TU DELFT

MASTER THESIS

Blockchain-based distributed tamper-proof filesystem using threshold encryption

Author:
Angela PLOMP

Supervisor:
Dr. Johan POUWELSE

A thesis submitted in fulfillment of the requirements for the degree of Master of Science

in the

Distributed Systems Group
Computer Science

September 3, 2018

Contents

Chapter 1

Introduction

1.1 Barbie's hospitalization

In January 2018, Dutch reality star Samantha "Barbie" de Jong received acute medical care in the Haga Hospital in The Hague (De Telegraaf 2018). Her hospitalization was met with great interest from several media companies, who speculated about possible causes. A few weeks later, it turned out that an abnormally large number of Haga Hospital employees had looked into one partical patient's files: Ms. De Jong's. This news resparked a debate about the merits and risks of storing medical data the way we do.

1.2 Digitalized medical records

1.2.1 History of medical records

One of the oldest medical documentations found is an Egyptian papyrus, dating from 1600 BC. It contains a didactic recording of a surgery (Gillum 2013). Later, in Ancient Greece, one of the most famous doctors of world history made a big contribution to medical records: Hippocrates (460-370 BC). He documented many medical case studies, notes and philosophical ponderins, bundled in the Hippocratic Corpus. As the interest in natural science in general and human anatomy in particular rose during the 17th and 18th century in the Western world, more and more records were kept on the suspected origin and possible treatment of diseases. Still, these records were kept for educational purposes and not to track individual patients' health trajectory. This only started to change in the 20th century. Most governments of european countries started requiring physicians to keep records on their patients in a specific format. With the rise of (affordable) computers in the 1960s, medical records moved towards digital files that can be shared between health care providers such as GP's, hospitals and specialized clinics.

1.2.2 Data ownership

A central question with regard to medical information is: Who owns the data? Many patients feel that they do not control access to their data, but would like to be able to access the data themselves, look at the history of data access and give or deny access permissions to healthcare providers (World Economic Forum 2012). The data is about them, so they feel they should have ultimate control over it. In a particularly bad case, patients that doubts the confidentiality of their records may not make completely honest disclosures, holding back potentially crucial information. On the other hand, the data has been collected and stored by the healthcare providers. They invest time and money into this process. Data ownership should not be seen as a

binary either/or choice. Moreover, the burden of coming up with policies and implementation of these policies lies on the health care provider (Kostkova et al. 2016).

1.3 Cyber crime and other concerns

1.3.1 Data theft or leakage

As in any digital information system, providing adequate protection of the data is a serious concern. EMRs may contain extremely sensitive data: most people would not want others to know if they suffer from stigmatized illnesses like sexually transmittable diseases or mental disorders. In a more practical way, information about someone's medical history may for example have a negative effect on their chances of being hired for a job. In some parts of the world, medical identity theft is a problem. This is when a person uses another person's identity to fraudulently receive health care or prescription drugs. According to a study on medical identity theft from 2016, the last years showed an upward trend in the number of medical identity theft cases in the USA. The main causes for this identity theft are the stealing or abusing of credentials of family members, a data breach at a healthcare provider or the submission of credentials on a phishing page (Ponemon Institute 2016).

1.3.2 Privacy concerns around EMRs

The United Kingdom launched NHS Care.Data in 2013, an initiative to centralize patient health care data. Patient information could be legally shared with stakeholders outside of the NHS or medical research community. A report found multiple severe problems with this system in terms of privacy and patients' power over their own data (Presser et al. 2015). Data was processed without properly consulting or even informing patients. Sometimes, data was optimistically categorized as anonymous or pseudonymous even though techniques exist deduce personal information from it (. Li, T. Li, and Venkatasubramanian 2007). GPs were required to send records to the central database, but were simultaneously required by another law to keep the records confidential, which led to legal complications. Another system by the NHS, the Detailed Record System, was classified by researchers as "almost certainly illegal under human rights or data protection law" (Anderson et al. 2009).

1.3.3 Impact of the GDPR

In May 2018, the General Data Protection Regulation (GDPR) came into effect in the European Union, as a replacement of the Data Protection Directive (DPD) of 1995. The DPD already forced EU member states to take into account data protection on computers and other electronic devices (Calder 2016). The GDPR presents six principles that should be adhered to when collecting, storing and processing data. These mainly concern the proportionality of the data gathering for a certain goal and transparency of and consent for the use of the data. Organisations are held responsible for proving that they comply with the rules. The GDPR is not specifically designed for medical data. There may exist conflicting objectives when it comes to ensuring privacy rights versus providing adequate access to data (European Society of Radiology 2017). The GDPR requires healthcare providers to grant patients access to their files, as long as the access requests are 'manifestly unfounded or excessive'. The Regulation provides several exemptions and derogations for the use of health data, if applying the law would prevent or seriously impair research (McCall 2018).

Chapter 2

Problem statement

2.1 Barbie's medical records in HiX

The hospital where Ms. De Jong was treated for her medical problems, Haga Hospital, uses ChipSoft's HiX software for the storage and processing of patient's medical records (HagaZiekenhuis 2016). HiX does record the access of users to the digital files. During a routine check, the access to Ms. De Jong records by staff who were not treating her came to light. This violation of her privacy is deemed unacceptable by many people. In this research, my hope is to contribute to the development of a more secure medical record file system in which the patient's involvement is central. Ms. De Jong should easily have access to the log of persons who viewed the record herself. Additionally, she should know the exact contents of these records and agree with their storage.

2.2 Research goal

The goal of this master thesis project, is to research the possibilities of expanding patients' power over and knowledge about their medical records. This power consists of two parts:

1. Accountability on access: knowing who has accessed the file;
2. Validation of EMR entries from both the health care provider and the patient.

In addition to this, the traditional security goals for any still stand: confidentiality, integrity and availability. In the earlier days of medical record systems, there was a lack of clear security policies for these kinds of systems, as a consequence of little awareness of the ethical and legal duties for medical data protection. Anderson (1996) presented a security policy model for clinical information systems, consisting of nine principles. In the next paragraph, the relevance of these principles and other frameworks for accountability on access and validation of entries in medical systems will be explored.

2.2.1 Accountability on access

In 2007, Scotland dealt with a very similar case to De Jong's when over 50 employees of an NHS hospital illicitly looked into a celebrity's medical record (Carvel 2017). This scandal occurred just before upgrading the medical file systems to a new and controversial version. However, an NHS spokesperson stated something very interesting: *"The reality of the situation is that, for the first time in the history of medical records, the new IT systems being implemented across the NHS have a fully integrated audit trail that tracks access to any care record to safeguard and maximise patient confidentiality."* The fact

of the matter is that the new system which provided the audit trail, made it possible to hold the health care providers accountable for their privacy invasion. Accountability on access means that a patient can verify who has accessed accessed a file, and when. There should be no way for someone to access the file without leaving a trace. When a patient questions the legitimacy of an access event, the person who looked into the file can be asked for an explanation. One of the aforementioned Anderson's nine principles is stated as follows: *"All access to clinical records shall be marked on the record with the subject's name, as well as the date and time. A audit trail must also be kept of all deletions"* (R. J. Anderson 1996). A recent paper that points out the lack of patient-centered transparency requirements for medical data systems, states that transparency is needed for accountability. The authors define ex-post transparency as *"enabling the patient to be informed or get informed about what happened to his/her medical and personal data"* (Spagnuelo and Lenzini 2016). In order to fulfill this ex-post transparency goal, a number of transparency requirements were formulated. When it comes to the relation between transparency and accountability, the most relevant of these requirements are:

1. The medical record system must provide the patient with accountability mechanisms.
2. The medical record system must provide the patient with evidence regarding permissions history for auditing purposes.
3. The medical record system must provide the patient with evidence of security breaches.

These requirements guide the design of an EMR system that center the patient's need of privacy and power over their own data. Thus, these criteria will be used in the set up of the requirements for the system presented in this thesis.

2.2.2 Validation of EMR entries

According to University of Leeds researchers, an EMR is valid if all events have been recorded and all records signify an event. Additionally, it should be clear what every record means (Neal, Heywood, and Morley 1996). Later researchers have extended this definition to: *"Medical records, whether paper or electronic, record health events. Records are valid when all those events that constitute a medical record are correctly recorded and all the entries in the record truly signify an event"* (Hassey, Gerrett, and Wilson 2001). In this master thesis, validation of EMR entries means that an entry becomes official only when both the patient and the health care provider have agreed to the entry. This is similar to a person sending a registered letter and the recipient signing for delivery. The patient cannot claim not to know the content of the entry. Research found out that there are significant discrepancies between health care reported by physicians themselves, patient surveys and written medical records (Stange et al. 1998). Another interpretation of the concept of validation of EMR entries is to verify whether the content of the records, e.g. lab results, are actually accurate. This is not related to patient power over data and therefore out of scope for this research.

2.3 Research question

Taking the aforementioned considerations into account, the research question for this thesis project is as follows:

R: *"How can blockchain technology be used to design an Electronic Medical Record (EMR) system, that guarantees accountability on access and validation on entry addition?'*

This question can be split into two subquestions:
R1: *"How can blockchain technology be used to guarantee accountability on access in an EMR?"*
R2: *"How can blockchain technology be used to validate entries in an EMR?"*

When the two subquestions are answered, the main research question can be answered as well. The proposed solution will be supported by a simple prototype as a proof-of-concept.

2.4 Requirements

Before making a design, it should be clear what the requirements are. These are used for both the design of the system and the validation after building the prototype. Some requirements are general, others are specifically needed for answering the research questions.

2.4.1 Requirements for accountability and validation

The proposed system should fulfill the following requirements directly related to the research questions:

1. Accountability on access: Every access to an entry in the EMR system is recorded. The log contains information on the name of the user who accessed the file, the name of the file itself, and the timestamp of the event.
2. Validation of entries: A user should be able to sign an entry with a secure digital signature. The digital signatures should be verifiable by anyone in the system.

2.4.2 Requirements from the CIA triad

A standard in the field of information security is the CIA triad. This stands for the security goals of confidentiality, integrity and availability that any secure information system should meet. Based on these goals, the following additional requirements are constructed:

1. Confidentiality: Information stored in the EMR system itself as well as the event log should only be accessible to the users it is intended for.
2. Integrity: Information stored in the EMR system cannot be changed by and adversary without being noticed.
3. Availability: Information stored in the EMR system is available for the users whenever they need or want to access it.

2.4.3 Requirements for the user experience

The prototype for the proposed system is not intended as a ready-to-use system for the real world. The user experience has a low priority as it is not really needed to demonstrate the qualities of the system for its intended goal. However, there are some minimal requirements:

1. The user should be able to navigate between the functionalities of the system without effort;
2. The information displayed to the user should be clear and easily understandable;
3. The user should be able to easily verify that the access log has not been tampered with.

2.5 Research method

First of all, a literature study is conducted on the topic of EMRs and the state-of-the-art of blockchain-based medical systems. The focus lies on the use of blockchains to improve patient's power and knowledge over their data. Then, possible design choices for a system that fills the requirements as stated in this chapter will be explored. Two aspects are taken into account. First, the desired functionality and ideas found in previous work by researchers that touch upon this subject. Second, the technologies available in practice. Recent developments in computer science will not always be available in the form of working code yet. After analyzing the design options, the prototype will be made. When the protoype is tested, it will be validated by checking it against the requirements stated in this chapter.

Chapter 3

Preliminaries

3.1 Introduction to blockchain

3.1.1 Hype or revolution

Blockchain emerged in 2008 with the implementation of the first cryptocurrency, Bitcoin. Essentially, blockchain is a peer-to-peer distributed ledger, which can only be updated via consensus (Nakamoto 2008). It runs as a layer on top of TCP/IP. Blockchains can be public, private or semi-private. Anyone can participate in a public (or permissionless) blockchain: all participants hold a copy of the ledger but none of the participants actually own the ledger. This ensures the decentralized nature of the blockchain. A private blockchain is open only to an organization or consortium. Semi-private blockchains are a combination of a public and private part (Bashir 2017). The idea of using blockchain as a solution for a problem is often met with scepticism by people who see blockchain technology merely as a hype. On one hand, some organisations seem to see the use of blockchain as a goal in itself. However, tamper-proof logging is the core functionality of a blockchain. That is why blockchain technology is very promising for the purpose of fulfilling the research goal of this thesis.

3.1.2 Blocks

As the name implies, a blockchain is in essence a chain of blocks. A block minimally consists of:

1. The hash of the previous block;
2. A nonce (number used only once);
3. A bundle of transactions.

The first block in a blockchain is called the genesis block. This is hardcoded at the time the blockchain was started. To add a block to the blockchain, all nodes must agree on a single version of truth (concensus).
There are roughly two categories of consensus mechanisms: Proof- and leader-based or Byzantine fault tolerance-based. Bitcoin uses the proof-of work consensus mechanism to prove that enough computational resources have been spent in order to propose an addition to the blockchain. Nodes can compete with each other to be selected in proportion to their computing capacity. For Bitcoin, the proof-of-work requirement is as follows:

H (N || P_{hash} || Tx || Tx || . . . Tx) < **target** where
H is an ideal hash function,

N represents a nonce,
P_{hash} is the hash value of the previous block, and
Tx are the transactions in the proposed block.
The hash value of these concatenated fields should be smaller than the set **target** for difficulty.

This problem cannot be solved with a smart algorithm: it must be brute forced. A consequence of this is the effectiveness against Sybil attacks. The high costs of creating pseudonymous identities prevents cheap attacks (Vukolić 2015). A drawback is that it is (obviously) computationally intensive, and therefore uses much energy, which is an unnecessary strain on the environment. The proof-of-stake algorithm uses the stake that a user has in the system, for example invested time, to trust that the benefits of performing malicious activities would not outweigh the benefits of staying in the system as a trusted member (Kiayias et al. 2017).
Deposit-based consensus requires putting in a deposit before proposing a block to be added to the blockchain. In case the block is rejected by others, the user loses its deposit (Solat 2017). Reputation-based mechanisms let members elect a leader node, based on the reputation it has built on the network. When a transaction is added to a block, it should be clear who has performed this transaction.

3.1.3 Identity and verification

Particularly in the medical use case, any access to the EMR should be linked to an identity. A digital signature confirms the identity, under the condition that such a signature can be verified but cannot be forged. Digital signatures can be issued using different algorithms. Bitcoin uses the Elliptic Curve Digital Signature Algorithm (ECDSA). Adding a block to the blockchain is done through the following consensus algorithm (Nakamoto 2008): new transactions are broadcast to all nodes; each node collects transactions into a block; in each round, a random node (selected by the proof-of-work) gets to broadcast its block; other nodes accept the block if and only if all transactions in it are valid; nodes express their acceptance of the block by including its hash in the next block they create.

3.1.4 Tamper-proof qualities of blockchains

As a rule of thumb, a block is 'permanently' added if it has been in in the blockchain for six rounds. The probability of another version of the blockchain, not containing this particular block, becoming longer and thus the official blockchain, is negligible. Because every block contains a hash pointer to the previous block, one can access the previous information, but also verify that it has not changed. Tampering is evident because the hash of the changed information would change, too. A binary tree with hash pointers is called a Merkle tree. An essential quality of a Merkle tree is that it can hold many items, but one just needs to remember the root hash one can verify membership of the tree in just $O(log\ n)$ time and space (Szydlo 2004). Although data can be stored in a blockchain directly, a blockchain is not suitable to store large amounts of data. This is why many blockchain-based systems use a distributed hash table (DHT) that only stores pointers to the actual data.

3.2 Identities and signatures

3.2.1 Self-sovereign identities

Accountability on access can only be established when it is guaranteed that the person accessing or modifying the file is indeed the person who is recorded as doing so. This means that we will need a solid identification and authentication method for the file system. Traditionally, this goal has been attained by using username/password systems. There are several drawbacks to this system. It provides a terrible user experience for many people, especially if they have to memorize a large amount of passwords and change them regularly. This sometimes leads to irresponsible password behaviour (Adams and Sasse 1999). Another issue is that a user has to create a new identity for each application. These identities only exist within the context of each specific website or application, leading to great volumes of data duplication (Tobin and Reed 2016).

3.2.2 Digital signatures

As paperwork has been replaced by digital entries, digital signatures have taken over the role of traditional signatures. A digital signature provides proof of the integrity of the authorship, because anyone can verify that the signature is based on the author's public key. On the other hand, only the person who creates the message should be able to generate a valid signature. In general, the steps to create a digital signature are as follows:

1. The signature algorithm is a function of the signer's private key k_{pr}. Hence, only one person can sign a message x, assuming that the private keys are kept secret.

2. The message x is an input to the signature algorithm as well, to make sure that the signature is related to the message and cannot be re-used.

3. A digital signature algorithm is run with the right inputs, which yields signature s. Then, s is appended to x and the pair (x, s) can be sent.

Digital signatures can be created using a range of different algorithms, based on for example Digital Signature Algorithm (DSA), prime factorization (RSA-based signatures) or the discrete logarithm problem (ElGamal-based signatures) or on the elliptic curve discrete logarithm problem.

3.2.3 Elliptic Curve Digital Signature Algorithm

Elliptic curves have some advantages over RSA and discrete logarithm-based schemes. Threshold versions of DSA are unusable in practice (R. Gennaro, Goldfeder, and Narayanan 2016). One of these advantages is that a small key length provides the same security as other schemes, but with a shorter processing time. The Elliptic Curve Digital Signature Algorithm (ECDSA) is defined over prime fields as well as over Galois fields. Here, the procedures for the more popular version over prime fields are given (Paar and Pelzl 2009).

1. For key generation, an elliptic curve E is chosen with modulus p, coefficients a and b and a point A which generates a cyclic group of prime order q. Choose a random integer d such that $0 < d < q$. Compute the new point $B = dA$.
   $k_{pub} = (p, a, b, q, A, B)$
   $k_{pr} = (d)$

2. In order to generate a signature, an integer such that $0 < k_E < q$ is chosen as an ephemeral key. Compute $R = k_E A$. Let $r = x_R$ (the x-coordinate of point R) and compute the signature $s \equiv (h(x) + d \cdot r)k_E^{-1} \bmod q$.

The main analytical attack against ECDSA, assuming that the parameters are chosen correctly, is trying to solve the elliptic curve discrete logarithm problem. Considering that this is an NP-complete problem, it is extremely unrealistic to solve this in time.

3.2.4 Elliptic curve threshold signatures

Similarly to the threshold encryption schemes discussed before, threshold cryptography can be applied to digital signatures. A scheme to achieve this was first presented in 1992 by Desmedt & Frankel. This method was based on the RSA signature scheme (Desmedt and Frankel 1991). Since then, many papers have been published presenting threshold signature schemes. One of them was a robust Elliptic Curve threshold DSA scheme. For this project, the focus will be on Elliptic Curve threshold signature schemes, because of the previously mentioned advantages. Specifically, a scheme is needed which is fit to execute on a distributed system.

3.2.5 Threshold ECDSA in a fully distributed system

In 2015, researchers at the Worcester Polytechnic institute presented a fully distributed signature system for threshold ECDSA, named *Nephele* (Green and Eisenbarth 2015). This system is mainly built to protect the key from side-channel attacks and is designed in such a way that a private key never even needs to appear in memory. The key generation as well as the signature generation algorithm is fully distributed. It also allows for fully distributed key re-sharing.

Key generation

The private key is chosen by all the nodes together using Joint Random Secret Sharing (JRSS). In this technique, each node chooses a random local secret value and shares it with the group, using Shamir's Secret Sharing (. Gennaro et al. 1996). Every node adds all the shares together (including its own), resulting in the joint random secret share. Just one of the nodes needs to introduce randomness to keep the joint secret unknown.

MORE ON THRESHOLD ECDSA

3.2.6 Identity-based signatures

Considering the wish for transition to self-sovereign identities as explained in paragraph 3.3.1, the possibility of using identity-based signatures should be researched. Because the core goal of this project is to design a system with patient's power in mind, it would be fitting if patients do not have to rely on an external party to provide their identification. The idea of identity-based signatures is a public key cryptosystem in which the users do not have to exchange public keys because the public key of a user is simply a person's email address or other personal identification (Shamir 1984). Requirements for this identification is that it uniquely identifies the user in a way that cannot be denied afterwards, and that the information is available to anyone within the system. A trusted party computes the private key for every user and issues the keys on a smart card.

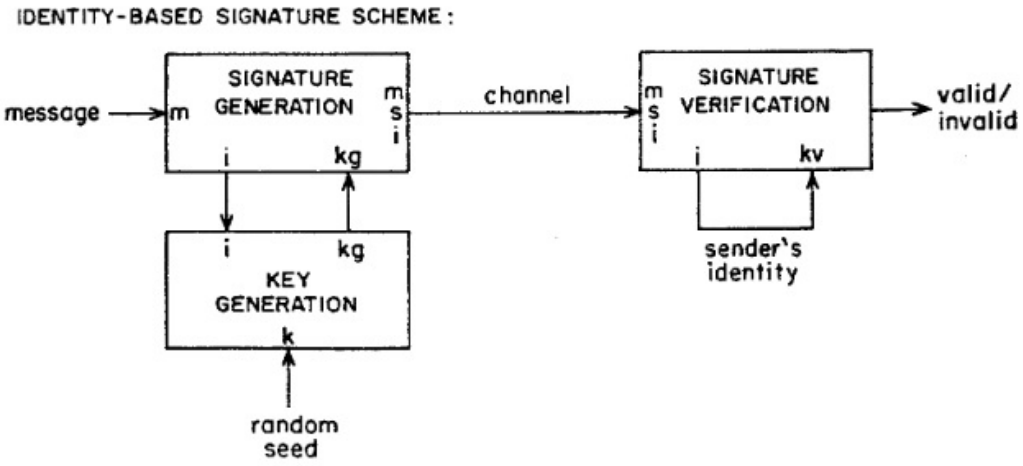


FIGURE 3.1: Identity-based signature scheme (Shamir 1984)

3.2.7 Schnorr signatures

Since a few years, some Bitcoin enthusiasts have been lobbying for the usage of Schnorr signatures to sign transactions. One of the major challenges for blockchains in general is scalability. Consider the scenario that a user would like to send a certain amount of bitcoins from multiple accounts to one account. In the current system, the transaction from each source account to the destination account requires its own signature. However, if it is just one user sending the transaction, they should be able to place just one signature for the combined transactions. Schnorr signatures enable users to do this. Cutting superfluous signatures could potentially achieve a significant reduction in bandwidth, which in turn makes up space for more transactions: increasing scalability.

Chapter 4

State of the art

4.1 Access logging in current widely-used EMR systems

The need to have an auditable access trail for EMR systems is not new, and current systems already have some kind of access logging. In The Netherlands, the EMR market is dominated by two parties: Chipsoft and Epic.

4.1.1 Chipsoft

The hospital where Ms. De Jong received care, uses Chipsoft's HiX (Healthcare information eXchange). Chipsoft is a dutch EMR developer and delivers several versions of their Microsoft-based HiX software.

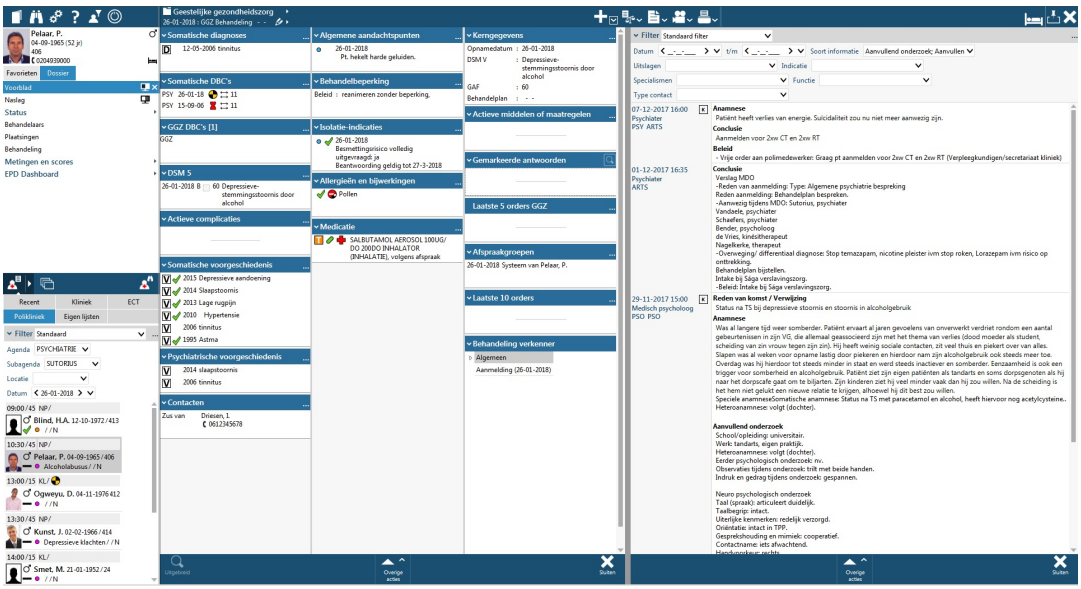


FIGURE 4.1: Screenshot of HiX software

4.1.2 Epic

Epic is an EMR software developer based in the United States.

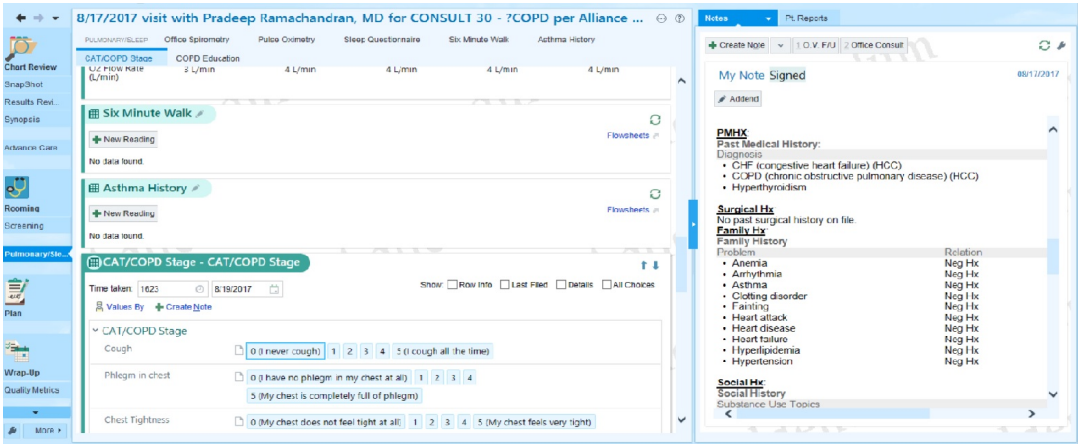


FIGURE 4.2: Screenshot of Epic software

4.2 Blockchain-based EMR systems

4.2.1 Scientific work

This research would definitely not be the first to incorporate blockchain into a EMR system, although it may be the first one to use blockchain technology for the specific purpose of empowering patients with knowledge over what happened to their data. In this section, four papers that present blockchain-based EMR systems are studied.

MedRec

MedRec is a EMR system aimed at managing authentication, confidentiality, accountability and data-sharing. The paper in which this system is presented identifies interoperability challenges between healthcare provider systems as a major barrier towards effective data sharing. The authors designed a public key cryptography-based blockchain structure that could be applied to create append-only, immutable, timestamped EMRs (Ekblaw et al. 2016). The block content consists of information about data ownership and viewership permissions. Smart contracts are used to log events such as data retrieval. A prototype was made to demonstrate the qualities of the system.

OpenPDS

Zyskind & Nathan proposed a model called OpenPDS for an information system in which a mechanism for returning computations on the data is included: return answers instead of data itself. The contribution of this paper is twofold: Combination of blockchain and off-blockchain storage to construct a personal data management platform focused on privacy; Perform trusted computing on blockchain-handled data. The proposed systems treats users as the owners of their data and provides them with data transparency and fine-grained access control. A rough sketch of the functionality of the system is as follows: A users installs the application on a smartphone. Data collected on the phone is encrypted using a shared encryption key and sent to the blockchain. The blockchain routes it to an off-blockchain key-value store using a DHT, only retaining a SHA-256 hash pointer. Anyone wanting to access the data can send a request to the blockchain, which in turn verifies the digital signature of the requester as well as the listed permissions for this user (Zyskind, Nathan,

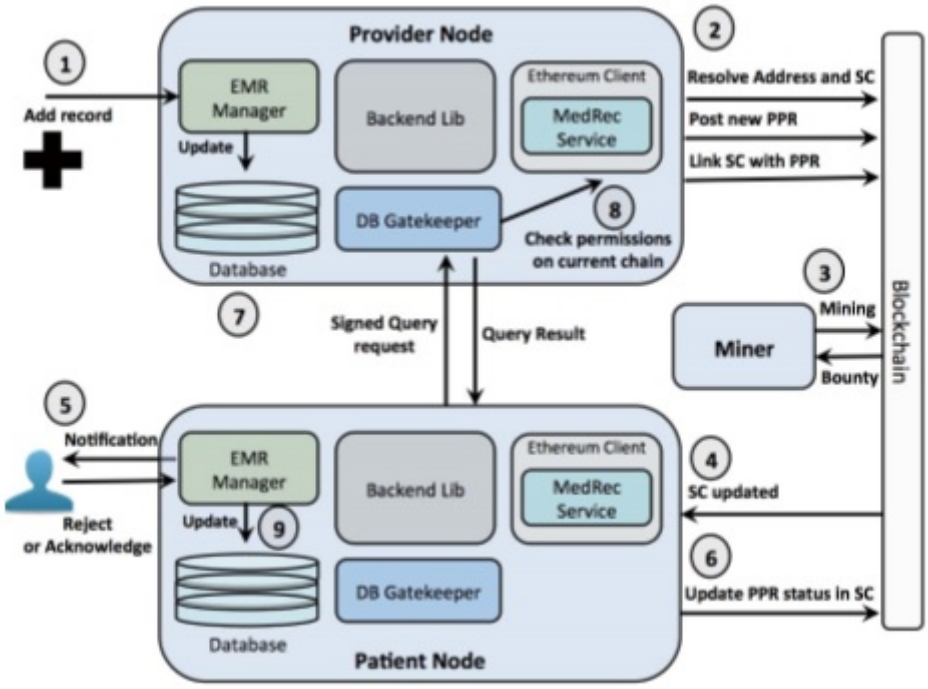


FIGURE 4.3: Overview of MedRec system (Ekblaw et al. 2016)

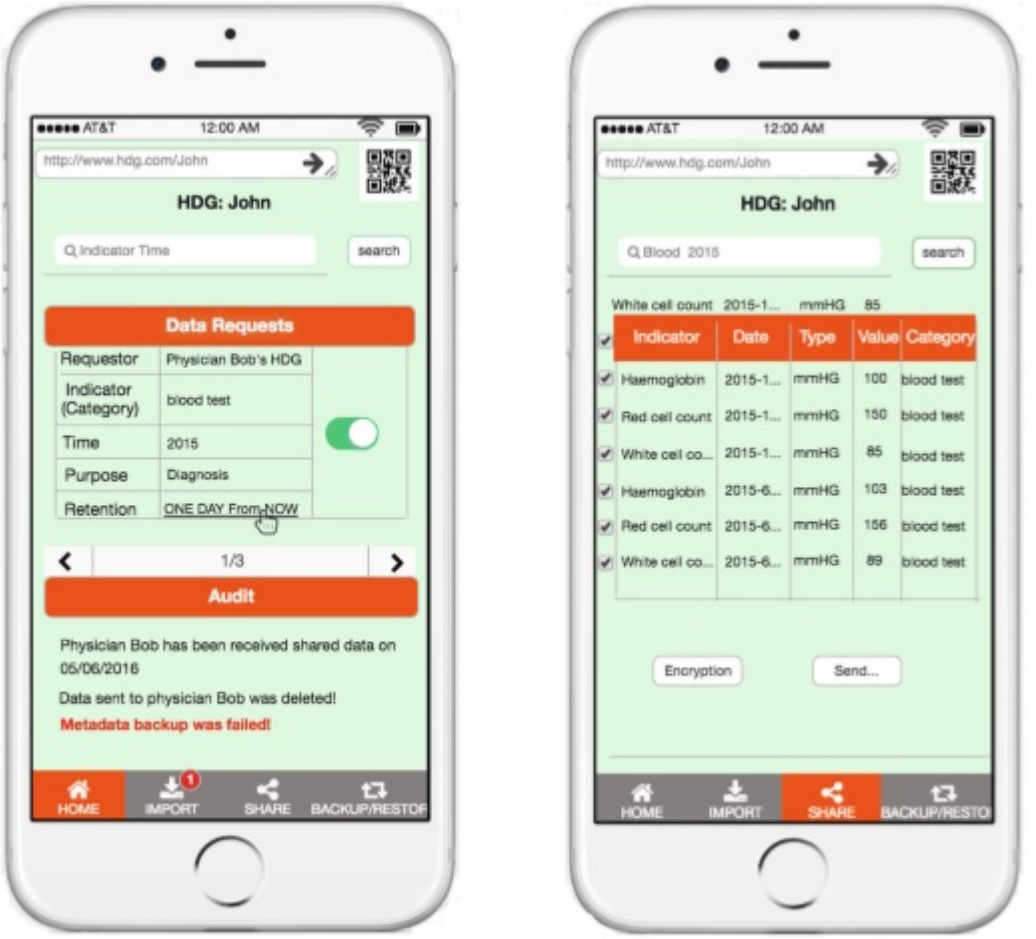
et al. 2015). Assuming that users manage their keys in a secure manner, the system provides security and privacy. An adversary cannot really learn interesting information from the blockchain itself, because it only stores hash pointers. Even if it would control a large amount of nodes, the raw data is still encrypted using a key that none of the nodes possess. Adversaries are prevented from posing as a user because of the digitally-signed transactions and the decentralized nature of blockchain.

Healthcare Data Gateway

In 2016, Xiao Yue presented a fairly similar system called the Healthcare Data Gateway app. It is a combination of a traditional database and a gateway. Personal electronic medical data is managed by a blockchain. All data requests are evaluated for permission. In case of a granted permission, secure multiparty computation (sMPC) is used to process patient data without risking patient privacy (Yue et al. 2016).

Enigma

Enigma is a computation platform proposed by Zyskind. Their paper states that blockchain can neither handle privacy nor heavy computations. Enigma can be connected to an existing blockchain. The goal of the platform is to facilitate developers to build privacy-by-design, decentralized applications without using a trusted third party (Zyskind, Nathan, and Pentland 2015). Just like most blockchain-based systems, it uses a DHT that stores references to the data. sMPC is used by splitting date between nodes and performing computation on these nodes without transferring any information from one node to another. Each node has a piece of seemingly random data, that is useless on its own. In general, sMPC systems are based on secret



(a) Home user interfaces of John's HDG

(b) Data sharing illustration of John's HDG

FIGURE 4.4: Example of HDG screenshots (Yue et al. 2016)

sharing. This is a category of threshold cryptosystems, in which a secret s is divided into n parts, and at least t shares are required to reconstruct s. Such a system is written as a (t, n) threshold system. Shamir's secret sharing scheme is a famous example of a secret sharing scheme, which uses polynomial interpolation. The Enigma platform provides an API which facilitates the uses of a sharing scheme based on Shamir's scheme. In total, there are three decentralized databases in the system: the public ledger, the DHT and the sMPC database. Nodes are compensated for their computational resources via computation fees.

4.2.2 Startups and industry-based projects

Several startups and government- or industry-based projects have come up in the last few years on the subject of blockchain in healthcare. These range from conceptual frameworks to functioning prototypes. A few Dutch projects are listed here.

Mijn Zorg Log

Mijn Zorg Log is a smartphone app, developed by the Dutch Health Care Institute (Dutch: *Zorginstituut Nederland*). This app can be used by people who receive home care to log the hours that the home help spent at their house and the nature of the

care. The home care provider can then verify these hours and use them for their administration. A permissioned blockchain is used, with two types of nodes: member nodes and authority nodes. Only authority nodes participate in the mining process. An experiment has been conducted using this app for administration in maternity care. The results were mainly positive, especially concerning the self-reported reduction of the administrative burden (Felix et al. 2018).

MedMij

MedMij is a framework that consists of agreements about how medical data should be exchanged in a blockchain-based healthcare application. It is therefore not a working product in itself. Health care providers that want to develop a digital healthcare application, can hire a MedMij-certified vendor to implement a compliant system.

Chapter 5

Design choices

5.1 Implementation details of existing systems

In the previous chapter, several papers presenting blockchain-based EMR systems were discussed. Unfortunately, not every presented system is accompanied by a prototype. Three systems provided an implementation or a description of a possible implementation. Here is an overview of the implementation details of these systems.

	MedRec	OpenPDS	Healthcare Data Gateway
Goal	Manage: data access	Manage: data ownership, transparency and auditability, access control	Own, control and share own data easily and securely
Blockchain	Ethereum	Not specified, but assumes qualities similar to Bitcoin	Not specified, mentions "private blockchain cloud"
Block content	Data ownership viewer permissions	Hash pointers (Kademlia)	Encrypted healthcare data
Programming language	Python	Not specified	Not specified
Consensus algorithm	Proof-of-Work	Proof-of-Work	Not specified
Mining reward	Access to aggregated anonymized medical data	Not specified	Not specified
Identity confirmation	DNS-like system that maps real-life ID to ETH address	Pseudonymous compound identities	Not specified

Of these three systems, only MedRec features a working prototype. The absence of a prototype or a very detailed description of a possible implementation, make it difficult to make an informed decision about the implementation choices based on just these examples. In the next secions, some implementation options will be explored and compared.

5.2 Blockchain choices

There are several ready-to-use blockchain libraries available that could be used for this project, or an own blockchain could be constructed.

5.2.1 TrustChain

Researchers at TU Delft developed TrustChain, a scalable blockchain with an emphasis on resilience against one of the primary challenges in permissionless blockchains: Sybil attacks (Otte 2017). A Sybil attack takes place when an adversary forges many fake identities to gain a larger influence of that system than it should actually have (Douceur 2002). The author states that when there is no central trusted authority to assert the one-on-one correspondence between an entity and its identity, it is practically impossible to distinguish identities. This poses a fundamental problem for permissionless blockchains, because they are fully decentralized.

5.2.2 Ethereum

Ethereum is a blockchain that has the possibility of smart contracts as its main feature. Just like Bitcoin, it uses a proof-of-work mining method to make sure that the longest blockchain is the one that has received the greatest investment in terms of computing power (Wood 2014). For Python, there exist several Ethereum libraries, one of which is PyEthereum.

5.2.3 Kademlia

Kademlia is a Distributed Hash Table (DHT) for peer-to-peer networks with an XOR-based metric network topology. A DHT stores (key, value) pairs, the key being a hash, providing a lookup service. Nodes in a Kademlia DHT use UDP to communicate, but has mechanisms to overcome packet loss (Maymounkov and Mazieres 2002).

5.2.4 Own blockchain

Besides using existing blockchains, there is the possibility to create an own blockchain from scratch. An advantage of this, is that it provides the researcher the opportunity to only implement the features that are necessary for the goal. A disadvantage is that it might take more time to write functions that are already defined in the available libraries.

5.3 Digital signature algorithm

In the previous chapter, the benefits of elliptic curve cryptography for digital signatures were discussed. For the prototype, a method is needed to implement ECDSA signatures. One way to achieve this is to implement a paper describing these types of signatures (Green and Eisenbarth 2015). However, implementing a cryptography paper without extremely thorough knowledge of the matter is very tricky. Considering that cryptography is not the main subject of this master thesis, it is wiser to use a ECDSA library for Python. Fortunately, there are several of them.

5.3.1 Using threshold signatures

During the literature study of this project, research was conducted on the state of the art of threshold ECDSA algorithms with the purpose of using these to validate entries in the EMR. In practice, the feasibility of using threshold ECDSA signatures for this project is questionable. First of all, the group of nodes is a dynamic coalition in the sense that health care providers are expected to enter or leave regularly. This makes key (re)distribution hard and time-consuming. The second concern is more practical in nature. There does not seem to be widely used and thoroughly tested threshold ECDSA library available for Python.

5.3.2 Regular EC signatures

Considering the lack of a reliable threshold ECDSA library for Python, the search was extended to libaries that support the creation and verification of regular ECDSA signatures. In the following table, three libraries are compared

	python-ecdsa	python-nss	ecpy
Stars on GitHub (16-07-18)	359	0	20
Language	Pure Python	C with Python wrap	Pure Python
Options	ECDSA only (compatible with OpenSSL)	Supports many network security services	Multiple EC crypto options
Documentation	Abundant documentation with clear examples	Limited and partially outdated documentation	Quality of documentation is sufficient
Speed	0.06-0.6s per generated signature, depending on key length (on laptop from 2008)	Not specified, assumed to be faster because it is written in C	Not specified for signatures
Weaknesses	Vulnerable for timing attacks	Not specified	Not specified

This comparison shows that python-ecdsa is the most popular library and has the most abundant documentation. In turn, the key signature generation is assumed to be slower than for the other libraries. Because the signatures generations are triggered manually in the system and can only be performed on a limited number of entries, it is acceptable to use a slower method. Therefore, python-ecdsa is chosen as the ecdsa library for the prototype.

5.4 Monitoring access to files

One of the core functionalities of the prototype should be the ability to monitor access to files and save these events to a blockchain. Therefore, a method is needed to monitor file events. The first option is to store the files in a directory on the operating system of the user and incorporate a file monitoring function into the prototype to watch for changes. The second option is to let the user download the files on the webpage and use the mouseclick event on the download button as the sign that the user has indeed accessed the file.

5.4.1 Monitoring files on OS

Hard disk drives retain data even after the device has been turned off. This retention is called persistent storage. Several operations can be executed on the stored files, as described in the CRUD (Create, Read, Update, Delete) and REST (Representational State Transfer) processes. Applied to the case of an EMR system, the four basic functions of persistent storage in CRUD are:

1. Create: uploading an entry of a medical record;
2. Read: accessing and reading an entry of the medical record;
3. Update: modifying an entry;
4. Delete: destroying an entry.

The research question is centered on accountability on access, so any read event of the medical files should be monitored. There are some Python tools suitable for this purpose.

	py-notify	watchdog	fsmonitor
Stars on GitHub (08-08-18)	1	3062	55
Language	C and Python	C and Python	Python
Platform	Linux only	Windows and Linux	Windows and Linux
Documentation	Limited documentation	Small tutorial, some blog posts	Small tutorial
Functionality	Tools for Observer programming pattern	Live filesystem monitoring API and shell utilities	Live filesystem monitoring API

Py-notify seems to be slightly outdated, as the current version is several years old and the documentation page contains dead links. Watchdog is the most popular API in terms of GitHub stars, contains the most extensive documentation and covers the functionality that is needed for the prototype.

5.4.2 Monitoring download of files on webpage

If this option is chosen, every file that has been uploaded should be downloadable by every user in the system. This has two advantages. The first one is that it is user friendly, as one can directly access the information. The second one is that monitoring the access becomes very straightforward: the download of the information is the access event. Of course, this option also has its disadvantage. The APIs listed in the previous section monitor all the CRUD functions, so if a file has been modified, the log will show that. In turn, this solution would not be able to detect this. This does not make the access log less tamper-proof however, because the modified file will create a new event on the access log if it is re-uploaded.

Chapter 6

MediChain prototype

6.1 Overview

The prototype consists of a blockchain and a website to which a user can upload files. The nodes in the blockchain are the patients and their health care providers. Each patients has their own blockchain. When a file is uploaded to the website, the event is logged on the blockchain. A user can take a look on the *logs* page on their personal page to see all the events.

6.2 Implementation details

The prototype is written in Python. The web framework Flask is used to construct a simple website and receive HTTP requests.

6.2.1 Home page

On the home page, there is a button to select a file from the user's computer to upload it to the system. Upon receipt of the file, it is stored and the event of the upload, including timestamp, is added to the blockchain. The user is presented with a page which gives the option to return to the home page or visit the logs page.

6.2.2 Logs page

The logs page consists of a visual representation of the blockchain.

Bibliography

Adams, A. and M. A. Sasse (1999). "Users are not the enemy". In: *Communications of the ACM* 42.12, pp. 40–46.

Anderson, Ross J (1996). "A security policy model for clinical information systems". In: *Security and privacy, 1996. proceedings., 1996 ieee symposium on*. IEEE, pp. 30–43.

Anderson et al. (2009). "Database State: A Report Commissioned by the Joseph Rowntree Reform Trust Ltd". In:

Bashir, I. (2017). *Mastering Blockchain*. Packt Publishing Ltd.

Calder, A. (2016). *EU GDPR A Pocket Guide*. IT Governance Ltd.

Carvel, J. (2017). "Concern over NHS's IT systems after 50 view celebrity's details". In: *The Guardian*.

De Telegraaf (2018). "Barbie met spoed naar ziekenhuis gebracht". In: *De Telegraaf*.

Desmedt, Y. and Y. Frankel (1991). "Shared generation of authenticators and signatures". In: *Annual International Cryptology Conference*. Springer, pp. 457–469.

Douceur, John R (2002). "The sybil attack". In: *International workshop on peer-to-peer systems*. Springer, pp. 251–260.

Ekblaw, A. et al. (2016). "A Case Study for Blockchain in Healthcare:"MedRec" prototype for electronic health records and medical research data". In: *Proceedings of IEEE Open & Big Data Conference*. Vol. 13, p. 13.

European Society of Radiology (2017). "The new EU General Data Protection Regulation: what the radiologist should know". In: *Insights into imaging* 8.3, pp. 295–299.

Felix, I. et al. (2018). *Praktijkproef blockchain kraamzorg met Mijn Zorg Log*.

Gennaro, Rosario, Steven Goldfeder, and Arvind Narayanan (2016). "Threshold-optimal DSA/ECDSA signatures and an application to Bitcoin wallet security". In: *International Conference on Applied Cryptography and Network Security*. Springer, pp. 156–174.

Gennaro, R. et al. (1996). "Robust threshold DSS signatures". In: *International Conference on the Theory and Applications of Cryptographic Techniques*. Springer, pp. 354–371.

Gillum, Richard F (2013). "From papyrus to the electronic tablet: a brief history of the clinical medical record with lessons for the digital age". In: *The American journal of medicine* 126.10, pp. 853–857.

Green, Marc and Thomas Eisenbarth (2015). "Strength in Numbers: Threshold ECDSA to Protect Keys in the Cloud". In: *IACR Cryptology ePrint Archive* 2015, p. 1169.

HagaZiekenhuis (2016). "HagaZiekenhuis stapt succesvol over naar EPD HiX". In:

Hassey, Alan, David Gerrett, and Ali Wilson (2001). "A survey of validity and utility of electronic patient records in a general practice". In: *Bmj* 322.7299, pp. 1401–1405.

Kiayias, A . et al. (2017). "Ouroboros: A provably secure proof-of-stake blockchain protocol". In: *Annual International Cryptology Conference*. Springer, pp. 357–388.

Kostkova, P . et al. (2016). "Who owns the data? Open data for healthcare". In: *Frontiers in public health* 4, p. 7.

Li, N., T Li, and S Venkatasubramanian (2007). "t-closeness: Privacy beyond k-anonymity and l-diversity". In: *Data Engineering, 2007. ICDE 2007. IEEE 23rd International Conference on*. IEEE, pp. 106–115.

Maymounkov, Petar and David Mazieres (2002). "Kademlia: A peer-to-peer information system based on the xor metric". In: *International Workshop on Peer-to-Peer Systems*. Springer, pp. 53–65.

McCall, Becky (2018). *What does the GDPR mean for the medical community?*

Nakamoto, S. (2008). "Bitcoin: A peer-to-peer electronic cash system". In:

Neal, Richard D, Philip L Heywood, and Stephen Morley (1996). "Real world data—retrieval and validation of consultation data from four general practices". In: *Family Practice* 13.5, pp. 455–461.

Otte, P. et al. (2017). "TrustChain: A Sybil-resistant scalable blockchain". In: *Future Generation Computer Systems*.

Paar, Christof and Jan Pelzl (2009). *Understanding cryptography: a textbook for students and practitioners*. Springer Science & Business Media.

Ponemon Institute (2016). *Sixth Annual Study on Privacy and Security of Healthcare Data*.

Presser, L. et al. (2015). "Care. data and access to UK health records: patient privacy and public trust". In: *Technology Science* 2015081103.

Shamir, Adi (1984). "Identity-based cryptosystems and signature schemes". In: *Workshop on the theory and application of cryptographic techniques*. Springer, pp. 47–53.

Solat, S. (2017). "RDV: Register, Deposit, Vote: a full decentralized consensus algorithm for blockchain based networks". In: *arXiv preprint arXiv:1707.05091*.

Spagnuelo, Dayana and Gabriele Lenzini (2016). "Patient-centred transparency requirements for medical data sharing systems". In: *New Advances in Information Systems and Technologies*. Springer, pp. 1073–1083.

Stange, Kurt C et al. (1998). "How valid are medical records and patient questionnaires for physician profiling and health services research?: A comparison with direct observation of patient visits". In: *Medical care*, pp. 851–867.

Szydlo, M. (2004). "Merkle tree traversal in log space and time". In: *International Conference on the Theory and Applications of Cryptographic Techniques*. Springer, pp. 541–554.

Tobin, A. and D. Reed (2016). "The Inevitable Rise of Self-Sovereign Identity". In: *The Sovrin Foundation*.

Vukolić, M. (2015). "The quest for scalable blockchain fabric: Proof-of-work vs. BFT replication". In: *International Workshop on Open Problems in Network Security*. Springer, pp. 112–125.

Wood, Gavin (2014). "Ethereum: A secure decentralised generalised transaction ledger". In: *Ethereum project yellow paper* 151, pp. 1–32.

World Economic Forum (2012). *Rethinking personal data: A new lens for strengthening trust*.

Yue, X. et al. (2016). "Healthcare data gateways: found healthcare intelligence on blockchain with novel privacy risk control". In: *Journal of medical systems* 40.10, p. 218.

Zyskind, G., O. Nathan, and A. Pentland (2015). "Enigma: Decentralized computation platform with guaranteed privacy". In: *arXiv preprint arXiv:1506.03471*.

Zyskind, G., O. Nathan, et al. (2015). "Decentralizing privacy: Using blockchain to protect personal data". In: *Security and Privacy Workshops (SPW), 2015 IEEE*. IEEE, pp. 180–184.