $\sigma_l \langle \pi, data \rangle$.

9.3 Security proof

This section demonstrates the security of the proposed method. It is assumed that honest nodes verify if elliptic curve points are in the correct group and if scalars are in the correct range, avoiding an entire class of issues [181].

9.3.1 Token forgery

This section proves that the token is non-malleable and cannot be forged within the defined threat model.

Lemma 4 *Let $i \in [1, \tau + 1] \cap \mathbb{Z}$ and (y, m, a, a_i) be unknown secrets, where (y_i, m_i, a_i) are the required shares to recover (y, m, a) . Then, Y_a and M_a can only be constructed via $\mathcal{L}^i(y_i) \times A = Y_a$ and $\mathcal{L}^i(m_i) \times A = M_a$.*

Proof: Trivially, since (y, m, a) and a_i are unknown secrets, then, both $y \cdot a \times G = Y_a$ and $m \cdot a \times G = M_a$ calculations cannot be performed. We can only get (Y_a, M_a) using the shares (y_i, m_i) and the parameterized point A .

Lemma 5 *τ compromised shares from the set $\{m_i : i \in [1, \tau + 1] \cap \mathbb{Z}\}$ where $\mathcal{L}^i(m_i) = m$, cannot force m to a known value.*

Proof: An adversary with τ compromised nodes can try a rogue key attack [41] for a rushing adversary. These nodes are allowed to select their m'_i shares after observing the M_1 value of the honest node (via a rogue client), trying to force a known m . The following is the result of the rogue Lagrange interpolation:

$$l_1(0) \times M_1 + \sum_{i=2}^{\tau+1} l_i(0) \cdot m'_i \times G = m \times G \quad (9.1)$$

Assuming that the adversary is unable to guess m_1 from $m_1 \times G = M_1$, the exact combination of m'_i values that produce m cannot be known due to the intractable DLP for M_1 . If we define $x = \sum_{i=2}^{\tau+1} l_i(0) \cdot m'_i$ and $M'_1 = l_1(0) \times M_1$ then, the DLP is reduced to $(m - x) \times G = M'_1$.

Lemma 6 *Assuming a maximum of τ compromised shares and a rogue client, m is distinct for each session ψ with a high probability.*

Proof: An adversary can force m to be reused if it can compromise $\tau + 1$ shares, since it can directly chose all m_i shares for $\mathcal{L}^i(m_i) = m$. However, for a maximum of τ compromised shares, there is always one m_i that depends on a hash result from (seq, tms, P_s) . Since the strictly increasing seq is controlled by the honest node, and due to the second-preimage resistance, it is hard for a rogue client to force the inputs (tms, P_s) such that the hash function results in the required m_i . If m has enough bits, the probability of deriving the same m using the distinct and honest m_i value should be negligible. Note that, the tms is mainly present to minimise overflows of the seq value in real implementations. Moreover, the range of the tms value should be checked against the internal clock, and P_s should be a valid subject's key for the respective profile I .

Theorem 6 *Both Y_a and M_a are secret points, where M_a is distinct (with high probability) for each token T_k .*

Proof: From Lemma 4 both results are intractable Diffie-Hellman computations that can only be resolved via Lagrange interpolations. However, both interpolations that can compute those points are never executed independently. Even if we assume that (r, k) are compromised and remove those values from the token, T_k is preserved as a linear combination of the form $(Y_a + x \times M_a)$, without exposing Y_a or M_a individually.

Both secrets are always present in the token since $a \neq 0$, $y \neq 0$ and $m \neq 0$ with high probability, and also because $(x = k \cdot c) \neq 0$.

M_a cannot be recovered from subtracting two different tokens. Assuming distinct values for m (with high probability) from Lemma 6, then M_a values are also distinct for each token. This leads to the impossibility of a rogue client to get M_a from two different requested tokens, by performing $(T_{1k} - T_{2k}) = (x_1 - x_2) \times M_a$ and recovering M_a , assuming that x_1 and x_2 are known. This recovering mechanisms cannot be used since M_a is always distinct, leading to $(T_{1k} - T_{2k}) = x_1 \times M_{1a} - x_2 \times M_{2a}$ or $(T_{1k} - T_{2k}) = x \times (M_{1a} - M_{2a})$ if $x_1 = x_2$. Both M_{1a} or M_{2a} cannot be recovered independently. Consequently, Y_a cannot be recovered by this process if M_a is unknown.

From Lemma 5, m cannot be tampered with. Yet, if a rogue client is able to chose any values for $(A'_r, A''_{k \cdot c})$ such that the respective secrets (a', a'') are known, the previously defined linear combination is reduced to $a' \times Y + a'' \cdot x \times M$. Since Y and M are known, by tampering with one value (a', a'') at a time the adversary can recover Y_a and M_a . For instance, by subtracting $a'' \cdot x \times M$ from the equation, one can recover Y_a if a' is left unchanged, where $a' = a$. To prevent this attack, the following verifications are required at each node: $e(A_r, G^\dagger) \stackrel{?}{=} e(R, A^\dagger)$ and $e(A_{k \cdot c}, G^\dagger) \stackrel{?}{=} e(K_c, A^\dagger)$. These assures that A'_r and $A''_{k \cdot c}$ are always generated from the base point A and that (a', a'') are unknown.

Lemma 7 *A valid token T_k cannot be produced by randomly selecting M , M_k and π , or by using these parameters from other valid tokens.*

Proof: From the *VerifToken* function, the verification at 4) is equivalent to $a^{-1} \times T_k \stackrel{?}{=} \pi + c \times M_k$. However, since a^{-1} is unknown, this verification can only be performed using bilinear pairings. Moreover, if we check that $H_p(M || M_k || \pi) \stackrel{?}{=} c$, this is the same verification procedure that is used in the Schnorr's signatures. The main difference is that a^{-1} (that should correspond to the r_σ value in σ , being that $a^{-1} \equiv r_\sigma$) is a fixed value, and $T_k \equiv G$ is a dynamic base point. In this way, the signature verification can be done with Pairing-Based Cryptography (PBC) without requiring to know a . Nonetheless, the presence of a is verified with A^\dagger . The σ_k signature verification binds M as the base point for M_k . Any linear changes in (M, M_k, π) produces non-linear changes in c with the same security proofs as the ones used in Schnorr's signatures. Note that, values such as $(M_{1a} - M_{2a})$, are not valid since c

has a different committed value. Furthermore, since Y_a and M_a are unknown from theorem 6, one cannot produce T_k from a combination of those secrets. For instance, a valid $T_k = r \times Y_a + c \cdot k \times M_a$ would be possible if r and k were known.

Lemma 8 *A valid token T_k cannot be produced from another invalid token.*

Proof: Note that c is used blindly and T_k is not included in it as a commitment value. In this context an invalid $T'_k = \pi'_a + c' \times M'_{k \cdot a}$ can be produced by submitting a fake c' such that $c = c' \cdot c''$, where $H_p(M || M_k || \pi) = c$ contains the π target that we want to attack. The invalid T'_k could be transformed to a valid one via $c'' \times T'_k$, if $c'' \times \pi' = \pi$. However this would require to have a profile with a R' value such that $c'' \times R' = R$. Since r' is not controlled by the rouge client or a single node, this reduces to the DLP for R' .

The second alternative is to attack $M'_{k \cdot a}$, such that when it is used in the token computation results in $c'' \cdot m \times A'_{k \cdot c} = \pi_a - c'' \times \pi_a + c \times M_{k \cdot a}$. From the token transformation, $c'' \times \pi_a$ is eliminated, resulting in a valid π_a . However, to forge a correct value for $A'_{k \cdot c} = m^{-1} \cdot (c''^{-1} - 1) \times \pi_a + c''^{-1} \cdot c \times A_k$, one needs to solve the DLP for the unknown m value. Moreover, $A'_{k \cdot c}$ requires a known scalar $k \cdot c$ that depends on m in order to pass the $e(A_{k \cdot c}, G^\dagger) \stackrel{?}{=} e(K_c, A^\dagger)$ validation.

Theorem 7 *A valid T_k cannot be forged within the defined threat model.*

Proof: From Theorem 6, since (Y_a, M_a) are secrets, only $\tau + 1$ nodes can produce a $(Y_a + x \times M_a)$ combination with an interlaced x . From Lemma 7 and Lemma 8 we conclude that T_k is non-malleable. A valid or invalid T'_k cannot be transformed into a valid T_k .

9.3.2 Pseudonymity disclosure

This section proves that the $I \mapsto \pi$ mapping is not revealed from the token or any other public parameters. Note that k can disclose the mapping, but this is not a public parameter. We assume that the owner of an authorized k key already knows the mapping.

Lemma 9 *The embedded information in session ψ (seq, tms, P_s) does not correlate with the token public values ($T_k, M_k, M, \pi, c, \sigma_k$). Consequently, the session does not reveal the $I \mapsto \pi$ mapping.*

Proof: Even in the presence of distinct values such as (seq, tms, P_s), the resulting interpolations $\mathcal{L}^i(m_i) \times G = M$ and $\mathcal{L}^i(y_i) \times R = \pi$ does not contain any information of those values. M_k is derived from M and T_k is derived from (M_k, π) , with no correlation with ψ . By definition c must not contain any information from the session. Since the session is not used in the *Access Request* and the token has no direct correlation with the session, the session does not reveal the $I \mapsto \pi$ mapping.

Lemma 10 *Values associated with the token (T_k, M_k, M, π, c) do not correlate with the public parameters ($R, A_r, K_c, A_{k \cdot c}$) that are associated with the profile I .*

Proof: The results from $c^{-1} \times K_c = K$ and $c^{-1} \times A_{k \cdot c} = A_k$ are also associated with the token, but maintained in the private space of the client. Without prior knowledge of (K, A_k) that are in the private space of the client, the payload c cannot provide a correlation with ($K_c, A_{k \cdot c}$), in the public space of federated nodes. The main vulnerability is to get correlations via bilinear pairings. We will use subscripts to include scalars in points such as $m \times A^\dagger = A_m^\dagger$. Important pairing correlations are in the list:

1. $e(T_k, G^\dagger) = e(\pi, A^\dagger) \cdot e(M_k, A^\dagger)^c$
2. $e(A_r, G^\dagger) = e(R, A^\dagger) = e(A, R^\dagger)$
3. $e(A_k, G^\dagger) = e(K, A^\dagger) = e(A, K^\dagger)$
4. $e(M_k, G^\dagger) = e(K, M^\dagger) = e(M, K^\dagger)$
5. $e(M_k, A^\dagger) = e(K, A_m^\dagger) = e(M, A_r^\dagger)$
6. $e(\pi, G^\dagger) = e(R, Y^\dagger) = e(Y, R^\dagger)$
7. $e(\pi, A^\dagger) = e(A_r, Y^\dagger) = e(A, Y_r^\dagger)$
8. $e(\pi, A^\dagger) = e(Y, A_r^\dagger) = e(R, A_y^\dagger)$

We may recover useful information from 4), 5), 6), 7) and 8); however, we only have access to (G^\dagger, A^\dagger) points in the \mathbb{G}_2 group. Also, there is no homomorphism $\phi : \mathbb{G}_2 \mapsto \mathbb{G}_1$ that can produce the required points. From this result we should notice that Y^\dagger is also an important secret to retain anonymity. The interpolation that reveals this secret $\mathcal{L}^i(y_i \times G^\dagger) = Y^\dagger$ is never executed.

Theorem 8 *The $I \mapsto \pi$ mapping can only be disclosed by $\tau + 1$ multiparty computations or by the owner of k , within the defined threat model.*

Proof: From Lemma 9 and Lemma 10 we conclude that there are no direct correlations disclosing the mapping. From the DLP hardness assumption and information-theoretic security of Shamir's Secret Sharing, τ shares are not enough to produce the result from $\mathcal{L}^i(y_i \times R) = \pi$, where $R \mapsto I \mapsto \pi$. With r one could directly map $r \times Y = \pi$, but r is not known. With k one can connect $c \times K = K_c$, where $K \mapsto K_c \mapsto R \mapsto I \mapsto \pi$.

9.4 Results and discussion

Our measurements do not take into account the required latency to establish a secure channel between the client and nodes. These values are not included because different connectivity schemes may be deployed, i.e. maintaining a permanent connection to prevent a re-connection penalty.

The simulation assumes the required parameters from a pre-existing setup. The results are taken from a network simulation and API requests via function calls, available in the *tatadr.rs* code file. *start* and *request* functions in the *NetworkSetup* structure represents the *Start Session* and *Token Request* flows, where the *verify* function in the *Token* structure performs the *VerifToken*.

The client spends an average of 1.15 milliseconds to initiate the session (before sending any messages). Processing times for individual nodes for *Start Session* and *Token Request* are stable at an average of 0.97 and 1.22 milliseconds, respectively, with a standard deviation below 1. Token verification is stable at an average of 5.25 milliseconds, also with a standard deviation below 1. Only the aggregation of token shares are dependent on the threshold configuration, reported in Table 9.1. Both results follow τ in a linear pattern.

Table 9.1.: The average time (in milliseconds) taken by the client to process and aggregate $\tau + 1$ shares, for *Start Session* and *Token Response*.

τ	4	8	16	32	64	128
start	6.272	10.107	20.295	34.089	65.221	128.949
response	3.477	5.402	10.563	17.490	32.932	64.987

9.4.1 Discussion

Using values from Table 9.1 we can extrapolate the expected latency for a token generation when $\tau = 32$. Assuming a client-node round-trip latency of 80ms, the token can be produced in 2 rounds in about $(1.15 + 80 + 0.97 + 34.1) + (80 + 1.22 + 17.5) = 215\text{ms}$, an acceptable value for the level of security that the scheme provides. Furthermore, the aggregation time taken by the client does not impact the scalability of the number of nodes for the network.

Furthermore, there is an open issue in the π verification. A wrong y_i share can result in a valid T_k for a random forged π . However, applying Lemma 5 to y_i shares, such an attack cannot force the result to a known π . We assume it is hard to find an existing and valid π for the token to be useful due to the DLP. However, this attack can be used to perform a denial-of-service. From Harn et al. [195] work, we can prove that π is correct or incorrect by using $2\tau + 1$ shares. With $2\tau + 1$ shares, the recovered polynomial $\pi(x)$ is correct if it has a degree of τ . Yet, this can be expensive to compute and cannot detect the perpetrator of the wrong share. We leave this open for further research.

9.5 Experimental setup

Experiments were carried out in a single machine running Linux (Ubuntu 18.04.1 LTS) with an Intel i7-7700HQ CPU @ 2.80GHz with 4 physical cores and 16GB of physical memory. The tool² to perform measurements is implemented in Rust³. Our implementation uses a BLS12-381⁴ pairing-friendly elliptic curve from the BLS family [205].

²TAT-ADR Tool: <https://github.com/shumy-tools/tat-adr>

³Rust-Lang: <https://www.rust-lang.org>

⁴bls12_381 from zkcrypto: https://crates.io/crates/bls12_381

Conclusion

Healthcare interoperability is not driven by a lack of standards [206]. Rather, the main challenges are: adoption costs and complexity [207], gaps and overlaps [208]. DLT has the potential to address some of these interoperability challenges by providing: transparency, security, trust and, a reusable and cost-effective layer where fundamental standards can exist. The Health Information and Management Systems Society (HIMSS) defines interoperability at three levels: **foundational**, **structural** and **semantic**¹. It is an immeasurable work to tackle all these levels in a single thesis. Our contributions are focused on the reformulation of foundational and structural interoperability. The focus was on the following challenges:

- **A standardized method of identifying patients.** Although the IHE PIX profile is defined primarily for this task, our target is not just to identify connected records, but for patients to manage and be sovereign of their own identity. The combination of SS-IDs and DLT has an enormous potential in improving the current state of verifiable digital identities [209] and patient identification. If we want to provide direct access to patients' records, it is unrealistic to expect that users will securely manage their profiles across hundreds of different healthcare providers. This leads to poor password hygiene and slowly turns those data curators into targets for hackers looking for large amounts of personal data. This thesis addresses the challenges of identity management, such as the cancellation and renovation of private keys when they are lost or compromised. Achieving such a connection without using any sort of CA requires new methods to validate the ownership of the new key. The challenges are inflated when combined with pseudonymity and encryption features.
- **Propose federated architectures to unify different data curators.** Medical records are inherently spread between different silos (hospitals, clinics, etc). Connecting all these via a registry (central or distributed) is not the impeditive part. Many attempts and solutions had already been made, largely by tackling interoperability. Also, such architectures are designed for localized govern-

¹<http://www.himss.org/library/interoperability-standards/what-is-interoperability>

ments. DLT have enormous potential to scale such a registry to a worldwide level. However, existing DLT networks are for the most part public, posing security risks for the confidentiality of patients' records. This thesis tackles the difficulties of using public networks for medical records without compromising patients' sensible data.

- **Improve GDPR compatibility via pseudonymisation and encryption.** As previously stated, GDPR creates incentives and relaxes several requirements for controllers who pseudonymise and encrypt data. Although the application of symmetric encryption is effortless for the most part, providing secure key-management and key escrow is the main challenge in public networks. Furthermore, the generation of pseudonyms using a secret suffers from similar problems. This thesis proposes a distributed architecture that minimises the disclosure of secrets by reusing public networks as much as possible. Only a limited number of federated nodes are required to maintain a set of reusable secrets, that are, for the most part, immutable and stable.
- **Improve implicit and explicit routes for patient consent.** Securing medical records is kind of a “hurdle” due to the necessary exceptions to the rules. Any access control mechanism in this field is not complete without providing implicit consent routes (or mostly known as break-the-glass), necessary for medical emergencies. However, such a mechanism is generally the weakest link in the overall system security. This is exceptionally hard if we mix it with other requirements, such as key management and pseudonymity. This thesis is designed from the beginning to handle both consent routes without compromising any of them, as well as maintaining other requirements.

GDPR is here to stay, and there are not many solutions for the problems presented here. This is especially worrying in medical systems when protecting sensitive information. Our proposal is able to work seamlessly with DLT, giving a single source of truth with enhanced security and availability. We proposed an architecture for unified pseudonymity, key-management and data minimisation, creating well-defined frontiers between privacy realms. We support the emission of claims and attestations, complying with the GDPR as much as possible. We addressed the limitations of current pseudonymisation and key escrow methods with break-the-glass compatibility via a (η, τ) -threshold architecture. This architecture offers protection against a collusion attack (or failures) from τ parties out of η .

Open challenges

There are still many challenges ahead when we consider real-world products and deploys.

- **Enforce interoperability standards across facilities.** While organizations agree on the importance of standardization, they often interpret and enforce these standards differently. Sharing information across institutional boundaries requires an understanding of the data structures in order not to limit their usefulness. Although we can devise methods to evaluate and select interoperability standards [210], DLT may be used as an automated and authoritative mechanism, enforcing standardization and effectively providing regulation by code [211].
- **Coordinate policies across the industry.** It is important to coordinate institutions to develop consistent policies in order to circumvent interoperability obstructions. DLT can help to deploy, secure and enforce those policies via Smart Contracts [159] or other scripting infrastructure. Additionally, new emerging concepts like RegTech [212], representing the digitization of regulatory processes, may also add value to the technological solution. Digitization and rule-based regulatory processes are capable of automating policies and maintaining consistent decisions across domain boundaries.
- **Ending data sharing impediments.** In accordance with the report of the US Congress from the Office of the National Coordinator for Health Information Technology, information blocking [213] is still a prevalent problem in health data exchange. However, by the “Right to Access” enforcement, this must change. Also, the Centers for Medicare & Medicaid Services² released guidelines³ to prevent information blocking or any actions that inhibit the exchange of health information. When all the statements in the guidelines are met, a “Prevention of Information Blocking Attestation” is released for the clinician or group. DLT can further extend this to attest that data Processors are compliant with guidelines to prevent information blockage.

²<https://www.cms.gov>

³<https://www.cms.gov/Medicare/Quality-Initiatives-Patient-Assessment-Instruments/Value-Based-Programs/MACRA-MIPS-and-APMs/ACT-Information-Blocking-fact-sheet.pdf>

- **The “Right to be Forgotten”** is in direct conflict with the ledger immutability. Given the immutable and inalterable nature of the ledger, this conflict is a major risk for adopting DLT. For the most part, this is achievable by not storing any sensitive information on the ledger. Any sensitive information may be shared by external channels, and the hash fingerprint can still be registered on the ledger for integrity and non-repudiation purposes. However, assertions and other sensible registers still live in the ledger. Some solutions opt for encrypting the data [214] where destroying the key renders the data unreadable, assuming this is no different from the scrambling process when data is deleted. However, encryption schemes are doomed to fail through improved technology [183], and encrypted registries with outdated schemes cannot be removed from the ledger. This can be partially solved by maintaining an off-ledger connection between these registries and the personal identification of the ownership. This off-ledger binding can be deleted, and, if the remaining ledger information is no longer traceable to the original owner, the registries are considered anonymous and no longer fall into the GDPR. The “Right to be Forgotten” also means that important access references are also forgotten, and data access can be lost by invoking this right. This would also give the citizen the power to delete certain assertion records (e.g. criminal and debt assertion records) that can conflict with “to be responsible for our own acts”. This is not just a technical problem, and, for a practical system to survive, this rule must be aligned with the technical solutions.

Bibliography

- [1] Ana Gomes, Inês Santos, Maria Albuquerque, and Paulo Pereira. “Consumption of Radiological Exams in the Continental Portugal District Hospitals between 2002 and 2006-analysis from the available on-line data by the ACSS Authority”. In: *Revista Lusófona de Ciências e Tecnologias da Saúde* 2 (2011) (cit. on p. 1).
- [2] Rebecca Smith-Bindman, Jafi Lipson, Ralph Marcus, Kwang-Pyo Kim, Mahadevappa Mahesh, Robert Gould, Amy Berrington De González, and Diana L Miglioretti. “Radiation dose associated with common computed tomography examinations and the associated lifetime attributable risk of cancer”. In: *Archives of internal medicine* 169.22 (2009), pp. 2078–2086 (cit. on p. 1).
- [3] Diana L Miglioretti, Eric Johnson, Andrew Williams, Robert T Greenlee, Sheila Weinmann, Leif I Solberg, Heather Spencer Feigelson, Douglas Roblin, Michael J Flynn, Nicholas Vanneman, et al. “The use of computed tomography in pediatrics and the associated radiation exposure and estimated cancer risk”. In: *JAMA pediatrics* 167.8 (2013), pp. 700–707 (cit. on p. 1).
- [4] Kyoung Ho Lee, Hak Jong Lee, Jae Hyoung Kim, Heung Sik Kang, Kyung Won Lee, Helen Hong, Ho Jun Chin, and Kyoo Seob Ha. “Managing the CT data explosion: initial experiences of archiving volumetric datasets in a mini-PACS”. In: *Journal of digital imaging* 18.3 (2005), pp. 188–195 (cit. on p. 1).
- [5] David C Goodman and Elliott S Fisher. “Physician workforce crisis? Wrong diagnosis, wrong prescription”. In: *New England Journal of Medicine* 358.16 (2008), pp. 1658–1661 (cit. on p. 1).
- [6] Reuben Binns, Max Van Kleek, Michael Veale, Ulrik Lyngs, Jun Zhao, and Nigel Shadbolt. “It’s Reducing a Human Being to a Percentage’: Perceptions of Justice in Algorithmic Decisions”. In: *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems*. ACM. 2018, p. 377 (cit. on p. 1).
- [7] James G Hodge. “Ethical issues concerning genetic testing and screening in public health”. In: *American Journal of Medical Genetics Part C: Seminars in Medical Genetics*. Vol. 125. 1. Wiley Online Library. 2004, pp. 66–70 (cit. on p. 2).
- [8] Anna Ferreira, Ricardo Cruz-Correia, Luis Antunes, Pedro Farinha, E Oliveira-Palhares, David W Chadwick, and Altamiro Costa-Pereira. “How to break access control in a controlled manner”. In: *Computer-Based Medical Systems, 2006. CBMS 2006. 19th IEEE International Symposium on*. IEEE. 2006, pp. 847–854 (cit. on pp. 2, 34).
- [9] Maged N Kamel Boulos, Dean M Giustini, and Steve Wheeler. “Instagram and WhatsApp in health and healthcare: an overview”. In: *Future Internet* 8.3 (2016), p. 37 (cit. on p. 2).

- [10] Geert Litjens, Thijs Kooi, Babak Ehteshami Bejnordi, Arnaud Arindra Adiyoso Setio, Francesco Ciompi, Mohsen Ghafourian, Jeroen AWM van der Laak, Bram van Ginneken, and Clara I Sánchez. “A survey on deep learning in medical image analysis”. In: *Medical image analysis* 42 (2017), pp. 60–88 (cit. on pp. 2, 49).
- [11] Adel Kermi, Sofia Marniche-Kermi, and Mohamed Tayeb Laskri. “3D-Computerized facial reconstructions from 3D-MRI of human heads using deformable model approach”. In: *Machine and Web Intelligence (ICMWI), 2010 International Conference on*. IEEE. 2010, pp. 276–282 (cit. on pp. 3, 27).
- [12] Jorge Miguel Silva, Eduardo Pinho, Eriksson Monteiro, João Figueira Silva, and Carlos Costa. “Controlled searching in reversibly de-identified medical imaging archives”. In: *Journal of biomedical informatics* 77 (2018), pp. 81–90 (cit. on pp. 3, 27).
- [13] Latanya Sweeney, Akua Abu, and Julia Winn. “Identifying Participants in the Personal Genome Project by Name”. In: (2013) (cit. on pp. 3, 27).
- [14] Arvind Narayanan and Vitaly Shmatikov. “Robust de-anonymization of large sparse datasets”. In: *Security and Privacy, 2008. SP 2008. IEEE Symposium on*. IEEE. 2008, pp. 111–125 (cit. on pp. 3, 27).
- [15] Rebecca Skloot and Bahni Turpin. *The immortal life of Henrietta Lacks*. Crown Publishers New York, 2010 (cit. on p. 3).
- [16] E Kihlström. “Infection of HeLa cells with Salmonella typhimurium 395 MS and MR10 bacteria.” In: *Infection and immunity* 17.2 (1977), pp. 290–295 (cit. on p. 3).
- [17] William F Scherer, Jerome T Syverton, and George O Gey. “Studies on the propagation in vitro of poliomyelitis viruses: IV. Viral multiplication in a stable strain of human malignant epithelial cells (strain HeLa) derived from an epidermoid carcinoma of the cervix”. In: *Journal of Experimental Medicine* 97.5 (1953), pp. 695–710 (cit. on p. 3).
- [18] Harald Zur Hausen. “Papillomaviruses and cancer: from basic studies to clinical application”. In: *Nature reviews cancer* 2.5 (2002), p. 342 (cit. on p. 3).
- [19] Ewen Callaway. “HeLa publication brews bioethical storm”. In: *Nature. News*. Available at: <http://www.nature.com/news/hela-publication-brews-bioethical-storm-1.12689>. Accessed: June 27 (2013), p. 2013 (cit. on p. 3).
- [20] Serge PJM Horbach and Willem Halffman. “The ghosts of HeLa: How cell line misidentification contaminates the scientific literature”. In: *PloS one* 12.10 (2017), e0186281 (cit. on p. 3).
- [21] Ken Jordan, Jan Hauser, and Steven Foster. “The Augmented Social Network: Building identity and trust into the next-generation Internet”. In: *First Monday* 8.8 (2003) (cit. on p. 10).
- [22] Kim-Kwang Raymond Choo. “Issue report on business adoption of Microsoft Passport”. In: *Information management & computer security* 14.3 (2006), pp. 218–234 (cit. on p. 10).
- [23] Xueping Liang, Sachin Shetty, Juan Zhao, Daniel Bowden, Danyi Li, and Jihong Liu. “Towards Decentralized Accountability and Self-sovereignty in Healthcare Systems”. In: *International Conference on Information and Communications Security*. Springer. 2017, pp. 387–398 (cit. on p. 12).

- [24] Isabel Cerqueira, Vítor J Sá, and Sérgio Tenreiro de Magalhães. “Study of the Perception on the Portuguese Citizen Card and Electronic Signature”. In: *Global Security, Safety and Sustainability & e-Democracy*. Springer, 2011, pp. 164–170 (cit. on p. 12).
- [25] Ricardo Santos, Manuel E Correia, and Luis Antunes. “Securing a health information system with a government issued digital identification card”. In: *2008 42nd Annual IEEE International Carnahan Conference on Security Technology*. IEEE. 2008, pp. 135–141 (cit. on pp. 12, 97).
- [26] Frank Pimenta, Cláudio Teixeira, and Joaquim Sousa Pinto. “GlobaliD: Federated identity provider associated with national citizen’s card”. In: *5th Iberian Conference on Information Systems and Technologies*. IEEE. 2010, pp. 1–6 (cit. on p. 13).
- [27] Leslie Lamport et al. “Paxos made simple”. In: *ACM Sigact News* 32.4 (2001), pp. 18–25 (cit. on p. 13).
- [28] Diego Ongaro and John K Ousterhout. “In search of an understandable consensus algorithm.” In: *USENIX Annual Technical Conference*. 2014, pp. 305–319 (cit. on p. 13).
- [29] Ben Temkow, A-M Bosneag, Xinjie Li, and Monica Brockmeyer. “PaxonDHT: Achieving consensus in distributed hash tables”. In: *Applications and the Internet, 2006. SAINT 2006. International Symposium on*. IEEE. 2006, 9–pp (cit. on p. 13).
- [30] Tushar D Chandra, Robert Griesemer, and Joshua Redstone. “Paxos made live: an engineering perspective”. In: *Proceedings of the twenty-sixth annual ACM symposium on Principles of distributed computing*. ACM. 2007, pp. 398–407 (cit. on p. 13).
- [31] Leslie Lamport, Robert Shostak, and Marshall Pease. “The Byzantine generals problem”. In: *ACM Transactions on Programming Languages and Systems (TOPLAS)* 4.3 (1982), pp. 382–401 (cit. on p. 13).
- [32] Michael J Fischer, Nancy A Lynch, and Michael S Paterson. “Impossibility of distributed consensus with one faulty process”. In: *Journal of the ACM (JACM)* 32.2 (1985), pp. 374–382 (cit. on p. 13).
- [33] John R Douceur. “The sybil attack”. In: *International workshop on peer-to-peer systems*. Springer. 2002, pp. 251–260 (cit. on p. 13).
- [34] Loi Luu, Yaron Velner, Jason Teutsch, and Prateek Saxena. “SMART POOL: Practical Decentralized Pooled Mining.” In: *IACR Cryptology ePrint Archive 2017* (2017), p. 19 (cit. on pp. 16, 38).
- [35] Adi Shamir. “How to share a secret”. In: *Communications of the ACM* 22.11 (1979), pp. 612–613 (cit. on pp. 17, 20, 30, 37, 46).
- [36] Cynthia Dwork and Moni Naor. “Pricing via processing or combatting junk mail”. In: *Annual International Cryptology Conference*. Springer. 1992, pp. 139–147 (cit. on p. 17).
- [37] Steven D Galbraith, Kenneth G Paterson, and Nigel P Smart. “Pairings for cryptographers”. In: *Discrete Applied Mathematics* 156.16 (2008), pp. 3113–3121 (cit. on pp. 18, 97).
- [38] Osmanbey Uzunkol and Mehmet Sabır Kiraz. “Still wrong use of pairings in cryptography”. In: *Applied Mathematics and Computation* 333 (2018), pp. 467–479 (cit. on p. 18).

- [39] Amos Fiat and Adi Shamir. “How to prove yourself: Practical solutions to identification and signature problems”. In: *Conference on the Theory and Application of Cryptographic Techniques*. Springer. 1986, pp. 186–194 (cit. on p. 19).
- [40] Paul Feldman. “A practical scheme for non-interactive verifiable secret sharing”. In: *28th Annual Symposium on Foundations of Computer Science (sfcs 1987)*. IEEE. 1987, pp. 427–438 (cit. on p. 22).
- [41] Thomas Ristenpart and Scott Yilek. “The power of proofs-of-possession: Securing multiparty signatures against rogue-key attacks”. In: *Annual International Conference on the Theory and Applications of Cryptographic Techniques*. Springer. 2007, pp. 228–245 (cit. on pp. 22, 100, 113).
- [42] Alexander Mense and Bernd Blobel. “HL7 Standards and Components to Support Implementation of the European General Data Protection Regulation”. In: *European Journal for Biomedical Informatics* 13.1 (2017), pp. 27–33 (cit. on p. 25).
- [43] European Society of Radiology (ESR et al. “The new EU General Data Protection Regulation: what the radiologist should know”. In: *Insights into imaging* 8.3 (2017), pp. 295–299 (cit. on p. 25).
- [44] John Mark Michael Rumbold and Barbara Pierscionek. “The effect of the General Data Protection Regulation on medical research”. In: *Journal of medical Internet research* 19.2 (2017) (cit. on p. 25).
- [45] Luís A Bastião Silva, Carlos Costa, and José Luis Oliveira. “A PACS archive architecture supported on cloud services”. In: *International journal of computer assisted radiology and surgery* 7.3 (2012), pp. 349–358 (cit. on pp. 25, 31).
- [46] Consulting Intersoft. *Processing of special categories of personal data*. 2018. URL: <https://gdpr-info.eu/art-9-gdpr/> (visited on Mar. 7, 2018) (cit. on p. 25).
- [47] Paul N Valenstein, Stephen S Raab, and Molly K Walsh. “Identification errors involving clinical laboratories: a College of American Pathologists Q-Probes study of patient and specimen identification errors at 120 institutions”. In: *Archives of pathology & laboratory medicine* 130.8 (2006), pp. 1106–1113 (cit. on p. 26).
- [48] Andrew Tobin and Drummond Reed. “The Inevitable Rise of Self-Sovereign Identity”. In: *The Sovrin Foundation* (2016) (cit. on pp. 27, 29, 46).
- [49] Michael Veale and Lilian Edwards. “Clarity, surprises, and further questions in the Article 29 Working Party draft guidance on automated decision-making and profiling”. In: *Computer Law & Security Review* 34.2 (2018), pp. 398–404 (cit. on p. 28).
- [50] Gianclaudio Malgieri and Giovanni Comandé. “Why a right to legibility of automated decision-making exists in the general data protection regulation”. In: *International Data Privacy Law* (2017) (cit. on p. 28).
- [51] Pedro Costa, Adrian Galdran, Asim Smailagic, and Aurélio Campilho. “A Weakly-Supervised Framework for Interpretable Diabetic Retinopathy Detection on Retinal Images”. In: *IEEE Access* 6 (2018), pp. 18747–18758 (cit. on p. 28).
- [52] Chris Olah, Arvind Satyanarayan, Ian Johnson, Shan Carter, Ludwig Schubert, Katherine Ye, and Alexander Mordvintsev. “The building blocks of interpretability”. In: *Distill* 3.3 (2018), e10 (cit. on p. 28).

- [53] Raja Parasuraman and Dietrich H Manzey. “Complacency and bias in human use of automation: An attentional integration”. In: *Human factors* 52.3 (2010), pp. 381–410 (cit. on p. 28).
- [54] Christopher D Wickens, Benjamin A Clegg, Alex Z Vieane, and Angelia L Sebok. “Complacency and automation bias in the use of imperfect automation”. In: *Human factors* 57.5 (2015), pp. 728–739 (cit. on p. 28).
- [55] David Lyell, Farah Magrabi, Magdalena Z Raban, LG Pont, Melissa T Baysari, Richard O Day, and Enrico Coiera. “Automation bias in electronic prescribing”. In: *BMC medical informatics and decision making* 17.1 (2017), p. 28 (cit. on p. 28).
- [56] Golnaz Elahi, Zeev Lieber, and Eric Yu. “Trade-off analysis of identity management systems with an untrusted identity provider”. In: *Computer Software and Applications, 2008. COMPSAC’08. 32nd Annual IEEE International*. IEEE. 2008, pp. 661–666 (cit. on p. 28).
- [57] Uciel Fragoso-Rodriguez, Maryline Laurent-Maknavicius, and José Incera-Dieguez. “Federated identity architectures”. In: *Proc. 1st Mexican Conference on Informatics Security 2006 (MCIS’2006)*. 2006 (cit. on p. 29).
- [58] Yanli Ren, Shuozhong Wang, Xinpeng Zhang, and Zhenxing Qian. “Fully secure anonymous identity-based encryption under simple assumptions”. In: *Multimedia Information Networking and Security (MINES), 2010 International Conference on*. IEEE. 2010, pp. 428–432 (cit. on p. 29).
- [59] Civic Technologies. *Civic White Paper*. 2017. URL: <https://tokensale.civic.com/CivicTokenSaleWhitePaper.pdf> (visited on Apr. 25, 2018) (cit. on p. 29, 46).
- [60] Christian Lundkvist, Rouven Heck, Joel Torstensson, Zac Mitton, and Michael Sena. *Uport: A platform for self-sovereign identity*. 2017. URL: https://whitepaper.uport.me/uPort_whitepaper_DRAFT20170221.pdf (visited on Apr. 20, 2018) (cit. on p. 29).
- [61] Germano Caronni. “Walking the web of trust”. In: *Enabling Technologies: Infrastructure for Collaborative Enterprises, 2000. (WET ICE 2000). Proceedings. IEEE 9th International Workshops on*. IEEE. 2000, pp. 153–158 (cit. on p. 30).
- [62] Ruggero Morselli, Bobby Bhattacharjee, Jonathan Katz, and Michael Marsh. *Keychains: A decentralized public-key infrastructure*. Tech. rep. University of Maryland, College Park College Park United States, 2006 (cit. on p. 30).
- [63] Douglas R. Stinson. “An explication of secret sharing schemes”. In: *Designs, Codes and Cryptography* 2.4 (1992), pp. 357–390 (cit. on p. 30).
- [64] Adebayo Omotosho and Justice Emuoyibofarhe. “A criticism of the current security, privacy and accountability issues in electronic health records”. In: *arXiv preprint arXiv:1501.07865* (2015) (cit. on p. 30).
- [65] Longhua Zhang, Gail-Joon Ahn, and Bei-Tseng Chu. “A role-based delegation framework for healthcare information systems”. In: *Proceedings of the seventh ACM symposium on Access control models and technologies*. ACM. 2002, pp. 125–134 (cit. on p. 30).

- [66] Josh Benaloh, Melissa Chase, Eric Horvitz, and Kristin Lauter. “Patient controlled encryption: ensuring privacy of electronic medical records”. In: *Proceedings of the 2009 ACM workshop on Cloud computing security*. ACM. 2009, pp. 103–114 (cit. on p. 30).
- [67] Bernhard Riedl, Veronika Grascher, Stefan Fenz, and Thomas Neubauer. “Pseudonymization for improving the privacy in e-health applications”. In: *Hawaii International Conference on System Sciences, Proceedings of the 41st Annual*. IEEE. 2008, pp. 255–255 (cit. on pp. 30, 33).
- [68] Marci Meingast, Tanya Roosta, and Shankar Sastry. “Security and privacy issues with health care information technology”. In: *Engineering in Medicine and Biology Society, 2006. EMBS’06. 28th Annual International Conference of the IEEE*. IEEE. 2006, pp. 5453–5458 (cit. on p. 30).
- [69] Marlon Cordeiro Domenech, Eros Comunello, and Michelle Silva Wangham. “Identity management in e-Health: A case study of web of things application using OpenID connect”. In: *e-Health Networking, Applications and Services (Healthcom), 2014 IEEE 16th International Conference on*. IEEE. 2014, pp. 219–224 (cit. on p. 30).
- [70] Marcelo Antonio de Carvalho Junior and Paulo Bandiera-Paiva. “Health information system role-based access control current security trends and challenges”. In: *Journal of healthcare engineering 2018 (2018)* (cit. on p. 31).
- [71] Lingfeng Chen and Doan B Hoang. “Novel data protection model in healthcare cloud”. In: *2011 IEEE International Conference on High Performance Computing and Communications*. IEEE. 2011, pp. 550–555 (cit. on p. 31).
- [72] M Fahim Ferdous Khan and Ken Sakamura. “A secure and flexible e-health access control system with provisions for emergency access overrides and delegation of access privileges”. In: *2016 18th International Conference on Advanced Communication Technology (ICACT)*. IEEE. 2016, pp. 541–546 (cit. on p. 31).
- [73] Dong-Hee Shin, Jaemin Jung, and Byeng-Hee Chang. “The psychology behind QR codes: User experience perspective”. In: *Computers in Human Behavior* 28.4 (2012), pp. 1417–1426 (cit. on p. 31).
- [74] Riccardo Focardi, Flaminia L Luccio, and Heider AM Wahsheh. “Usable cryptographic QR codes”. In: *2018 IEEE International Conference on Industrial Technology (ICIT)*. IEEE. 2018, pp. 1664–1669 (cit. on p. 31).
- [75] Katharina Krombholz, Peter Frühwirt, Thomas Rieder, Ioannis Kapsalis, Johanna Ullrich, and Edgar Weippl. “QR Code Security—How Secure and Usable Apps Can Protect Users Against Malicious QR Codes”. In: *2015 10th International Conference on Availability, Reliability and Security*. IEEE. 2015, pp. 230–237 (cit. on p. 31).
- [76] Michael Jones and Dick Hardt. *The oauth 2.0 authorization framework: Bearer token usage*. Tech. rep. RFC 6750, October, 2012 (cit. on p. 31).
- [77] Victor Sucasas, Georgios Mantas, Ayman Radwan, and Jonathan Rodriguez. “An OAuth2-based protocol with strong user privacy preservation for smart city mobile e-Health apps”. In: *2016 IEEE International Conference on Communications (ICC)*. IEEE. 2016, pp. 1–6 (cit. on p. 31).

- [78] Victor Sucasas, Georgios Mantas, Ayman Radwan, and Jonathan Rodriguez. “A lightweight privacy-preserving OAuth2-based protocol for smart city mobile apps”. In: *2016 IEEE Globecom Workshops (GC Wkshps)*. IEEE. 2016, pp. 1–6 (cit. on p. 31).
- [79] Rohitash Kumar Banyal, Pragya Jain, and Vijendra Kumar Jain. “Multi-factor authentication framework for cloud computing”. In: *2013 Fifth International Conference on Computational Intelligence, Modelling and Simulation*. IEEE. 2013, pp. 105–110 (cit. on p. 32).
- [80] Zeeshan Siddiqui, Abdul Hanan Abdullah, Muhammad Khurram Khan, and Abdullah S Alghamdi. “Smart environment as a service: three factor cloud based user authentication for telecare medical information system”. In: *Journal of medical systems* 38.1 (2014), p. 9997 (cit. on p. 32).
- [81] Abhilasha Bhargav-Spantzel, Anna C Squicciarini, Shimon Modi, Matthew Young, Elisa Bertino, and Stephen J Elliott. “Privacy preserving multi-factor authentication with biometrics”. In: *Journal of Computer Security* 15.5 (2007), pp. 529–560 (cit. on p. 32).
- [82] Wayne Newhauser, Timothy Jones, Stuart Swerdloff, Warren Newhauser, Mark Cilia, Robert Carver, Andy Halloran, and Rui Zhang. “Anonymization of DICOM electronic medical records for radiation therapy”. In: *Computers in biology and medicine* 53 (2014), pp. 134–140 (cit. on p. 32).
- [83] Ronald L Rivest, Adi Shamir, and Yael Tauman. “How to leak a secret”. In: *International Conference on the Theory and Application of Cryptology and Information Security*. Springer. 2001, pp. 552–565 (cit. on p. 32).
- [84] Jian Ren and Lien Harn. “Generalized ring signatures”. In: *IEEE Transactions on Dependable and Secure Computing* 5.3 (2008), pp. 155–163 (cit. on p. 32).
- [85] Klaus Pommerening and Michael Reng. “Secondary use of the EHR via pseudonymisation”. In: *Studies in health technology and informatics* (2004), pp. 441–446 (cit. on p. 33).
- [86] Benjamin Fabian, Tatiana Ermakova, and Philipp Junghanns. “Collaborative and secure sharing of healthcare data in multi-clouds”. In: *Information Systems* 48 (2015), pp. 132–150 (cit. on p. 33).
- [87] Rita Noumeir, Alain Lemay, and Jean-Marc Lina. “Pseudonymization of radiology data for research purposes”. In: *Journal of digital imaging* 20.3 (2007), pp. 284–295 (cit. on p. 33).
- [88] Yang Yang, Ximeng Liu, and Robert H Deng. “Lightweight break-glass access control system for healthcare internet-of-things”. In: *IEEE Transactions on Industrial Informatics* 14.8 (2017), pp. 3610–3617 (cit. on pp. 33, 34).
- [89] Helmut Petritsch and Achim D Brucker. “Extending access control models with break-glass”. In: *Proceedings of the 14th ACM Symposium on Access Control Models and Technologies*. ACM, 2009, pp. 197–206 (cit. on pp. 33, 34).
- [90] Tiago Pedrosa, Rui Pedro Lopes, Joao C Santos, Carlos Costa, and José Luís Oliveira. “Towards an EHR architecture for mobile citizens”. In: *HealthInf 2010* (2010), pp. 288–293 (cit. on p. 33).

- [91] Lillian Rostad and Ole Edsberg. “A study of access control requirements for healthcare systems based on audit trails from access logs”. In: *2006 22nd Annual Computer Security Applications Conference (ACSAC’06)*. IEEE. 2006, pp. 175–186 (cit. on pp. 33, 34).
- [92] Bernhard Riedl, Thomas Neubauer, Gernot Goluch, Oswald Boehm, Gert Reinauer, and Alexander Krumböck. “A secure architecture for the pseudonymization of medical data”. In: *The Second International Conference on Availability, Reliability and Security (ARES’07)*. IEEE. 2007, pp. 318–324 (cit. on p. 33).
- [93] Harald Aamot, Christian Dominik Kohl, Daniela Richter, and Petra Knaup-Gregori. “Pseudonymization of patient identifiers for translational research”. In: *BMC medical informatics and decision making* 13.1 (2013), p. 75 (cit. on p. 33).
- [94] Taha Belkhouja, Xiaojiang Du, Amr Mohamed, Abdulla K Al-Ali, and Mohsen Guizani. “Biometric-based authentication scheme for Implantable Medical Devices during emergency situations”. In: *Future Generation Computer Systems* 98 (2019), pp. 109–119 (cit. on p. 34).
- [95] Mark D Ryan. “Cloud computing security: The scientific challenge, and a survey of solutions”. In: *Journal of Systems and Software* 86.9 (2013), pp. 2263–2268 (cit. on p. 34).
- [96] Srdjan Marinovic, Robert Craven, Jiefei Ma, and Naranker Dulay. “Rumpole: a flexible break-glass access control model”. In: *Proceedings of the 16th ACM symposium on Access control models and technologies*. ACM. 2011, pp. 73–82 (cit. on p. 34).
- [97] Ana Ferreira, David Chadwick, Pedro Farinha, Ricardo Correia, Gansen Zao, Rui Chilro, and Luis Antunes. “How to securely break into RBAC: the BTG-RBAC model”. In: *2009 Annual Computer Security Applications Conference*. IEEE. 2009, pp. 23–31 (cit. on p. 34).
- [98] Kin Suntana Tep, Ben Martini, Ray Hunt, and Kim-Kwang Raymond Choo. “A taxonomy of cloud attack consequences and mitigation strategies: The Role of Access Control and Privileged Access Management”. In: *2015 IEEE Trustcom/BigDataSE/ISPA*. Vol. 1. IEEE. 2015, pp. 1073–1080 (cit. on p. 34).
- [99] Claudio Agostino Ardagna, Sabrina De Capitani di Vimercati, Tyrone Grandison, Sushil Jajodia, and Pierangela Samarati. “Regulating exceptions in healthcare using policy spaces”. In: *IFIP Annual Conference on Data and Applications Security and Privacy*. Springer. 2008, pp. 254–267 (cit. on p. 34).
- [100] Arya Adriansyah, Boudewijn F Van Dongen, and Nicola Zannone. “Controlling break-the-glass through alignment”. In: *2013 International Conference on Social Computing*. IEEE. 2013, pp. 606–611 (cit. on p. 34).
- [101] C Lakshmi, Karuppusamy Thenmozhi, John Bosco Balaguru Rayappan, and Rengarajan Amirtharajan. “Encryption and watermark-treated medical image against hacking disease—An immune convention in spatial and frequency domains”. In: *Computer methods and programs in biomedicine* 159 (2018), pp. 11–21 (cit. on p. 35).
- [102] Dan Boneh, Giovanni Di Crescenzo, Rafail Ostrovsky, and Giuseppe Persiano. “Public key encryption with keyword search”. In: *International conference on the theory and applications of cryptographic techniques*. Springer. 2004, pp. 506–522 (cit. on p. 35).

- [103] Raluca Ada Popa, Catherine MS Redfield, Nickolai Zeldovich, and Hari Balakrishnan. “CryptDB: protecting confidentiality with encrypted query processing”. In: *Proceedings of the Twenty-Third ACM Symposium on Operating Systems Principles*. 2011, pp. 85–100 (cit. on p. 35).
- [104] Georgios Kellaris, George Kollios, Kobbi Nissim, and Adam O’neill. “Generic attacks on secure outsourced databases”. In: *Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security*. ACM. 2016, pp. 1329–1340 (cit. on p. 35).
- [105] Paul Grubbs, Marie-Sarah Lacharité, Brice Minaud, and Kenneth G Paterson. “Learning to Reconstruct: Statistical Learning Theory and Encrypted Database Attacks”. In: *IEEE Symposium on Security and Privacy (S&P) 2019*. 2019, pp. 1067–1083 (cit. on p. 35).
- [106] Sha Ma, Yi Mu, Willy Susilo, and Bo Yang. “Witness-based searchable encryption”. In: *Information Sciences* 453 (2018), pp. 364–378 (cit. on p. 35).
- [107] Yu-Chi Chen, Xin Xie, Peter Shaojui Wang, and Raylin Tso. “Witness-based searchable encryption with optimal overhead for cloud-edge computing”. In: *Future Generation Computer Systems* 100 (2019), pp. 715–723 (cit. on p. 35).
- [108] Tom Doel, Dzhoshkun I Shakir, Rosalind Pratt, Michael Aertsen, James Moggridge, Erwin Bellon, Anna L David, Jan Deprest, Tom Vercauteren, and Sébastien Ourselin. “GIFT-Cloud: A data sharing and collaboration platform for medical imaging research”. In: *computer methods and programs in biomedicine* 139 (2017), pp. 181–190 (cit. on p. 35).
- [109] Ross Anderson, Steven M Bellovin, Josh Benaloh, Matt Blaze, Whitfield Diffie, John Gilmore, Peter G Neumann, Bruce Schneier, Harold Abelson, Ronald L Rivest, et al. *The risks of key recovery, key escrow, and trusted third-party encryption*. 1997 (cit. on p. 35).
- [110] Jinyuan Sun, Xiaoyan Zhu, Chi Zhang, and Yuguang Fang. “HCPP: Cryptography based secure EHR system for patient privacy and emergency healthcare”. In: *2011 31st International Conference on Distributed Computing Systems*. IEEE. 2011, pp. 373–382 (cit. on p. 35).
- [111] Yue Tong, Jinyuan Sun, Sherman SM Chow, and Pan Li. “Cloud-assisted mobile-access of health data with privacy and auditability”. In: *IEEE Journal of biomedical and health Informatics* 18.2 (2013), pp. 419–429 (cit. on p. 35).
- [112] Yao-Jen Chang, Wende Zhang, and Tsuhan Chen. “Biometrics-based cryptographic key generation”. In: *Multimedia and Expo, 2004. ICME’04. 2004 IEEE International Conference on*. Vol. 3. IEEE. 2004, pp. 2203–2206 (cit. on p. 35).
- [113] Virginia Ruiz-Albacete, Pedro Tome-Gonzalez, Fernando Alonso-Fernandez, Javier Galbally, Julian Fierrez, and Javier Ortega-Garcia. “Direct attacks using fake images in iris verification”. In: *European Workshop on Biometrics and Identity Management*. Springer. 2008, pp. 181–190 (cit. on p. 35).
- [114] Abdenour Hadid. “Face biometrics under spoofing attacks: Vulnerabilities, countermeasures, open issues, and research directions”. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*. 2014, pp. 113–118 (cit. on p. 35).

- [115] André Anjos and Sébastien Marcel. “Counter-measures to photo attacks in face recognition: a public database and a baseline”. In: *Biometrics (IJCB), 2011 international joint conference on*. IEEE. 2011, pp. 1–7 (cit. on p. 35).
- [116] Danan Thilakanathan, Rafael A Calvo, Shiping Chen, Surya Nepal, and Nick Glozier. “Facilitating secure sharing of personal health data in the cloud”. In: *JMIR medical informatics* 4.2 (2016), e15 (cit. on p. 35).
- [117] Wei-Bin Lee and Chien-Ding Lee. “A cryptographic key management solution for HIPAA privacy/security regulations”. In: *IEEE Transactions on Information Technology in Biomedicine* 12.1 (2008), pp. 34–41 (cit. on p. 35).
- [118] Danan Thilakanathan, Shiping Chen, Surya Nepal, Rafael Calvo, and Leila Alem. “A platform for secure monitoring and sharing of generic health data in the Cloud”. In: *Future Generation Computer Systems* 35 (2014), pp. 102–113 (cit. on p. 35).
- [119] Manas Ranjan Patra, Rama Krushna Das, and Rabi Prasad Padhy. “CRHIS: cloud based rural healthcare information system”. In: *Proceedings of the 6th International Conference on Theory and Practice of Electronic Governance*. ACM. 2012, pp. 402–405 (cit. on p. 36).
- [120] Carlos Oberdan Rolim, Fernando Luiz Koch, Carlos Becker Westphall, Jorge Werner, Armando Fracalossi, and Giovanni Schmitt Salvador. “A cloud computing solution for patient’s data collection in health care institutions”. In: *eHealth, Telemedicine, and Social Medicine, 2010. ETELEMED’10. Second International Conference on*. IEEE. 2010, pp. 95–99 (cit. on p. 36).
- [121] Matthias Mettler. “Blockchain technology in healthcare: The revolution starts here”. In: *e-Health Networking, Applications and Services (Healthcom), 2016 IEEE 18th International Conference on*. IEEE. 2016, pp. 1–3 (cit. on p. 36).
- [122] Xiao Yue, Huiju Wang, Dawei Jin, Mingqiang Li, and Wei Jiang. “Healthcare data gateways: found healthcare intelligence on blockchain with novel privacy risk control”. In: *Journal of medical systems* 40.10 (2016), p. 218 (cit. on p. 36).
- [123] Drew Ivan. “Moving toward a blockchain-based method for the secure storage of patient records”. In: *ONC/NIST Use of Blockchain for Healthcare and Research Workshop. Gaithersburg, Maryland, United States: ONC/NIST*. 2016 (cit. on p. 36).
- [124] Guy Zyskind, Oz Nathan, et al. “Decentralizing privacy: Using blockchain to protect personal data”. In: *Security and Privacy Workshops (SPW), 2015 IEEE*. IEEE. 2015, pp. 180–184 (cit. on p. 36).
- [125] DS Baars. “Towards self-sovereign identity using blockchain technology”. MA thesis. University of Twente, 2016 (cit. on p. 36).
- [126] Ori Jacobovitz. *Blockchain for identity management*. 2016 (cit. on p. 36).
- [127] Benedikt Herudek. “INTEGRATION–THE BLOCKCHAIN ‘KILLER USECASE’–PART I”. In: *Service Technology Magazine* (2015) (cit. on p. 36).
- [128] LM Axon and Michael Goldsmith. “PB-PKI: a privacy-aware blockchain-based PKI”. In: (2016) (cit. on p. 36).
- [129] Martin Steinert and Larry Leifer. “Scrutinizing Gartner’s hype cycle approach”. In: *Technology Management for Global Economic Growth (PICMET), 2010 Proceedings of PICMET’10*: IEEE. 2010, pp. 1–13 (cit. on p. 36).

- [130] Karl Wüst and Arthur Gervais. “Do you need a Blockchain?” In: *IACR Cryptology ePrint Archive* 2017 (2017), p. 375 (cit. on p. 36).
- [131] Tsung-Ting Kuo, Hyeon-Eui Kim, and Lucila Ohno-Machado. “Blockchain distributed ledger technologies for biomedical and health care applications”. In: *Journal of the American Medical Informatics Association* 24.6 (2017), pp. 1211–1220 (cit. on p. 36).
- [132] Leemon Baird. *Hashgraph consensus: fair, fast, byzantine fault tolerance*. Tech. rep. Swirls Tech Report, 2016 (cit. on p. 36).
- [133] Petar Maymounkov and David Mazieres. “Kademlia: A peer-to-peer information system based on the xor metric”. In: *International Workshop on Peer-to-Peer Systems*. Springer. 2002, pp. 53–65 (cit. on p. 37).
- [134] Damon McCoy, Kevin Bauer, Dirk Grunwald, Tadayoshi Kohno, and Douglas Sicker. “Shining light in dark places: Understanding the Tor network”. In: *International Symposium on Privacy Enhancing Technologies Symposium*. Springer. 2008, pp. 63–76 (cit. on pp. 37, 72).
- [135] Tim Swanson. “Consensus-as-a-service: a brief report on the emergence of permissioned, distributed ledger systems”. In: *Report, available online, Apr* (2015) (cit. on p. 38).
- [136] Cynthia Dwork, Adam Smith, Thomas Steinke, and Jonathan Ullman. “Exposed! a survey of attacks on private data”. In: *Annual Review of Statistics and Its Application* 4 (2017), pp. 61–84 (cit. on p. 38).
- [137] Stefan Schulz and Catalina Martinez-Costa. “How ontologies can improve semantic interoperability in health care”. In: *Process Support and Knowledge Representation in Health Care*. Springer, 2013, pp. 1–10 (cit. on p. 38).
- [138] Vipul Kashyap, Christoph Bussler, and Matthew Moran. *The semantic web: semantics for data and services on the web*. Springer Science & Business Media, 2008, pp. 35–36 (cit. on p. 38).
- [139] W Xu, Z Guan, J Sun, Z Wang, and Y Geng. “Development of an open metadata schema for prospective clinical research (openPCR) in China”. In: *Methods of information in medicine* 53.01 (2014), pp. 39–46 (cit. on p. 38).
- [140] Ayako Morozumi, Satomi Nomura, Mitsuharu Nagamori, and Shigeo Sugimoto. “Metadata framework for Manga: A multi-paradigm metadata description framework for digital comics”. In: *International Conference on Dublin Core and Metadata Applications*. 2009, pp. 61–70 (cit. on p. 38).
- [141] Stuart A Sutton. “Conceptual design and deployment of a metadata framework for educational resources on the Internet”. In: *Journal of the Association for Information Science and Technology* 50.13 (1999), p. 1182 (cit. on p. 38).
- [142] Michael Kifer, Georg Lausen, and James Wu. “Logical foundations of object-oriented and frame-based languages”. In: *Journal of the ACM (JACM)* 42.4 (1995), pp. 741–843 (cit. on p. 39).
- [143] Asuman Dogac, Gokce B Laleci, Thomas Aden, and Marco Eichelberg. “Enhancing IHE XDS for federated clinical affinity domain support”. In: *IEEE Transactions on Information Technology in Biomedicine* 11.2 (2007), pp. 213–221 (cit. on p. 40).

- [144] M Elon Gale and Daniel R Gale. “DICOM modality worklist: an essential component in a PACS environment”. In: *Journal of digital imaging* 13.3 (2000), pp. 101–108 (cit. on p. 40).
- [145] Melanie Bettina Späth and Jane Grimson. “Applying the archetype approach to the database of a biobank information management system”. In: *International journal of medical informatics* 80.3 (2011), pp. 205–226 (cit. on p. 41).
- [146] Sebastian Garde, Evelyn Hovenga, Jasmin Buck, and Petra Knaup. “Expressing clinical data sets with openEHR archetypes: a solid basis for ubiquitous computing”. In: *International journal of medical informatics* 76 (2007), S334–S341 (cit. on p. 41).
- [147] Jasmin Buck, Sebastian Garde, Christian D Kohl, and Petra Knaup-Gregori. “Towards a comprehensive electronic patient record to support an innovative individual care concept for premature infants using the openEHR approach”. In: *International journal of medical informatics* 78.8 (2009), pp. 521–531 (cit. on p. 41).
- [148] Cui Tao, Guoqian Jiang, Thomas A Oniki, Robert R Freimuth, Qian Zhu, Deepak Sharma, Jyotishman Pathak, Stanley M Huff, and Christopher G Chute. “A semantic-web oriented representation of the clinical element model for secondary use of electronic health records data”. In: *Journal of the American Medical Informatics Association* 20.3 (2012), pp. 554–562 (cit. on p. 41).
- [149] Cui Tao, Craig G Parker, Thomas A Oniki, Jyotishman Pathak, Stanley M Huff, and Christopher G Chute. “An OWL meta-ontology for representing the clinical element model”. In: *AMIA annual symposium proceedings*. Vol. 2011. American Medical Informatics Association. 2011, p. 1372 (cit. on p. 41).
- [150] James Mayfield and Tim Finin. “Information retrieval on the Semantic Web: Integrating inference and retrieval”. In: *Proceedings of the SIGIR Workshop on the Semantic Web*. 2003 (cit. on p. 41).
- [151] Jonathan M Mortensen, Evan P Minty, Michael Januszyk, Timothy E Sweeney, Alan L Rector, Natalya F Noy, and Mark A Musen. “Using the wisdom of the crowds to find critical errors in biomedical ontologies: a study of SNOMED CT”. In: *Journal of the American Medical Informatics Association* 22.3 (2014), pp. 640–648 (cit. on p. 41).
- [152] Catalina Martínez-Costa, Marcos Menárguez-Tortosa, and Jesualdo Tomás Fernández-Breis. “An approach for the semantic interoperability of ISO EN 13606 and OpenEHR archetypes”. In: *Journal of biomedical informatics* 43.5 (2010), pp. 736–746 (cit. on p. 42).
- [153] Duane Bender and Kamran Sartipi. “HL7 FHIR: An Agile and RESTful approach to healthcare information exchange”. In: *Computer-Based Medical Systems (CBMS), 2013 IEEE 26th International Symposium on*. IEEE. 2013, pp. 326–331 (cit. on p. 42).
- [154] Robert H Dolin, Liora Alschuler, Calvin Beebe, Paul V Biron, Sandra Lee Boyer, Daniel Essin, Elliot Kimber, Tom Lincoln, and John E Mattison. “The HL7 clinical document architecture”. In: *Journal of the American Medical Informatics Association* 8.6 (2001), pp. 552–569 (cit. on p. 42).
- [155] John Barkley. “Comparing simple role based access control models and access control lists”. In: *Proceedings of the second ACM workshop on Role-based access control*. ACM. 1997, pp. 127–132 (cit. on p. 42).
- [156] Aleksi Ritsilä et al. “GraphQL: The API Design Revolution”. In: (2018) (cit. on p. 43).

- [157] Chino Srls. *Chino.io Platform Security: Features and Guarantees*. 2018. URL: <https://chino.io/static/content/Chino.io-Security-White-Paper.pdf> (visited on May 8, 2018) (cit. on p. 43).
- [158] Asaph Azaria, Ariel Ekblaw, Thiago Vieira, and Andrew Lippman. “Medrec: Using blockchain for medical data access and permission management”. In: *Open and Big Data (OBD), International Conference on*. IEEE. 2016, pp. 25–30 (cit. on p. 43).
- [159] Pablo Lamela Seijas, Simon J Thompson, and Darryl McAdams. “Scripting smart contracts for distributed ledger technology.” In: *IACR Cryptology ePrint Archive 2016* (2016), p. 1156 (cit. on pp. 44, 121).
- [160] Raffaele Montella, David Kelly, Wei Xiong, Alison Brizius, Joshua Elliott, Ravi Madduri, Ketan Maheshwari, Cheryl Porter, Peter Vilter, Michael Wilde, et al. “FACE-IT: A science gateway for food security research”. In: *Concurrency and Computation: Practice and Experience* 27.16 (2015), pp. 4423–4436 (cit. on p. 44).
- [161] Eva Sciacca, Marilena Bandieramonte, Ugo Becciani, Alessandro Costa, Mel Krokos, Piero Massimino, Catia Petta, Costantino Pistagna, Simone Riggi, and Fabio Vitello. “Vi-sIVO Science Gateway: a Collaborative Environment for the Astrophysics Community.” In: *IWSG*. 2013 (cit. on p. 44).
- [162] Sandra Gesing, Jens Krüger, Richard Grunzke, Sonja Herres-Pawlis, and Alexander Hoffmann. “Using science gateways for bridging the differences between research infrastructures”. In: *Journal of Grid Computing* 14.4 (2016), pp. 545–557 (cit. on p. 44).
- [163] Jim Basney and Von Welch. “Science gateway security recommendations”. In: *Cluster Computing (CLUSTER), 2013 IEEE International Conference on*. IEEE. 2013, pp. 1–3 (cit. on p. 44).
- [164] Daniel S Marcus, Timothy R Olsen, Mohana Ramaratnam, and Randy L Buckner. “The extensible neuroimaging archive toolkit”. In: *Neuroinformatics* 5.1 (2007), pp. 11–33 (cit. on p. 44).
- [165] Stefan Klein, Erwin Vast, Johan Van Soest, Andre Dekker, Marcel Koek, and Wiro Niessen. “XNAT imaging platform for BioMedBridges and CTMM TraIT”. In: *Journal of clinical bioinformatics*. Vol. 5. 1. BioMed Central. 2015, S18 (cit. on p. 44).
- [166] Daniel S Marcus, Tracy H Wang, Jamie Parker, John G Csernansky, John C Morris, and Randy L Buckner. “Open Access Series of Imaging Studies (OASIS): cross-sectional MRI data in young, middle aged, nondemented, and demented older adults”. In: *Journal of cognitive neuroscience* 19.9 (2007), pp. 1498–1507 (cit. on p. 44).
- [167] Daniel S Marcus, Anthony F Fotenos, John G Csernansky, John C Morris, and Randy L Buckner. “Open access series of imaging studies: longitudinal MRI data in non-demented and demented older adults”. In: *Journal of cognitive neuroscience* 22.12 (2010), pp. 2677–2684 (cit. on p. 44).
- [168] Olaf Sporns, Giulio Tononi, and Rolf Kötter. “The human connectome: a structural description of the human brain”. In: *PLoS computational biology* 1.4 (2005), e42 (cit. on p. 44).
- [169] J Almeida, Ricardo Ribeiro, and José Luis Oliveira. “A modular workflow management framework”. In: *Proceedings of the 11th International Conference on Health Informatics (HealthInf 2018)*. 2018 (cit. on p. 45).

- [170] Luís Bastião Silva, Alina Trifan, and José Luís Oliveira. “MONTRA: An agile architecture for data publishing and discovery”. In: *Computer methods and programs in biomedicine* 160 (2018), pp. 33–42 (cit. on p. 45).
- [171] Enis Afgan, Jeremy Goecks, Dannon Baker, Nate Coraor, Anton Nekrutenko, James Taylor, Galaxy Team, et al. “Galaxy: A gateway to tools in e-Science”. In: *Guide to e-Science*. Springer, 2011, pp. 145–177 (cit. on p. 45).
- [172] Ian Foster, Yong Zhao, Ioan Raicu, and Shiyong Lu. “Cloud computing and grid computing 360-degree compared”. In: *Grid Computing Environments Workshop, 2008. GCE’08*. Ieee. 2008, pp. 1–10 (cit. on p. 45).
- [173] Rosa Sánchez-Guerrero, Florina Almenárez Mendoza, Daniel Díaz-Sánchez, Patricia Arias Cabarcos, and Andrés Marín López. “Collaborative eHealth Meets Security: Privacy-Enhancing Patient Profile Management”. In: *IEEE journal of biomedical and health informatics* 21.6 (2017), pp. 1741–1749 (cit. on p. 47).
- [174] William M Tierney, Sheri A Alpert, Amy Byrket, Kelly Caine, Jeremy C Leventhal, Eric M Meslin, and Peter H Schwartz. “Provider responses to patients controlling access to their electronic health records: a prospective cohort study in primary care”. In: *Journal of general internal medicine* 30.1 (2015), pp. 31–37 (cit. on p. 49).
- [175] Julie E Ruble and Barbara Lom. “Online protocol annotation: a method to enhance undergraduate laboratory research skills”. In: *CBE-Life Sciences Education* 7.3 (2008), pp. 296–301 (cit. on p. 50).
- [176] Andreas Abraham. “Self-Sovereign Identity”. In: (2017) (cit. on p. 50).
- [177] Advait Deshpande, Katherine Stewart, Louise Lepetit, and Salil Gunashekar. “Distributed Ledger Technologies/Blockchain: Challenges, opportunities and the prospects for standards”. In: *Overview report The British Standards Institution (BSI)* (2017) (cit. on p. 50).
- [178] Aniket Kate and Ian Goldberg. “Distributed key generation for the internet”. In: *2009 29th IEEE International Conference on Distributed Computing Systems*. IEEE. 2009, pp. 119–128 (cit. on pp. 55, 77).
- [179] Zhenhua Chen, Shundong Li, Qiong Huang, Jianhua Yan, and Yong Ding. “A Joint Random Secret Sharing Scheme with Public Verifiability.” In: *IJ Network Security* 18.5 (2016), pp. 917–925 (cit. on pp. 55, 77).
- [180] Rosario Gennaro and Steven Goldfeder. “Fast multiparty threshold ecDSA with fast trustless setup”. In: *Proceedings of the 2018 ACM SIGSAC Conference on Computer and Communications Security*. ACM. 2018, pp. 1179–1194 (cit. on pp. 55, 77).
- [181] Adrian Antipa, Daniel Brown, Alfred Menezes, René Struik, and Scott Vanstone. “Validation of elliptic curve public keys”. In: *International Workshop on Public Key Cryptography*. Springer. 2003, pp. 211–223 (cit. on pp. 56, 112).
- [182] Ivan Damgård. “Commitment schemes and zero-knowledge protocols”. In: *School organized by the European Educational Forum*. Springer. 1998, pp. 63–86 (cit. on p. 60).
- [183] John Proos and Christof Zalka. “Shor’s discrete logarithm quantum algorithm for elliptic curves”. In: *arXiv preprint quant-ph/0301141* (2003) (cit. on pp. 64, 122).

- [184] Matthew Amy, Olivia Di Matteo, Vlad Gheorghiu, Michele Mosca, Alex Parent, and John Schanck. “Estimating the cost of generic quantum pre-image attacks on SHA-2 and SHA-3”. In: *International Conference on Selected Areas in Cryptography*. Springer. 2016, pp. 317–337 (cit. on p. 64).
- [185] Daniel J Bernstein, Daira Hopwood, Andreas Hülsing, Tanja Lange, Ruben Niederhagen, Louiza Papachristodoulou, Michael Schneider, Peter Schwabe, and Zooko Wilcox-O’Hearn. “SPHINCS: practical stateless hash-based signatures”. In: *Annual International Conference on the Theory and Applications of Cryptographic Techniques*. Springer. 2015, pp. 368–397 (cit. on p. 64).
- [186] Alan Mislove, Bimal Viswanath, Krishna P Gummadi, and Peter Druschel. “You are who you know: inferring user profiles in online social networks”. In: *Proceedings of the third ACM international conference on Web search and data mining*. ACM. 2010, pp. 251–260 (cit. on p. 66).
- [187] Zhipeng Cai, Zaobo He, Xin Guan, and Yingshu Li. “Collective data-sanitization for preventing sensitive information inference attacks in social networks”. In: *IEEE Transactions on Dependable and Secure Computing* 15.4 (2018), pp. 577–590 (cit. on p. 66).
- [188] Ronald Kainda, Ivan Flechais, and AW Roscoe. “Usability and security of out-of-band channels in secure device pairing protocols”. In: *Proceedings of the 5th Symposium on Usable Privacy and Security*. ACM. 2009, p. 11 (cit. on p. 70).
- [189] Frederico Valente, Luís A Bastião Silva, Tiago Marques Godinho, and Carlos Costa. “Anatomy of an extensible open source PACS”. In: *Journal of digital imaging* 29.3 (2016), pp. 284–296 (cit. on p. 73).
- [190] Rui Lebre, Luís Bastião, and Carlos Costa. “An Accounting Mechanism for Standard Medical Imaging Services”. In: *2019 IEEE 6th Portuguese Meeting on Bioengineering (ENBENG)*. IEEE. 2019, pp. 1–4 (cit. on p. 73).
- [191] E Eugene Schultz. “A framework for understanding and predicting insider attacks”. In: *Computers & Security* 21.6 (2002), pp. 526–531 (cit. on p. 76).
- [192] Sai Krishna Deepak Maram, Fan Zhang, Lun Wang, Andrew Low, Yupeng Zhang, Ari Juels, and Dawn Song. “CHURP: Dynamic-Committee Proactive Secret Sharing”. In: *Proceedings of the 2019 ACM SIGSAC Conference on Computer and Communications Security*. ACM. 2019, pp. 2369–2386 (cit. on p. 77).
- [193] Claude E Shannon. “Communication theory of secrecy systems”. In: *Bell system technical journal* 28.4 (1949), pp. 656–715 (cit. on p. 78).
- [194] V Gayoso Martínez, L Hernández Encinas, and C Sánchez Ávila. “A survey of the elliptic curve integrated encryption scheme”. In: *ratio* 80.1024 (2010), pp. 160–223 (cit. on p. 81).
- [195] Lein Harn and Changlu Lin. “Detection and identification of cheaters in (t, n) secret sharing scheme”. In: *Designs, Codes and Cryptography* 52.1 (2009), pp. 15–24 (cit. on pp. 85, 87, 97, 118).
- [196] Hossein Ghodosi. “Comments on Harn-Lin’s cheating detection scheme”. In: *Designs, Codes and Cryptography* 60.1 (2011), pp. 63–66 (cit. on pp. 85, 97).

- [197] Miguel Castro, Barbara Liskov, et al. “Practical Byzantine fault tolerance”. In: *OSDI*. Vol. 99. 1999, pp. 173–186 (cit. on p. 85).
- [198] Jan Camenisch and Markus Stadler. “Proof systems for general statements about discrete logarithms”. In: *Technical Report/ETH Zurich, Department of Computer Science 260* (1997) (cit. on p. 85).
- [199] David Wagner. “A generalized birthday problem”. In: *Annual International Cryptology Conference*. Springer. 2002, pp. 288–304 (cit. on p. 89).
- [200] Feng Hao and Peter Ryan. “J-PAKE: authenticated key exchange without PKI”. In: *Transactions on computational science XI*. Springer, 2010, pp. 192–206 (cit. on p. 97).
- [201] Alexandre Savaris, Theo Härder, and Aldo von Wangenheim. “DCMDSM: a DICOM decomposed storage model”. In: *Journal of the American Medical Informatics Association 21.5* (2014), pp. 917–924 (cit. on p. 101).
- [202] Dandu Ravi Varma. “Managing DICOM images: Tips and tricks for the radiologist”. In: *The Indian journal of radiology & imaging 22.1* (2012), p. 4 (cit. on p. 101).
- [203] Tiago Marques Godinho, Carlos Viana-Ferreira, Luís A Bastião Silva, and Carlos Costa. “A Routing Mechanism for Cloud Outsourcing of Medical Imaging Repositories”. In: *IEEE journal of biomedical and health informatics 20.1* (2014), pp. 367–375 (cit. on p. 103).
- [204] Rune Thorbek. “Proactive Linear Integer Secret Sharing.” In: *IACR Cryptology ePrint Archive 2009* (2009), p. 183 (cit. on p. 107).
- [205] Paulo SLM Barreto, Ben Lynn, and Michael Scott. “Constructing elliptic curves with prescribed embedding degrees”. In: *International Conference on Security in Communication Networks*. Springer. 2002, pp. 257–267 (cit. on p. 118).
- [206] Timothy J St Cyr. “An overview of healthcare standards”. In: *Southeastcon, 2013 Proceedings of IEEE*. IEEE. 2013, pp. 1–5 (cit. on p. 119).
- [207] Miriam Reisman. “EHRs: The Challenge of Making Electronic Data Usable and Interoperable”. In: *Pharmacy and Therapeutics 42.9* (2017), p. 572 (cit. on p. 119).
- [208] Rachel L Richesson and Jeffrey Krischer. “Data standards in clinical research: gaps, overlaps, challenges and future directions”. In: *Journal of the American Medical Informatics Association 14.6* (2007), pp. 687–696 (cit. on p. 119).
- [209] Abdul Ghafoor Abbasi and Zaheer Khan. “VeidBlock: Verifiable identity using blockchain and ledger in a software defined network”. In: *Companion Proceedings of the 10th International Conference on Utility and Cloud Computing*. ACM. 2017, pp. 173–179 (cit. on p. 119).
- [210] Juha A Mykkänen and Mika P Tuomainen. “An evaluation and selection framework for interoperability standards”. In: *Information and Software Technology 50.3* (2008), pp. 176–197 (cit. on p. 121).
- [211] Primavera De Filippi and Samer Hassan. “Blockchain technology as a regulatory technology: From code is law to law is code”. In: *First Monday 21.12* (2016) (cit. on p. 121).

- [212] Douglas W Arner, János Barberis, and Ross P Buckley. “FinTech and RegTech in a Nutshell, and the Future in a Sandbox”. In: *Research Foundation Briefs* 3.4 (2017), pp. 1–20 (cit. on p. 121).
- [213] Office of the National Coordinator for Health Information Technology. *Report on Health Information Blocking*. 2015. URL: http://www.healthit.gov/sites/default/files/reports/info_blocking_040915.pdf (visited on May 11, 2018) (cit. on p. 121).
- [214] Roxana Geambasu, Tadayoshi Kohno, Amit A Levy, and Henry M Levy. “Vanish: Increasing Data Privacy with Self-Destructing Data.” In: *USENIX Security Symposium*. Vol. 9. 2009 (cit. on p. 122).

Glossary

ABCI Application Blockchain Interface. 94

ACL Access Control List. 42

ADL Archetype Definition Language. 39

AES Advanced Encryption Standard. 101, 102

AES-CBC Advanced Encryption Standard - Block Cipher Mode. 101, 103

AOM 1.4 Archetype Object Model 1.4. 43

API Application Programming Interface. 42, 44, 73

APPC Advanced Patient Privacy Consents. 40

AQL Archetype Query Language. 39, 43

ATNA Audit Trail and Node Authentication. 40

BPPC Basic Patient Privacy Consents. 40

CA Certificate Authorities. 29, 30, 119

CAD Computer-Aided Diagnosis. 1, 2, 4, 27

CC Citizen Card. 12, 13

CDA Clinical Document Architecture. 42

- CDH** Computational Diffie-Hellman. 18, 56, 80, 86
- CHAP** Challenge Handshake Authentication Protocol. 70
- CIP** Customer Identification Program. 29
- dac** Discretionary Access Control. 31
- DAG** Directed Acyclic Graph. 15, 16
- DFS** Distributed File System. 50, 53
- DICOM** Digital Imaging and Communications in Medicine. 25, 32, 35, 36, 39, 40, 42, 53, 85, 93, 94, 101, 102, 103
- DID** Decentralized Identifier. 29, 46
- DLP** Discrete Logarithm Problem. 18, 22, 23, 56, 68, 79, 80, 81, 86, 98, 99, 113, 115, 117, 118
- DLT** Distributed Ledger Technology. 13, 14, 15, 16, 17, 30, 36, 38, 43, 50, 53, 56, 94, 119, 120, 121, 122
- DNA** Deoxyribonucleic acid. 3
- DoS** Denial of Service. 13, 14, 36, 37
- DPO** Data Protection Officers. 7
- ECIES** Elliptic Curve Integrated Encryption Scheme. 81, 83
- ECR** Ethereum Claims Registry. 47
- EDI** ANSI X12 Electronic Data Interchange. 42
- EHR** Electronic Health Records. 2, 3, 25, 30, 32, 34, 38, 39, 42, 43, 49, 50, 55, 95

- EMIF-AD** EMIF-Alzheimer's Disease. 45
- EMIF-EHR** EMIF-Electronic Health Record Data. 45
- eP-ID** Explicit Pseudonym Identifier Derivation. 54, 55, 84, 85, 105
- FLP** Fischer, Lynch and Patterson. 13
- GDPR** General Data Protection Regulation. 2, 3, 4, 7, 25, 27, 32, 40, 41, 42, 43, 45, 49, 50, 53, 68, 69, 75, 81, 88, 93, 120, 122
- GPS** Global Positioning System. 72
- HIMSS** Health Information and Management Systems Society. 119
- HIPAA** Health Insurance Portability and Accountability Act. 42, 43
- HL7** Health Level Seven International. 40, 42
- IBC** Identity-Based Cryptography. 29, 30
- IdP** Identity Provider. 12, 28
- IHE** Integrating the Healthcare Enterprise. 31, 39, 40, 119
- IMD** Implantable Medical Devices. 34
- IPFS** InterPlanetary File System. 93, 95
- iP-ID** Implicit Pseudonym Identifier Derivation. 54, 55, 82, 84, 85, 89, 90, 149
- IRWF** Import Reconciliation Workflow. 40
- JR-VSS** Joint Random Verifiable Secret Sharing. 22, 76
- KYC** Know Your Customer. 28, 29

- MFA** Multiple Factor Authentication. 32, 46
- MRI** Magnetic Resonance Imaging. 3, 27
- NEMA** National Electrical Manufacturers Association. 40
- OASIS** Open Access Series of Imaging Studies. 44
- OCR** Optical Character Recognition. 32
- ODIN** Object Data Instance Notation. 39
- ONC** Office of the National Coordinator for Health Information Technology. 42
- PAKE** Password Authenticated Key Exchange. 97
- PAM** Privileged Access Management. 34
- PBFT** Practical Byzantine Fault Tolerance. 17
- PCHA** Personal Connected Health Alliance. 39
- P-ID** Pseudonym Identifier Derivation. 6, 75, 80, 81, 82, 85, 93
- PACS** Picture Archive and Communication System. 2, 4, 6, 25, 31, 35, 53, 73, 85, 93, 95
- PIN** Personal Identification Number. 13
- PIX** Patient Identifier Cross-referencing. 26, 28, 75, 81, 119
- PKI** Public Key Infrastructure. 29
- PoS** Proof of Stake. 16
- PoW** Proof of Work. 14, 17

PSH preimage/second-preimage resistance of the hash function. 98, 99

QR Quick Response. 31

RBAC Role-Based Access Control. 30, 31, 33, 34, 42, 45

RDF Resource Description Framework. 41

RFID Radio-Frequency IDentification. 34

RIS Radiology Information System. 25

SAML Security Assertion Markup Language. 10

SQL Structured Query Language. 40, 43

SS-IDs Self-Sovereign Identities. 4, 11, 27, 29, 36, 50, 59, 119

SSO Single Sign-On. 10, 28

SSS-PS Shamir's Secret Sharing perfect secrecy. 98, 99

TLS Transport Layer Security. 72

UDI Unique Digital Identity. 26, 59, 60, 63, 64, 65, 66, 67, 68

UI User Interface. 45

UPI Universal Patient Identifier. 9, 12, 50

VSS Verifiable Secret Sharing. 22, 96

WBSE Witness-Based Searchable Encryption. 35

XDS Cross-enterprise Document Sharing. 40

XML Extensible Markup Language. 39

XUA Cross-Enterprise User Assertion. 40

List of figures

2.1	Digital Identities perspective of provider-centric vs identity-centric. . .	12
2.2	Blockchain vs Directed Acyclic Graph.	16
5.1	Overall architecture the proposed thesis.	53
5.2	Architecture for the proposed security model.	54
6.1	Master identity structure.	60
6.2	Card evolution.	61
6.3	An identity registry.	63
6.4	Anonymous profiles architecture	66
6.5	Profile anchors	67
6.6	Overall authorisation architecture	69
6.7	Sequence diagram for the authorisation protocol	71
7.1	Implicit consent mode (iP-ID)	82
7.2	iP-ID multiparty computation	83
7.3	Explicit consent mode (eP-ID)	84
7.4	Interpolation of multiparty computations per second	90
8.1	Architecture for the federated storage	94
8.2	Federation of data curators	95
8.3	Structure of encrypted records and files	97
8.4	The throughput results (in record per second, R_n/s) for different sizes of R_n chains and τ values. Values from Table 8.1.	103
9.1	Token request/response and token usage architecture.	106
9.2	Protocol flows for the anonymous token.	111

List of tables

7.1	The average time and memory consumption for the master key setup. .	79
7.2	The average of a run for different numbers of parties and threshold values.	81
7.3	The average time of a multiparty computation for different threshold values.	89
7.4	The expected throughput for the iP-ID function.	90
8.1	The throughput results for different chain sizes.	102
8.2	Derivation times for different threshold values.	102
9.1	The average time taken by the client to process and aggregate shares. .	118

