WU
WIRTSCHAFTS
UNIVERSITÄT
WIEN VIENNA
UNIVERSITY OF
ECONOMICS
AND BUSINESS

EQUIS
ACCREDITED

Bachelor Thesis

Towards a more privacy-aware social web: Challenges and opportunities for decentralized web platforms

Stefan Reithofer

Subject Area: Information Business

Studienkennzahl: J033/561

Supervisor: Dr. Sabrina Kirrane

Date of Submission: 21. November 2019

Department of Information Systems and Operations, Vienna University of Economics and Business, Welthandelsplatz 1, 1020 Vienna, Austria

DEPARTMENT FÜR INFORMATIONS-
VERARBEITUNG UND PROZESS-
MANAGEMENT DEPARTMENT
OF INFORMATION SYSTEMS AND
OPERATIONS

Contents

List of Tables

List of Figures

"Arguing that you don't care about the right to privacy because you have nothing to hide is no different than saying you don't care about free speech because you have nothing to say."

Edward Snowden, Former National Security Agency subcontractor

List of acronyms

ABE	Attribute Based Encryption
ACL	Access Control List
AES	Advanced Encryption Standard
API	Application Programming Interface
CA	Certificate Authority
DAG	Directed Acyclic Graph
dBFT	Delegated Byzantine Fault Tolerance
DDoS	Distributed Denial of Service
DES	Data Encryption Standard
DHT	Distributed Hash Tables
DLT	Distributed Ledger Technologies
GAFA	Google, Amazon, Facebook and Apple
HREF	Hypertext Reference
HTML	Hypertext Markup Language
HTTP	Hypertext Transfer Protocol
IaaS	Infrastructure as a Service
JSON	JavaScript Object Notation
JWA	JSON Web Algorithms
JWE	JSON Web Encryption
JWS	JSON Web Signature
JWT	JSON Web Token
LDP	Linked Data Platform
LDPC	Linked Data Platform Containers

LDPR	Linked Data Platform Resources
LDW	Linked Data Web
N3	Notation 3
OAEP	Optimal Asymmetric Encryption Padding
P2P	Peer-to-Peer
P3P	Platform for privacy preferences
PaaS	Platform as a Service
PDS	Personal Data Stores
PKI	Public Key Infrastructure
PoC	Proof-of-Capacity
POD	Personal Online Data Store
PoET	Proof of Elapsed Time
PoS	Proof-of-Stake
PoW	Proof-of-Work
RDF	Resource Description Framework
RSA	Rivest–Shamir–Adleman
SaaS	Software as a Service
SME	Small and Medium-sized Enterprises
SMTP	Simple Mail Transfer Protocol
SSL	Secure Sockets Layer
TLS	Transport Layer Security
URI	Uniform Resource Identifiers
W3C	World Wide Web Consortium
WAC	Web Access Control
WICG	Web Incubator Community Group

Abstract

When Tim Berners-Lee laid the foundation for the World Wide Web back in 1989 it marked a paradigm shift in the exchange of information. Besides all the positive implications in terms of economic growth and shared knowledge, the evolution of the internet took a disputable course over the last two decades. Centralized services have allowed for tremendous amounts of data to be stored by a few corporate-controlled mega-platforms. As data generated by social networking applications, banking platforms, medicine or insurance services is getting more and more sensitive, it raises significant concerns regarding privacy and transparency in data processing. In this thesis, we provide a comparative analysis of three different decentralization initiatives, namely Solid, Digi.me and Mastodon. We examine them in terms of their key technical and non-technical challenges in order to demonstrate the difficulties and opportunities with respect to the development of decentralized social web platforms.

1 Introduction

Nowadays more than 3.1 billion people are connected to the internet [51]. The ubiquity of online services and the ever advancing computational power and storage space had led to an unprecedented amount of data. This data is not only collected, but is regularly analyzed to gain valuable information which leads to innovation and economic growth [143]. Despite the many benefits a data-driven society provides to its users, there are always two sides of the coin. The value created through personalized services and recommendation systems, whose algorithms operate on large amounts of data input, are often in contrast to violations of user privacy in the form of information dissemination, data exposure or improper usage.

The initial idea of the World Wide Web was to create a decentralized, global network that allows everyone to share any type of resource with the world [126]. As the early web advocate John Perry Barlow stated, *"We are creating a world where anyone, anywhere may express his or her beliefs, no matter how singular, without fear of being coerced into silence or conformity"* [9]. Ibanez et al. [64] identified two fundamental principles to capture the vision of the internet: On the one hand, decentralization allows everyone to share anything on the web without permission from a central authority; and on the other hand, common standards such as the Hypertext Transfer

Protocol (HTTP)[1] used for communication between web servers and clients, provide universality and thus, interoperability for the involved actors. However, the web has become increasingly centralized and the success of most of the widely used web applications today is generated by avoiding these principles. This may be explained partly due to the ease with which developers were able to build these systems, and partly due to the fact that these platforms attracted advertiser's attention on a large scale. Reinforced by network effects and market forces, natural monopolies formed, resulting in the centralization of consumers, data, and finally corporate-controlled power [10]. Processing data on centralized servers and thus creating closed silos empowered a few mega-platforms such as Google, Amazon, Facebook and Apple (GAFA) to have alarming control over operations and user data [126].

More than 3.5 billion search queries per day on Google and over 1.2 billion active users daily on Facebook has made these platforms accountable for 80 percent of all incoming traffic to online news sources in the United States [8]. These, de facto monopolies make it hard for innovative competitors and especially Small and Medium-sized Enterprises (SME) to enter markets such as the social media market [57], and impede the ability of consumers to choose a different platform [62]. The success of the underlying business models is highly correlated with the importance of data in todays time. Targeted advertising, which is by far the most important revenue source of Facebook, is based on tracking user behaviour not only out of Facebook activities but also of activities which take place on affiliated third party sites (e.g. web-browsing histories or shopping habits) [67]. Although many of these data sharing agreements have been made public, the details specifying how, what and to which extent data is shared between the companies has remained secret, resulting in a constant uncertainty of consumers about where surveillance begins and where it ends [36].

Facebook has collected over 300 petabytes of personal data since its inception in 2004 [143]. This corresponds to 300 million gigabytes of valuable information on the user's interests and habits. As a result, the GAFA find themselves in a position which enables them to have a serious impact on individuals' behaviour. Content Nudging, if done properly, can affect consumer behaviour in a subtle way without restricting the user's freedom of choice [85]. By placing marketing content on well-suited touchpoints, Facebook and others can influence the decision-making process of users in favour of advertisers. Our individual search manner leveraged by recommendation algorithms often results in filter bubbles that can cause a state of intellec-

[1] https://tools.ietf.org/html/rfc2616

tual isolation [115]. The isolated information and beliefs are often reinforced and amplified by repetition inside closed silos. In media this phenomenon is referred to as echo chambers. Thus, users often become victims of confirmation bias [80]. A systematic error of inductive reasoning due to the fact that we tend to consume information based on confirmation factors that reinforce our preexisting views and beliefs.

However, the surveillance of social media activity [124] and the collection of personal data do not only harm privacy of individuals. Using digital footprints and social media activities to systematically derive psychographic profiles even has the potential to influence major political and democratic events. The Facebook-Cambridge Analytica data scandal in which information from millions of Facebook profiles was collected without the user's consent and used for political advertising purposes, led to a dismay in larger contexts and a watershed moment for the public understanding of data privacy. According to the report 'Assessing Russian Activities and Intentions in Recent US Elections' [137] published by the US Office of the Director of National Intelligence, millions of social networking records on various platforms like Twitter or Facebook were influenced in favour of Donald Trump in the run-up to the 2016 elections. In fact, many commentators believe that Donald Trump would not have won the 2016 elections without the influence of social media or fake news [4].

Several investigations have shown that their were Cambridge Analytica interventions in the run-up to the UKs Brexit vote as well [23, 24]. The systematic psychographic profiling in correlation with Facebook activities allowed Cambridge Analytica to derive personality profiles, religious affiliation, sexual orientation, intelligence or political views based on users' digital footprints [65]. Once someone is in possession of this information, they could actively manipulate the voting behaviour of individuals by placing well-directed, political advertising. This well-directed disinformation, intended or not, put Facebook in a position to even have an impact on our democracy. The influence on an individual's behaviour leveraged by the absence of transparency in data processing practises, and a constantly present uncertainty about information disclosure (such as huge amounts of recently exposed phone numbers [136] or millions of passwords stored in plaintext [93]) lead to substantial issues with todays web.

These problems have been present for many years, and many proposals such as WebBox [127], Musubi [40] or efforts from the Diaspora Foundation[2] aimed to re-decentralize the social web. However, none of these projects

[2] https://diasporafoundation.org/

succeeded in their attempt to be a genuine alternative to centralized silos. Reason enough to take a deeper look into the characteristics of decentralized personal data architectures and in particular decentralized social web applications. From 2015 onwards, platforms like Solid[3], Digi.me[4], or Mastodon[5], started to gain traction and brought new hope to make the web a more self determining and better place. In this thesis, we will exemplify some of the most challenging obstacles these platforms face. Furthermore, we will highlight the implications on user privacy that arise with a more transparent and decentralized data processing paradigm.

The remainder of this thesis is structured as follows: *Chapter 2* outlines the background, the research questions and describes the chosen research approach as well as the various stages of the review process. *Chapter 3* presents an overview of centralized, decentralized and distributed system architectures. *Chapter 4* provides a taxonomy of data privacy and further outlines some of the most important techniques that are used to decentralize the social web. *Chapter 5* presents the initiatives of Solid, Digi.me and Mastodon in terms of their technical functionality and further provides a discussion based on the status quo and some key challenges of these initiatives. Finally, *Chapter 6* concludes the thesis.

2 Research methodology and background

This chapter outlines the research background covered during the review process. Moreover, we briefly present the research questions and provide our approach to the research work used to answer the posed questions.

2.1 Re-decentralize the social web: A brief historic overview

The first attempts to decentralize the social web reach back to the late 1990s and started with the so-called 'negotiated privacy techniques'. The concept of infomediary initiatives was shaped by the 1999 book Net worth [54] by John Hagel and Marc Singer and aimed to connect users and commercial entities such as marketers. Information intermediaries act as data brokers, which provide consumers with the possibility to maximize the value of their

[3] https://solid.inrupt.com/

[4] https://digi.me/

[5] https://mastodon.social/about

data [26]. A mix of private contracts between consumers and commercial entities, and decentralized data storage was supposed to solve privacy problems. However, within half a decade all commercial efforts such as Lumeria [77], AllAdvantage [87], PrivaSeek [33] or Persona [7] failed [91]. Their approach was to enhance privacy by giving the user the possibility to decide which service provider gets access to what kind of personal information. The Platform for privacy preferences (P3P)[6], started by the World Wide Web Consortium (W3C), was seen as the most promising community initiative. Similar to their commercial counterparts, the P3P aimed to enable users to set privacy preferences over data processing practises of different web sites. However, the lack of trust users had in such infomediaries and the difficult and convoluted use prevented the success of this projects [96, 110].

Around the year 2008, a series of alarming privacy mishaps by Google and Facebook were made public [111], leading to a flood of simultaneous approaches for distributed social networks. Nearly all of them build on the combination of a distributed approach to store user information in Personal Data Stores (PDS) or personal data servers, and access control and encryption techniques such as Attribute Based Encryption (ABE) or Public Key Infrastructure (PKI) [39]. Diaspora [15], as perhaps one of the most well known projects, enables users to act as local servers, and thus, keeping control over their personal data. In Safebook [116] and My3 [92], privacy is achieved by selecting a set of trusted friends, each storing content with a different level of trust. PeerSoN [21] deals with privacy issues using access control and encryption techniques combined with a Peer-to-Peer (P2P) effort to supplant the centralized service. LifeSocial.KOM [49] and Cachet [95] use Distributed Hash Tables (DHT) to store and replicate data in a cryptographically secured way.

Many more efforts such as Likir [2], LotusNet [3], SuperNova [109] or SOUP [74] focused on privacy-aware social networking services. The WebBox project by Van Kleek et al. [112, 126, 127] presented an approach for tackling fragmentation among applications by focusing on personal data lockers. Thus, providing unified personal data spaces under the user's control, instead of storing data in centralized online silos.

[6] https://www.w3.org/TR/NOTE-P3P-CACM/

In light of the above stated problems and the conducted background research, we propose the following research question, which can be further divided into three subsequent questions:

> **What key challenges and opportunities arise when it comes to decentralized data processing on social web platforms?**
>
> *- What techniques can be used to facilitate authentication, access control and encryption in decentralized platforms?*
> *- What platforms are available that offer users more control with respect to personal data processing?*
> *- How do these platforms differ in terms of their approach to authentication, access control and encryption?*

2.2 Research methodology

In order to answer the aforementioned research questions, a systematic literature review is used to acquire an in-depth understanding of the discussed topics. Higgins et al. [59] stated the need for a systematic review as follows:

> *"A systematic review attempts to collate all empirical evidence that fits pre-specified eligibility criteria in order to answer a specific research question. It uses explicit, systematic methods that are selected with a view to minimizing bias, thus providing more reliable findings from which conclusions can be drawn and decisions made."*

According to the guidelines described by Kitchenham and Charters [73], the three main stages contain: (i) Planning the review; (ii) Conducting the review; and (iii) Reporting the review. All of them are outlined in the subsequent sections more detailed.

2.2.1 Planning the review

In order to gain as much understanding of the relevant topics and papers and in order to be able to select the proper literature, we developed a review protocol (see *Table 1*). The protocol aims to outline the criteria we used to select the most useful and appropriate literature. The search process has mainly comprised the keywords *decentralized web, personal data, privacy, decentralization, information security* and *decentralized identity.* We used three online databases, namely: (i) Google Scholar; (ii) ProQuest; and (iii) DBLP. Google Scholar offers one of the largest digital databases for peer-reviewed,

Review protocol		
Stage of the review process	Description	Number of papers left
Stage 1: Discovery and data extraction	Discover online databases for relevant literature. Criteria for the selection: *(i) the title; (ii) the keywords of the given paper; and (iii) the keywords of the citing papers.*	61
Stage 2: Selection criteria	Set language criteria for the discovered literature to English	58
Stage 3: Selection criteria	Set document type for the discovered literature to journal, conference or workshop paper	52
Stage 4: Abstract evaluation	Evaluate papers based on the abstract. Criteria for the selection: *(i) the conceptual association to the relevant topics; (ii) the temporal relevance and topicality; and (iii) a suitable level of abstraction.*	42
Stage 5: Full paper evaluation	Evaluate papers based on full reading. Criteria for the selection: *(i) the conceptual association to the relevant topics; (ii) the temporal relevance and topicality; and (iii) a suitable level of abstraction.*	36

Table 1: Stages of the review process

Search syntax on online databases	
Data Source: End of May 2019	Applied search syntax
(i) Google Scholar database (https://scholar.google.com/) - Keywords used: 'decentralized web' and 'personal data' and 'privacy'	'decentralized web AND personal data AND privacy'; data range: since 2008; language: English
(ii) DBLP database (https://dblp.uni-trier.de/) - Keywords used: 'personal data' and 'information security' and 'data privacy'	'personal data' AND 'data privacy'; 'information security' AND 'personal data'
(iii) ProQuest database (https://search.proquest.com/) - Keywords used: 'information security' and 'personal data' and 'privacy'	(TITLE-ABS-KEY('information security') AND TITLE-ABS-KEY('personal data') AND TITLE-ABS-KEY('privacy')) AND (LIMIT-TO(LANGUAGE,'English')) AND (LIMIT-TO(DATE,'Last10years')) AND (LIMIT-TO(DOCTYPE,'article') OR LIMIT-TO(DOCTYPE,'conferencepaper') OR LIMIT-TO(DOCTYPE,'reviewpapers'))

Table 2: Applied search syntax on digital databases

multidisciplinary conference and academic journal papers, and provides an in-build ranking algorithm. It was thus used for extracting the maximum number of useful and relevant papers. In addition, we considered the bibliographies of DBLP and ProQuest in order to discover literature that is not listed on Google Scholar.

2.2.2 Conducting the review

The protocol outlined in *Table 1* serves as a starting point for the search syntax presented in *Table 2*. The keywords we used for the search on Google Scholar were mainly *decentralized web, personal data* and *privacy.* These were searched for in the abstract, keywords, and in the title. We set the `AND` operator to discover only papers in which all of the used keywords appear. On top of that, we applied different compositions of keywords in an trial search-manner to enhance the outcomes of the selection process. To further vary the search, we adjusted the keywords on DBLP and ProQuest to various combinations of *information security, personal data* and *privacy.*

Document Type/Year	2019	2018	2017	2016	2015	2014	2013 and older	Total	%
Conference Paper	1	2	2	2	1	1	4	13	25.5%
Journal Paper	1	4	3	3	3	1	6	21	41.2%
Workshop Paper	-	-	-	1	-	1	-	2	3.9%
Whitepaper/Specification/ Product documentation	-	2	-	1	1	6	5	15	29.4%
Total	2	8	5	7	5	9	15	51	100%

Table 3: Document type per year

2.2.3 Reporting the review

Given the search syntax for Google Scholar, 52 papers were considered as relevant for the further review process. Google Scholar listed several thousand publications depending on the keyword combination. Therefore, a first, rough evaluation has been conducted based on the following criteria: (i) the title; (ii) the keywords of the given paper; and (iii) the keywords of the citing papers. Given the search syntax shown in *Table 2*, only nine papers listed on ProQuest or DBLP, which were not discovered on Google Scholar, were considered as relevant for further review steps. This left us with 61 papers in total after the first stage of the review process. In the second part we restricted the selection to papers written in English. This left us with 58 publications. Moreover, we restricted the extracted literature to journal, conference or workshop papers. This left us with 52 papers, which were evaluated based on the abstract and on full reading. The criteria chosen for the selection was: (i) the conceptual association to the relevant topics; (ii) the temporal relevance and topicality; and (iii) a suitable level of abstraction. Based on these criteria, we assessed 36 of the papers as relevant for our research work.

Papers such as [20], [44] or [114] were excluded and not analysed in more detail because they may correlate with the field of interest but do not meet all of the stated criteria. Due to criteria (ii), papers such as [50] or [135] were excluded as they are outdated and do not include the most recent information regarding the specific topics. In regards to criteria (iii), papers such as [61] or [102] were not considered as they go beyond the scope of this thesis. To acquire even more knowledge on the relevant topics, we also used first-class online sources, for instance, W3C recommendations and articles published in the wake of the Decentralized Web Summit. *Table 3* shows, that 25% of the publications were conference papers, while 41% of the included studies were journal papers. Only two of the relevant publications were workshop papers.

	Keywords								
Author (year of publication)	Decentralization	Decentralized Web	Linked Data	Peer-to-peer systems	Blockchain	Distributed Ledgers	RDF	Semantic Web	Decentralized Social Networks
Kim et al. (2019) [69]					✓				
Machado et al. (2018) [83]			✓					✓	
Halpin et al. (2018) [55]	✓				✓				
Zignani et al. (2018) [142]									✓
De Salve et al. (2017) [39]									✓
Tronocoso et al. (2017) [123]	✓			✓					
Ibanez et al. (2017) [64]		✓							
Kirrane et al. (2017) [71]			✓				✓	✓	
Third et al. (2017) [119]			✓		✓	✓			
Chakravorty et al. (2017) [28]					✓				✓
De Filippi et al. (2016) [38]					✓				
English et al. (2016) [42]			✓		✓			✓	
Faisca et al. (2016) [44]	✓				✓			✓	
Sambra et al. (2016) [107]	✓								
Van Kleek et al. (2015) [126]		✓							
Chowdhury et al. (2015) [30]				✓					
Zyskind et al. (2015) [143]					✓				
Vogel et al. (2015) [131]					✓				
Sambra et al. (2014) [106]	✓		✓						
Nilizadeh et al. (2012) [95]				✓					
Van Kleek et al. (2012) [127]			✓					✓	
Bizer et al. (2011) [16]			✓					✓	
Kapanipathi et al. (2011) [68]								✓	✓
Cutillo et al. (2009) [34]				✓					
Buchegger et al. (2009) [20]				✓					
Cutillo et al. (2009) [35]				✓					

Table 4: Literature classification based on keywords

Whitepapers, specifications and product documentations complemented our research. This was even more important because of the fact that the information on the discussed protocols and standards was at some point limited when it comes to academic papers. Including W3C specifications or product documentations thus allowed us to gain more information on even recently published efforts.

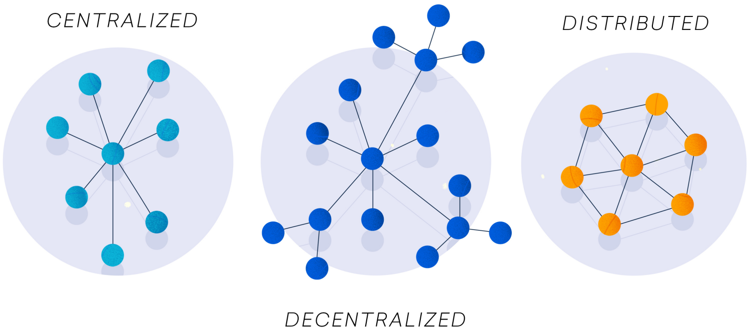

Figure 1: Taxonomy of system architectures [103]

3 Taxonomy of network architectures

In this chapter, we present an overview of existing network architectures and highlight the technical differences between centralized, decentralized and distributed systems. In this regard, we also outline the concepts of Linked Data and the Blockchain technology. *Table 4* provides a classification of the literature which we considered in particular for this chapter. With regard to Linked Data and the Semantic Web, several W3C specifications or product documentations complemented our research.

As shown in *Figure 1*, centralized architectures rely on a single, trusted party to facilitate operations, data storage and network maintenance. These systems are built around single servers, which perform all major coordination and processing tasks in order to keep the system going. Centralized service providers such as Facebook control all data flows and operations inside the network in a client-server manner, as well as the servers hosting all of the user's content. In contrast, data processing and control in decentralized systems is not handled by a single entity, but instead shared among multiple independent machines.

While centralized and decentralized systems may be distinguished regarding the control over the network, the distribution of the network refers to the distinction of locations. Parts of the system in a distributed network are spread across multiple physical locations. Distributed Ledger Technologies (DLT) such as Blockchain or Directed Acyclic Graph (DAG) DLTs build on top of that. By providing a database that is replicated, stored and updated by each node or computing device within the network, these systems can live without centralized servers maintaining them [101]. The Bitcoin network for instance is: (i) decentralized, as the transaction record can not be altered by a single entity; and (ii) distributed, as it is spread across a

P2P network full of independent nodes. A cloud storage provider such as Dropbox is: (i) distributed, as users can physically share and replicate their data on multiple computing devices; and (ii) centralized, as at the end of the day, Dropbox controls all of them.

3.1 Centralized data silos

The rise of centralized web services did not come by chance. Architectures based on a single point of data processing and control encompass various technical advantages. Single machines mean lower hosting, development and maintenance costs, resulting in remarkable economies of scale. Moreover, they are not forced to utilize standardization of technical protocols to facilitate interoperability between heterogeneous systems like decentralized systems do. Thus, enabling them to better capitalise on network effects for their own benefit [92]. In addition, being able to efficiently patch and update system functionalities over a single central server requires fewer system administrators and less effort regarding IT management, resulting in a more affordable network infrastructure.

Nevertheless, relying on centralized platforms to store, maintain and transfer user data comes with several drawbacks. Central servers are by nature more vulnerable to technical, legal or regulatory attacks [100]. Moreover, the fact that these corporations often control the data access APIs and regularly change its features, makes it difficult for developers to create applications that can run not only on the specific platform but also on multiple ones [107]. Thus, the lack of interoperability affects the user as they are neither able to smoothly move data between different platforms nor to switch easily between applications that could potentially retrieve and reuse data from similar applications [107].

As a prime example for centralized services, cloud-based Software as a Service (SaaS) solutions gained massive traction over the last several years. The term cloud computing describes a novel computational paradigm and covers various services (such as software, storage space or computing power) delivered over the internet. SaaS, as one form of cloud computing, is a model to distribute software by hosting applications on the cloud infrastructure and making them available as internet-based services, instead of having them installed on the customers' local machines [5]. As a result, SaaS providers such as SalesForce CRM can drastically reduce the expense of hardware acquisition, software licensing or installation. Besides SaaS, cloud computing also contains the service delivery models Platform as a Service (PaaS) and

Infrastructure as a Service (IaaS). In a PaaS scenario, customers can develop, run and manage their applications on platforms and tools delivered by cloud service providers without the need to install them on their local machines. IaaS providers such as Amazon EC2 deliver storage space, network, or computation resources over the internet.

Cloud-based services offer massive benefits in terms of cost reduction, flexibility, and vertical scalability, which allows customers to easily scale up or down services based on their business demand [25]. However, with cloud platforms providing their solutions as internet-based services, novel dimensions have entered into the scope of security and privacy-related problems. Al Morsy et al. [5] as well as Hussain et al. [63] discussed various security implications for each of the three service delivery models. Bhadauria & Sanyal [14], and Gellman [48] highlighted how security threats could harm privacy in cloud platforms.

3.2 Decentralized networks and Linked Data

According to Narayanan et al. [90], there are different factors that together make an architecture decentralized. First, whether the data is hosted centrally, locally on the user's device, distributed on a P2P network or on a third party's hardware. Second, if data portability in a given architecture requires any type of Application Programming Interface (API) or not. Lastly, whether implementation of applications and development are done based on open standards and open-source technology approaches, or proprietary software. The key distinction to a centralized system, the absence of a single point of failure, is at the same time the most significant benefit of decentralized architectures, thereby making the system more reliable and resistant to various classes of attacks. In addition, decentralized systems can also be better scaled by providing more computational power through simply adding more machines.

The emergence of Semantic Web technologies, Linked Data initiatives and the Resource Description Framework (RDF) have shed new light on re-decentralizing the Social Web [19]. Linked Data aims to provide a decentralized Web of Data, in which data is linked to various other data sets from different sources by using machine readable formats [16]. The Internet as we know it can be seen as a web of linked documents. The Hypertext Markup Language (HTML) sets relationships and connections through hyperlinks in hypertext documents (i.e. web sites), which cross-refer to each other. Thus, machines are mainly used to deliver and present the information included in those documents. The lack of interpretation and context requires humans to use background knowledge to derive facts of incomplete information and to draw connections. The Semantic Web aims to solve this by allowing us to share and reuse data across application boundaries[7]. At the core of this effort stands the RDF data model, which represents data in a graph form by building subject-predicate-object triples. RDF data is presented in common standardization formats such as RDF/XML, Notation 3 (N3), Turtle or JSON-LD. Based on the RDF data format, resources can be described in a way that enables them not only to be linked, but also describe relations between them [70]. To achieve this across application boundaries, common vocabularies ('schemas' or 'ontologies') such as the FOAF[8] vocabulary provide a dictionary of basic terms to describe the things included in FOAF documents.

Instead of a global information space by linking HTML documents, the vision of Linked Data is to reframe the web in a way that published data is added to a global data space in a natural way. Resources in the Linked Data Web (LDW) can be identified by using HTTP(S) Uniform Resource Identifiers (URI). According to Heath and Bizer [58], URIs can not only be used to refer to resources, but also to link those resources similar to the way in which the Hypertext Reference (HREF) attribute in HTML links web documents. In order to realize such a LDW, Tim Berners-Lee proposed a set of rules [12], commonly referred to as the *Linked Data Principles*:

> " 1. Use URIs as names for things.
>
> 2. Use HTTP URIs so that people can look up those names.
>
> 3. When someone looks up a URI, provide useful information, using the standards (RDF*, SPARQL).
>
> 4. Include links to other URIs so that they can discover more things. "

In order to efficiently read, write and modify data in a Web of Linked Data, the W3C proposed the Linked Data Platform (LDP) specifications[9]. Linked Data Platform Resources (LDPR) represent the minimum data granularity such as an event in a calendar application. Similar to a traditional file system hierarchy, LDPRs can be combined to Linked Data Platform Containers (LDPC). LDPCs represent collections of linked resources and own, just like LDPRs do, HTTP URIs. These URIs are used to identify, detect and address web resources. Furthermore, LDP defines RESTfully HTTP opera-

[7] https://www.w3.org/2001/sw/

[8] http://xmlns.com/foaf/spec/

[9] https://www.w3.org/TR/ldp/

tions between clients that send HTTP requests and servers that send HTTP responses in order to service those requests, such as HTTP GET to access a resource or HTTP POST to create a LDPR to a LDPC by referencing to its HTTP URI.

Although we made a good stride towards standardization through common standards such as RDF[10], decentralized systems still pose challenges. For instance, they cannot react and roll out new features as fast as Facebook and other centralized entities can do [92]. Furthermore, the slower decision making process and the duplication of work could harm the efficiency of such architectures [1]. Another drawback may arise on a cognitive level as well [91]. Having control over your personal data through a decentralized systems mostly requires a more conscientious and diligent handling of the required software. A lack of user's expertise in this field may result in security vulnerabilities and a cognitive overload. However, when it comes to personal data processing on social networking applications we face even more obstacles. For instance, updates propagation, an efficient mechanism that enables search and addressing or openness for third-party applications are challenges that often cause trade-offs, such as whether search quality or privacy should be prioritized [37].

3.3 Blockchain and Distributed Ledger Technology

With the Bitcoin white paper in 2008 [89], an unknown person or group of people under the pseudonym Satoshi Nakamoto introduced the Blockchain technology. While the concept of cryptographically linked blocks in an immutable data structure finds its origin in the early 1990s [52], Bitcoin is often referred to as the birth of Blockchain technology as it demonstrated a first real-world and profound use case. As Narayanan and Clark described *"This is not to diminish Nakamoto's achievement but to point out he stood on the shoulders of giants"* [90].

Blockchain is a distributed ledger system that stores data as sets of transactions. Each of these sets is associated to a specific block. Transactional data is not restricted to the transfer of cryptocurrencies such as Bitcoin, but can represent any transfer of digital value such as property ownership, shares or votes. The ledger containing the record of transactional data is spread across the P2P network and replicated on every node belonging to the network. For Blockchain to function as a decentrally maintained, and at the same time censorship resistant, tamper-proof and trustworthy ledger,

[10] https://www.w3.org/RDF/

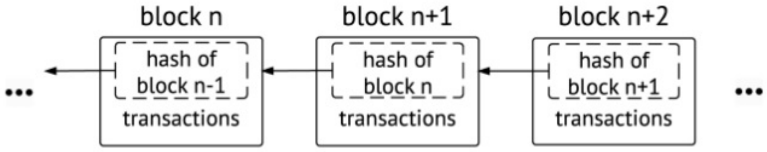

Figure 2: Blockchain data structure - hash pointers [31]

it builds upon multiple core mechanisms, namely: (i) cryptographic hash functions to confirm the Blockchain's current state; (ii) digital signatures to ensure for the correctness of placed transactions; and (iii) a distributed consensus mechanism [141]. Each of the blocks, except the genesis block (i.e. the first block in the chain), contains a cryptographic hash, which links back to the previous block (see also *Figure 2*). A hash function is used to apply a mathematical algorithm that maps data of an arbitrary size to a fixed-size bit string (often known as hash value or hash) [99]. The hashing algorithm SHA256, which is applied in the Bitcoin and Ethereum Blockchain, is a one-way function. The same is also true for any other cryptographic hash function. As a consequence, they are nearly infeasible to invert and it has not been proven possible to alter the content or properties of a given block. Tampering a historical block would require one to re-create the complete blockchain since the one block they are trying to manipulate [46].

Blockchain technology and distributed ledgers are often but mistakenly seen as something interchangeable. While all blockchains employ a chain of blocks to facilitate a distributed consensus mechanism, not all DLTs do so. Thus, every blockchain can be seen as a distributed ledger, but not every distributed ledger can be seen as a blockchain [101]. However, they do not differ in their major benefits. Both allow multiple nodes to enjoy the same privileges and access to information in a P2P network, and to transfer value without having to rely on a trusted intermediary to validate the transaction.

To agree upon the validity of new transactions in a P2P network full of independent nodes, there is a need for a distributed consensus mechanism. In Proof-of-Work (PoW) blockchains (such as Bitcoin) the block creation is facilitated by so called full nodes, or often referred to as miners [46]. A full node waits for a transaction to be placed, spreads it to other miners and tries to generate a new block to which the transaction is then associated, by solving complex computational puzzles. In order to incentivise nodes to contribute their computing power, they receive rewards (for instance coins in the cryptocurrency world). This facilitates the decentralization of the blockchain network through the activeness of the participating nodes [28].

The most widely implemented alternative to PoW is currently Proof-of-Stake (PoS). In PoS blockchains such as Tezos[11], the validation of transactions and the creation of new blocks is executed by nodes that hold a specific amount of coins. Various other consensus algorithms such as Proof of Elapsed Time (PoET)[12], Delegated Byzantine Fault Tolerance (dBFT)[13], or Proof-of-Capacity (PoC)[14] exist. However, the taxonomy of Blockchain and currently applied consensus mechanisms are beyond the scope of this thesis. Therefore, we point to *Blockchain challenges and opportunities: a survey* [141] published by Zheng et al. for a detailed overview.

While the focus in the early days of blockchain was mainly on financial services such as the P2P cash system Bitcoin, efforts of industry and privacy advocates have expanded to various application fields. In 2016 the Steemit project [29] introduced the first blockchain-based social media platform[15]. Steemit allows users to share content that is stored in an immutable blockchain ledger on the Steem blockchain. Moreover, the token-based ecosystem enables users to gain advantages from actively participating in the network, awarding content creators and curators for publishing with digital tokens called Steem. Until October 2018 more than 1.5 million comments resulted in rewards worth over USD 40 million [69]. SocialX[16] is yet another community-driven, blockchain-based social media platform with a build-in token reward ecosystem. Striving for an autonomous and decentralized way of social interaction on the web, platforms such as Steemit and SocialX enforce the possibility to keep value within the network of participating users instead of allocating it to a few single entities.

Given an increasing adoption, the variety of use cases for distributed ledgers will raise several challenges on the path to a more decentralized web. Data is getting more diverse the more real world applications exist, which makes efficient querying of information considerably harder. Providing at least a low level of granularity for indexing information will enhance the capacity to search across multiple ledgers, enabling distributed systems for better usability and performance. Furthermore, one has to find a way for integrating distributed ledger data with data from heterogeneous systems as well as reconcile ledgers and existing technology stacks [119]. As the applica-

[11] https://tezos.com/

[12] https://blockonomi.com/proof-of-elapsed-time-consensus/

[13] https://www.cryptocompare.com/mining/guides/delegated-byzantine-fault-tolerance-dbft-generals-problem-explained/

[14] https://cryptoverse.fandom.com/wiki/Proof-of-capacity

[15] https://steemit.com/

[16] https://socialx.network/

tion of distributed ledgers appears to be organic due to the fact that different applications have different requirements in terms of privacy or security, an all-embracing, universal distributed ledger seems hard to achieve [64]. As Third and Domingue state [119], "[..] *there are needs to integrate these data with arbitrary and diverse external data sources and to integrate smart contracts with services available on the Web - in other words, there is a need for Linked Data*". A smart contract is a computerized transaction protocol that runs on top of the blockchain. They contain contractual agreements computerized in an if-then logic. Therefore, smart contracts are able to control the execution, enforce the terms of an contract and hence improve the functionalities of distributed ledger technology [43, 134].

The vision of Linked Data to provide a global data space by linking data from heterogeneous sources can potentially allow for smoother integration and interoperability of distributed ledger systems within a decentralized web. Third and Domingue [119] proposed an approach for a semantic Linked Data index for distributed ledgers. Using existing Semantic Web technology stacks and tools (e.g. iServe [98]) to efficiently query and retrieve smart contract information and data that is stored on distributed ledgers (in this case the Ethereum blockchain) as well as smoothly linking this information to various other data sources.

4 Authentication, access control and encryption for a decentralized web

This chapter presents a detailed overview of authentication, access control and encryption techniques. We further outline how these techniques can be used to enhance privacy in personal data processing. *Table 5* shows the relevant literature that served as a basis for this chapter. With regard to the topics of access control, authentication and cryptography, we complemented our research by product documentations (see for instance [56]), W3C specifications (such as [78]), and high-quality online sources (such as [86]).

4.1 A Motivating Scenario

Leon recently moved from his hometown Vienna to New York. To stay in touch with his family and friends left behind, Leon uses a new generation of social web platforms. Leon's web activities involve a wide range of personal and sensitive information. He posts personal pictures of his children

	Keywords							
Author (year of publication)	Privacy	Security	Access Control	Authentication	Cryptography	Personal Data	Social Web	Online Social Networks
Kirrane et al. (2018) [72]	✓	✓	✓					
Hadar et al. (2018) [53]	✓							
Gonzalez et al. (2018) [22]				✓				
Halpin et al. (2018) [55]							✓	
Thelwall et al. (2018) [118]							✓	
Tronocoso et al. (2017) [123]	✓							
Kirrane et al. (2017) [71]			✓					
Farooqi et al. (2017) [45]		✓		✓				✓
Tourani et al. (2017) [121]	✓	✓	✓					
De Salve et al. (2017) [39]	✓							
Faisca et al. (2016) [44]				✓				
Sambra et al. (2016) [107]							✓	
Van Kleek et al. (2015) [126]	✓							
Zyskind et al. (2015) [143]	✓					✓		
Chowdhurry et al. (2015) [30]								✓
Sambra et al. (2014) [106]	✓						✓	
Schwittmann et al. (2013) [108]	✓							✓
Nilizadeh et al. (2012) [95]	✓							✓
Van Kleek et al. (2012) [127]	✓							
Krupa et al. (2012) [75]	✓							
Miculan et al. (2011) [88]				✓				
Kapanipathi et al. (2011) [68]	✓						✓	
Takabi et al. (2010) [117]	✓	✓						
Zhang et al. (2010) [140]	✓	✓						✓
Datta et al. (2010) [37]			✓					✓
Tootoonchian et al. (2009) [120]								✓
Cutillo et al. (2009) [34]	✓		✓		✓			
Buchegger et al. (2009) [20]	✓				✓			
Cutillo et al. (2009) [35]	✓							
Venter et al. (2003) [128]	✓	✓	✓		✓			

Table 5: Literature classification based on keywords

on his social media timeline, shares financial information from his banking applications in a private chat with his wife, or even publishes fitness activities containing biomedical parameters such as his body weight or pulse and respiratory rates in his social media profile on a weekly basis. Doing this, Leon does not have to be concerned that any party expect for the ones he explicitly gave authorization to, can see or access his personal data. These platforms allow him to: (i) allocate his personal data from different sources such as his social media profiles, banking accounts or health applications in one single place fully controlled by himself, instead of multiple corporate-controlled data silos; (ii) manage his digital identity in a decentralized way so he can use it for authentication purposes on multiple applications; (iii) send messages to his family and friends with the certainty that only they can read them and this in a non-modified way; and (iv) have full knowledge about what data is stored by application providers, how it is used, to whom it is passed and for how long it is retained, all coming in the form of a data usage policy.

Leon can be sure that, whenever he shares personal information with one of these applications, they only use it on his own predefined terms. Application providers who are obligated to adhere to the data usage policy are supposed to reveal all of their data collecting, sharing and processing practises and further capture it in a publicly accessible place, so that all parties can prove that Leon's data is used on the predefined terms.

In order to provide a secure and at the same time privacy-aware data processing in a decentralized setup and to allow for a scenario such as the one described above, we have to rethink several techniques, namely: (i) how are users identified and authenticated; (ii) how can we enforce distributed access control so that only authorized users can access specific resources; and (iii) how do we protect data from undesired modifications and apply encryption mechanisms to ensure end-to-end confidentiality. The following sections aim to define the terms of authentication, access control and encryption, which will be discussed in detail in this chapter [17, 60, 113].

> **Authentication.** In the field of information security, authentication describes the process of verifying the identity of a given thing (i.e. user, resource, artifact, etc.). For social web applications to provide confidentiality and trust, identification and authentication of all involved parties are an integral part. Users as well as service providers are ensured that their counterpart is indeed the one he claims to be. Moreover, by handling the recognition, validation and authentication of network participants, identity management is able to lay the foundation of confidential data access and utilization [6].

Access Control. Based on the privileges an entity is granted through the authentication process, one can set restrictions in terms of the access to a specific resource or place inside the system. On the one hand, users should have the certainty that their data is only seen and used by the company, individual or organization to which they explicitly expressed their consent. On the other hand, service providers must be able to define fine-grained permissions that limit the access to their services. The risk of breaches of confidentiality such as the public disclosure of personal information can thus be limited.

Encryption. The purpose of cryptographic mechanisms is to translate data into a different form, so that only actors that can prove the ownership of a secret key (i.e. decryption key) can access and read the data. Modification detection when combined with origin authentication can prevent uncertainty about potential changes to the data and thus, provide end-to-end confidentiality. As a consequence, the recipient is able to build on the sender's trustworthiness as well as on the integrity of the data. Furthermore, if tied to an actor's identity, encryption techniques can also provide non-repudiation as data processors can no longer deny any taken action regarding the processing of personal data.

4.2 Authentication

In earlier days, when service providers tried to gain access to user data from other applications, they asked for user credentials (i.e username and password) and logged in on the user's behalf. Let's say one wants to export their friends list from Facebook to their sports tracking app. They would have to provide the company behind the sports tracking app with their Facebook credentials without knowing what Facebook can or will do with them.

OAuth 2.0 [56] The OAuth protocol addresses these problems by performing identity verification and permission granting without exposing end-user credentials. In OAuth, the role of the resource owner (i.e. the end-user) is separated from those of the client (i.e. sports tracking app) requesting access to specific resources. Instead of accessing protected resources by logging in with the users resource server (i.e. Facebook) credentials, the client receives a temporary access token, which is issued by an authorization server (i.e. either the resource server, or preferably a separate entity) and approved by the resource owner. Tokens obtain various access attributes such as the scope or the lifetime, all granted by the end-user. Thus, third party applications can access protected resources from resource servers without the need to fetch user credentials.

OpenID Connect 1.0 [105] While the OAuth protocol builds on an authorization layer that allows for protection and access of resources, OpenID Connect 1.0 extends the OAuth protocol by providing an identity layer used for decentralized user authentication. Other than OAuth, OpenID is unaware of any existing resources but merely focuses on verifying that a user is the one he claims to be. The authentication process in OpenID is based on an ID Token. ID Tokens are represented in the form of a JSON Web Token (JWT) and include claims about the end-users authentication such as the subject identifier, the time of the authentication or the expiration time. The method for authenticating the end-user (e.g. username and password) is left to the OpenID Provider (i.e. an OAuth-2.0-authorization server that implements the OpenID Connect protocol). Once the user is authenticated and authorized, the OpenID Provider issues an ID Token and an Access Token to the client, who can then log in the user and access the desired resources from the resource server. The combination of OAuth and OpenID provides a single handshake, facilitating both user identification and data transfer [86].

WebID[17] is an openly extensible W3C standard, which allows agents (i.e. individuals, groups or even organizations) to create their own digital identity. At the heart of the WebID protocol stands the WebID profile document (i.e. a RDF-based web page in a Linked Data format (JSON-LD, Turtle etc.)). Besides basic information such as name, nickname or e-mail address, which is expressed by the FOAF[18] vocabulary, WebID profile documents also include information on public key certificates used for the authentication of users and a pointer to users root storage location (a detailed explanation of both is given in *Section 4.4* and *Section 5.1.2*, respectively). Every WebID profile document comes with an HTTP(S) URI, acting as an unique identifier of an agent by referring to its given profile document. On top of that, the WebID-TLS[19] specification describes distributed and privacy-enhancing authentication mechanisms that combine WebID profile documents and public key certificates in order to identify agents and further allow for fine-grained access control [70]. By using the FOAF vocabulary and specific FOAF properties, WebID can establish a Web of Trust, where services can

[17] https://www.w3.org/wiki/WebID

[18] http://xmlns.com/foaf/spec/

[19] https://dvcs.w3.org/hg/WebID/raw-file/tip/spec/tls-respec.html

implement authorization decisions based on the agents properties (for instance if an agent corresponds to a specific group, works at a specific company or knows specific people).

4.3 Access Control

Once a user has been identified and authenticated, we have to define what resources the given user is allowed to access and what kind of operations (read, write, append, etc.) the user is allowed to perform on these resources. As their is no centralized party that applies such restrictions, their is a need for distributed access control mechanisms.

WAC. The Web Access Control (WAC)[20] specification provides, when combined with WebID, such a distributed access control system. As explained in the previous section, users are identified by URIs, which point to their WebID profile. Resources and containers can be identified by HTTP URIs as well. The access control framework is based on the RDF vocabulary. RDF properties such as `accessTo` and `accessToClass` are used to specify resources, properties such as `agent` or `agentClass` are used to specify agents (i.e. users, organizations, etc.) [70]. The access mode is specified using classes such as `Read, Write, Control` or `Append`. Both Sacco and Passant [104], and Villata et al. [130] extended the RDF vocabularies for WAC. Using this extensions allows to apply access control mechanisms on individual RDF resources (subjects, predicates and objects), as well as on collections of RDF resources (i.e. named graph) [71]. As the WAC ontology is subsequently evolved for systems that implement LDP specifications, access control policies can be applied on a resource or container level, allowing for fine-grained access control.

ACL. By dereferencing the URI of a resource or container, the client is able to spot the Access Control List (ACL) attached to this specific resource. ACLs are separate WAC documents which reveal authorization statements on a given resource or container and come with their own URI. For example, *https://socialdata.com/images/.acl* would be the ACLs URI for the resource *https://socialdata.com/images/* and can be discovered by dereferencing the resource URI (i.e. HTTP GET/HEAD on the given resource). As both, WebID profile documents and ACLs are represented by Linked Data serialization formats (such as Turtle) and use the FOAF vocabulary, access control polices can be set

[20] https://github.com/solid/web-access-control-spec

based on the agents properties (for instance if an agent holds a specific `agentClass`). Thus, facilitating fine-grained access control on resource level, while allowing for attribute-based restrictions based on user properties (such as the `agentGroup` statement for granting access only to specific groups of users).

4.4 Encryption

In order to provide end-to-end confidentiality through distributed authentication mechanisms and the protection of data integrity, encryption techniques occupy an important position. We will focus on two parts: (i) digital signatures and the underlying public-key cryptography used to authenticate agents (i.e. users, organizations, etc.); and (ii) different data encryption techniques and algorithms which ensure that data integrity is preserved.

Data encryption techniques are supposed to enhance data integrity by protecting it from malicious or accidental modifications. Combining cryptographic mechanisms with other techniques such as time-stamping can consequently provide non-repudiation of processing steps. In general, there are two ways to cryptographically protect data: (i) symmetric encryption, meaning that both, encryption and decryption of the data block are done using the same key; and (ii) asymmetric encryption (often referred to as public-key encryption). The Data Encryption Standard (DES) was one of the first standardized techniques for symmetric-key encryption, developed in the early 1970s by IBM.

The issue with DES and the reason why it was withdrawn as a standard in 2005 is the short key length of only 56 bits. As the strength of the encryption techniques is proportional to the key length, shorter keys increase the plausibility of brute-force attacks [82]. Triple DES, the successor of DES, aimed to solve this issue by applying the cipher algorithm (i.e. translating the data block into encrypted form, named ciphertext) to each data block for three times. Thus, the total key size is 168 bits, instead of 56 bits. Besides Tripe DES, Advanced Encryption Standard (AES) is the widely adopted industry standard for symmetric-key algorithms at this point. In terms of asymmetric-key cryptography, the Rivest–Shamir–Adleman (RSA) cryptosystem has asserted itself for secure data transmission.

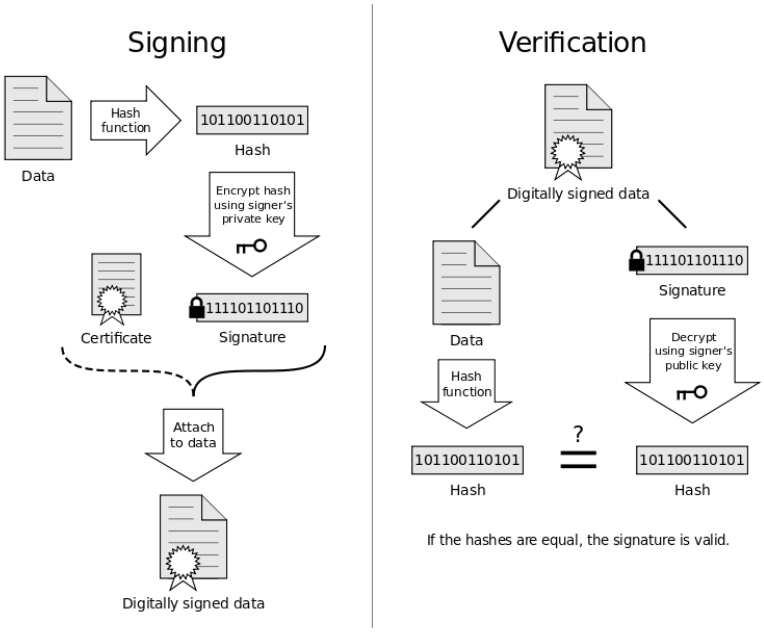

Figure 3: Operating principle of digital signatures[1]

Digital signatures rely on asymmetric cryptography and complex mathematical methods to encrypt digital documents or messages. Asymmetric cryptography comes with a key pair, including a private key (i.e. the one that only the owner knows) and a public key (i.e. the one that can be spread across multiple parties). As shown in Figure 3, the signer (i.e. the sender) applies a hash function on a given piece of data. A hash function is used to apply a mathematical algorithm that maps data of an arbitrary size to a fixed-size bit string (often known as hash or hash value) [99]. These hashing algorithms operate as one-way functions and thus are nearly infeasible to invert. The signing party then encrypts the given hash value with the private key. The result is the cryptographically signed data, which is validated by the sender and in some cases from a certificate-issuing authority. The other party (i.e. the recipient) would then continue the process by performing two steps. First, creating a hash value out of the data as well. And second, using the signer's public key to decrypt and extract the signer's encrypted hash value. In case the two hashes match, the signature can be seen as valid. While this technique is often implemented to digitally sign data (e.g. documents or messages) and hence ensure for data integrity and modification detection, digital signatures can also be used to authenti-

[1] https://en.wikipedia.org/wiki/Electronic_signature

cate the data source and thus, increase end-to-end confidentiality and trust.

Digital certificates or often known as public key certificates, are electronic documents that are designed to verify the possession of a public key. Besides information on the public key, certificates also contain statements on the owners' identity (i.e. name etc.) and the digital signature which proves the validity of the certificate. In a common PKI scheme, certificates are signed by a Certificate Authority (CA), such as IdenTrust[21], Comodo[22] or Let's Encrypt[23]. Before multiple browsers such as Firefox and Google Chrome removed the key generator-element, these certificates could have been generated directly by the client browser without relying on a trusted third party (i.e. the CA). The <keygen> HTML tag was supposed to generate a cryptographic key pair from the client browser, but is seen as deprecated by now which also reflects in the lack of browser support. The standardized format for public key certificates is X.509. Various protocols, including the Transport Layer Security (TLS)[24] protocol, are using X.509 certificates. TLS is the successor of the Secure Sockets Layer (SSL) protocol and provides secured client-server communications by using cryptographic algorithms to encrypt data and public-key cryptography to authenticate the involved parties (e.g. RSA, Diffie-Hellmann or elliptic-curve key exchange). The CA signs the certificate by encrypting it with the owners' private key. The owners' identity can now be seen as verified and used for authentication purposes.

WebID certificates are used to identify and authenticate the end-user in a decentralized manner. The public key is stored in the WebID Certificate, while the private key is stored in a secure key store. The key store can either be located on the client (i.e. protected by a password), in an users external device (i.e. offline hardware such as Nitrokey[25]) or on a separated process running on the operating system[26]. WebID certificates contain a field called `SubjectAlternativeName`, which holds the user's WebID URI to dereference the WebID profile document[27]. A WebID verifier or verification agent possesses a list of WebIDs and cor-

[21] https://www.identrust.com/
[22] https://www.comodo.com/
[23] https://letsencrypt.org/
[24] https://tools.ietf.org/html/rfc8446
[25] https://www.nitrokey.com
[26] https://dvcs.w3.org/hg/WebID/raw-file/tip/spec/tls-respec.html
[27] https://www.w3.org/2005/Incubator/webid/spec/tls/

responding public keys attached to them. Once a client makes a request, the verification agent has to prove that the user is indeed in control of the private key that belongs to the public key. In case the keys match, access control rules according to the requested resources' ACL can be applied. In OpenID Connect 1.0, cryptographic security in the verification process is applied by the use of JSON Web Algorithms (JWA). Based on JavaScript Object Notation (JSON) data structures, the signing and encryption of content is achieved by using both, JSON Web Signature (JWS) and JSON Web Encryption (JWE)[105].

5 Decentralization in Solid, Digi.me and Mastodon

Over the last few years several initiatives have been consolidated to provide users with a more privacy-aware way to interact in a social web environment. This chapter presents three of these approaches, namely Solid[28], Digi.me[29] and Mastodon[30]. We provide a high level abstraction of the core architectures and further outline their approaches in regards to authentication, access control and encryption.

5.1 Solid

Backed by the desire to redirect the development of the World Wide Web, Sir Tim Berners-Lee and some of his MIT cohorts founded the Solid (Social Linked Data) project. Started as an academic, community-driven effort, Solid added commercial support in means of Inrupt[31] in 2018. Solid focuses on decoupling personal data from applications accessing and using the data in order to provide their services. Competition on the application and the storage level is thus based on the quality of the service. This makes it easier for innovative competitors to provide their services without the need for tremendous amounts of user data [129]. Based on the founding web technology standards such as HTML and HTTP, as well as on a set of Semantic Web standards such as RDF and SPAQRL, Solid offers users the ability to port and plug in personal data to a multitude of interoperable services [8]. To enable decentralized social web applications acting in a secure while also

[28] https://solid.inrupt.com/

[29] https://digi.me/

[30] https://joinmastodon.org/

[31] https://inrupt.com/

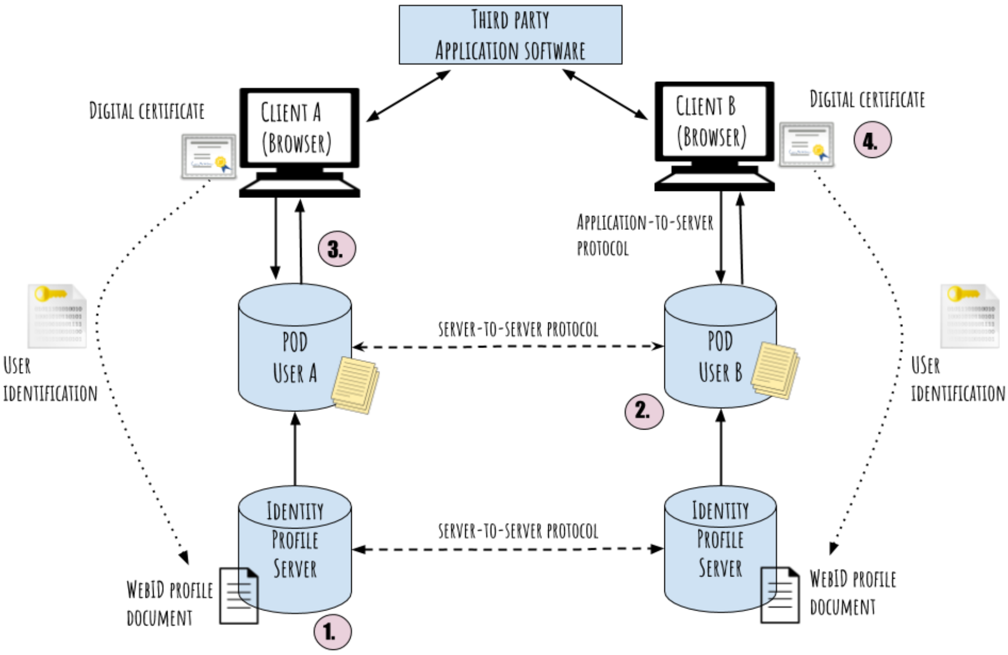

Figure 4: Solid architecture

privacy-aware way, Solid builds upon the following main pillars. Each of the pillars is shown in *Figure 4* and discussed in more detail below[32].

1 A global identity management system based on the WebID protocol to enable users a more transparent way of managing their digital identity while allowing applications to easily authenticate users.

2 A decentralized data management architecture based on the LDP protocol to avoid centralized data aggregation, lower insecurity about improper data access and utilization while providing efficient resource manipulation and retrieval.

3 Application-agnostic Personal Online Data Store (POD) servers utilizing standardized WAC policies for cross-domain authorization for resources within a given POD.

4 Digital certificates facilitating secure user authentication by using public key cryptography.

[32] The numbers correspond to those in Figure 4

5.1.1 Global identity management architecture

Solid pursues to conceive a global ID space with more user control on hosting and managing their credentials, thus facilitating global single sign-on mechanisms and innovating interoperability between various applications. This is a critical part due to the fact that Facebook, Amazon and Google implemented identity protocols such as OpenID Connect [105] and OAuth 2.0 [56]. By doing this they can spare users from handling various credentials (i.e. username and password) and thus centralize identity management and have yet another lock-in mechanism [129]. As in the earlier days the OpenID protocol was restricted to authentication, Facebook created their own protocol namely Facebook Connect, offering affiliated third party applications authorization features such as access to profile information or friend lists instead of just user credentials [86].

While such single sign-on solutions surely make user lifes easier, it leaves open how secure and transparent these data flows are at the end of the day. In 2011, Miculan and Urban [88] outlined how vulnerable the Facebook Connect protocol was against replay and masquerade attacks. OAuth 2.0, which is now used by both OpenID Connect and Facebook Connect, came with various improvements such as TLS-protected communication channels or short-term tokens for access control [22]. However, it was the incapability to protect data flows across affiliated sites down to the smallest detail that resulted in various data scandals such as Cambridge Analytica [133] or the Gmail plug-ins scandal [18].

As OAuth and OpenID do not fully support RDF-based profile information and user attribute extension on RDF-based profiles [107], Solid relies on the decentralized authentication protocol WebID. In the Solid ecosystem, WebID profiles are stored on identity profile servers (see also *Figure 4*), which can be run and maintained on the users own machines[33] or on public servers offered by providers such as Inrupt[34] or Solid community[35]. The information included in WebID profile documents can help to link user profiles and to establish a Web of trust, used by applications to set authorization statements which are enforced in a decentralized manner (see also *Section 5.1.3*). As a result, the global identity management architecture provided by Solid can by nature simplify the way users manage their digital identities by bypassing the need for multiple security credentials, while enhancing end-to-end confidentiality through a decentralized identification process.

[33] https://solid.inrupt.com/docs/installing-running-nss

[34] https://inrupt.net/

[35] https://solid.community/

5.1.2 Decentralized data management

Several proposals have been made for decentralizing existing storage systems, such as Freenet [32], OceanStore [76] or FOAF-based architectures [19]. In those content is stored in nodes run by selected friends [79, 122]. More recent attempts such as Storj [138], Sia [132], IPFS [11] or the protocol proposed by Zyskind et al. [143] utilize blockchain technology to store personal data in a decentralized manner. While these approaches are also conceivable, Solid uses PODs for data storage and management. Similar to identity profile servers, POD servers can be run and maintained on users own machines or on public servers offered by POD providers such as Inrupt or Solid community.

Solid PODs rely on LDP specifications, which define RESTfully HTTP operations, thus facilitate the reading and writing of linked data resources. Both, RDF-based structured data (JSON, Turtle) and non-structured data (images, binary and text files) are stored as a LDPR and grouped together to LDPC. The nested structure of LDPCs can be compared to a file system hierarchy [107]. In fact, several POD implementations by Solid such as **gold**[36], **ldnode**[37] and **ldphp**[38] use file systems to store structured as well as unstructured resources as files. The **meccano**[39] server, in contrast, uses graph database systems based on the Jena framework[40] to store structured data (i.e. RDF data) and file systems to store unstructured data. To identify and detect them, both LDPRs and LDPCs are provided with HTTP(S) URIs and include ACL resources, supporting fine-grained access control. Applications rely on WebID not only to verify the identity of users, but also to discover those links (i.e. HTTP URIs) to profile data.

While POD providers act in a similar way to well-known cloud storage services, they differ in terms of their reliability, privacy or even legal protection [107]. However, the user benefits from an highly interoperable ecosystem, as new applications can easily access existing resources from the user's POD. This higher degree of control in the hands of the user is not only desirable as it avoids centralized data aggregation and as a possible consequence, misuse. Being in full control over one's personal data means controlling who can see and access one's information on the one hand, and more importantly on the other hand, how and for what purpose someone uses it. Fine-grained access control when combined with flexible reverse licensing models could thus

[36] https://github.com/linkeddata/gold

[37] https://github.com/linkeddata/ldnode

[38] https://github.com/linkeddata/ldphp

[39] https://meccano.io

[40] http://jena.apache.org

empower and maybe even monetize intellectual properties [37].

5.1.3 Web Access Control for cross-domain authorization

Solid implemented the WAC ontology to enable decentralized authorization mechanisms. By combining WAC with WebID, Solid applications can apply access control techniques in a distributed manner by granting access to other user's PODs (i.e. the resources stored inside the POD) only to users that feature specific properties inside their own WebID profile. For instance, the **contacts** application[41] allows a user to store and manage a list of contacts (which are based on the vCard ontology[42]) on their own POD. As each of these contacts contains their own WebID, access restrictions can be set based on the social graph (i.e. contacts, contacts of contacts, and so on) of the user. Because of the interoperability Solid provides, access control based on the user's social graph can be applied to various other Solid applications [84].

Similar to traditional file system hierarchies, WAC also provides ACL inheritance algorithms. These ensure that LDPRs (without an ACL attached to it) adopt the ACL from the LDPC in which they are located. Given that the LDPC does not contain an ACL either, this step is repeated until the root container, which must contain an ACL, comes into play. However, what differs WAC from an access restrictions set in usual file systems is its decentralized and cross-domain character, allowing for access to a specific document on one web service, even it is hosted on a different one[43]. If we observe the example of an user who creates an appointment on the **calendar** application[44] and stores it as a resource on his POD with an ACL attached to it, they can load this calendar resource into a different application (e.g. the microblogging app **cimba**[45]) and rely on the same authorization statements as applied on the calendar app.

5.1.4 WebID certificates for user authentication

For Solid applications to securely authenticate users and make sure that one is really who he claims to be, the Solid ecosystem uses authentication mechanisms based on the WebID-TLS protocol. At the point of writing this,

[41] http://linkeddata.github.io/contacts/

[42] https://www.w3.org/TR/vcard-rdf/

[43] https://github.com/solid/web-access-control-spec

[44] http://mzereba.github.io/calendar/

[45] http://cimba.co

the Solid team is implementing support for the WebID-OIDC protocol[46]. Other authentication mechanisms such as WebID-TLS Delegation are under investigation. However, WebID-TLS and WebID-OIDC are currently used as the major primary authentication mechanisms for Solid.

In this context, the HTTP(S) URI of the WebID profile document can be seen as an individuals' username. Subsequently, to substitute the password, WebID certificates come into play. They ensure that the user really is the person he claims to be by sending the username. In fact and befitting Solid's decentralized character, authentication is mainly executed between the POD (or the user's WebID profile respectively) and the client (i.e. the browser). Solid applications only search the WebID certificate to obtain the WebID of the given user. For the user the authentication process requires only one step: choosing the WebID certificate. In order to authenticate to an application, one does not have to remember and disclose security credentials (i.e. password) or type in their WebID. The W3C specifications on WebID-TLS further state: "*If the authenticity of the server hosting the WebID profile document is proven through the use of HTTPS, then the trust one can have in the agent at the end of the TLS connection being the referent of the WebID is related to the trust one has in the cryptography, and the likelihood that the private key could have been stolen.*"

5.2 Digi.me

While Solid started as an academic, community-driven effort and added commercial support in means of Inrupt in a later stage, Digi.me was founded as a commercial effort from the very beginning. After a good funding series in 2016 and the merger with the personal data service Personal in 2017, Digi.me consolidated their focus in rethinking the way individuals and businesses exchange data by creating mutual value in a more privacy-driven manner. Similar to Solid, they aim to aggregate personal information in a data space fully controlled by the user. Thus, providing individuals with the possibility to share data only with service providers who they trust.

In fact, the certificate system used by Digi.me is designed in a way such that the user triggers data exchange by pushing data to the application, instead of the other way around. Digi.me provides the reversal in data exchange triggering by implementing a set of proxy services and cloud components. These services allow users to control how and where their personal data is shared by granting and denying access to third party applications requesting

[46] https://github.com/solid/webid-oidc-spec

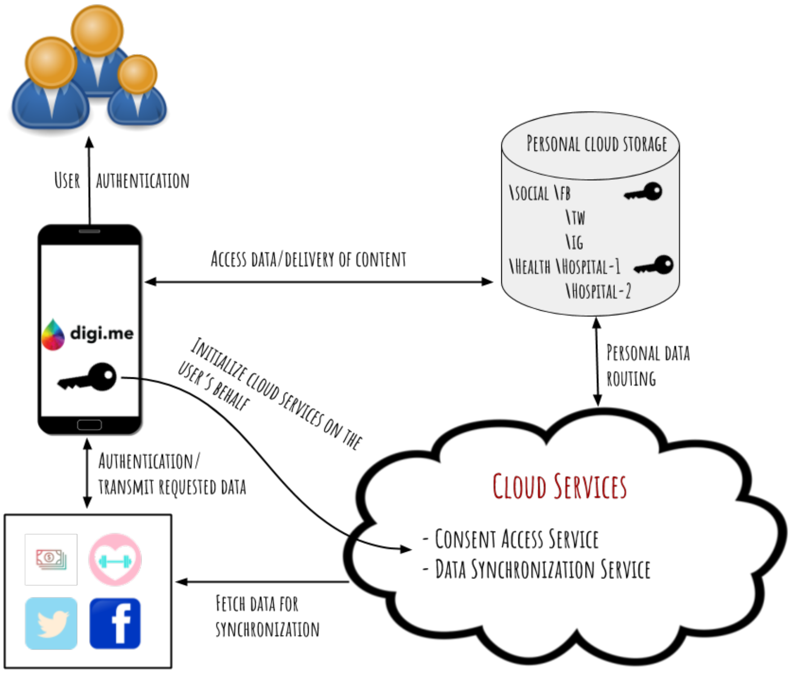
User
Authentication
Personal cloud storage
\Social \FB
\TW
\IG
\Health \Hospital-1
\Hospital-2
Access data/delivery of content
digi.me
Initialize cloud services on the user's behalf
Personal data routing
Authentication/ transmit requested data
Cloud Services
- Consent Access Service
- Data Synchronization Service
Fetch data for synchronization

Figure 5: Core components of Digi.me

their data. This in turn facilitates the application of flexible reverse license agreements, which would benefit not only consumers. Companies would also gain access to high-quality data, enabling them to provide better services. Moreover, they would replace current methods for data acquisition which are often based on opacity through more transparent and traceable methods [81].

5.2.1 Core components

As *Figure 5* shows, the Digi.me ecosystem is based on the combination of personal cloud storages, proxy cloud services and the Digi.me app acting in a coordination role on behalf of the user. Cloud storage services (e.g. Dropbox, Google Drive) allow users to aggregate data from various sources (e.g. social media, finance, health) and store them in a cryptographically secured way. Cloud services can be seen as personal data routers, facilitating data exchange between third party apps and users. They rely on single-purpose processing blocks so that processing threads are destroyed after completion. Because data processing can be performed locally on the user's device via their own temporary virtual machine instance sitting on Microsoft's Azure cloud, one does not have to share personal information with third party applications [97].

For instance, the Finsights application[47] allows for personal spending analysis without storing user's financial data on the servers hosted by the application provider. Instead, the user aggregates financial data from various sources (banking platforms and financial institutions) inside their personal cloud storage and is then able to specify what resources Finsights is allowed to access. The Finsights application performs financial analysis over proxy cloud services without the need to store and maintain data on their own. More control over personal data and improved transparency on how and from whom private information is used might increase the willingness of users to share bigger amounts of diverse data such as health or financial data [97].

Authentication. Digi.me does not rely on standards such as WebID and WAC to conceive a global identity system with decentralized authentication and authorization mechanisms. Instead, user authentication is handled by the Digi.me app over traditional credentials (i.e. username and password). To identify and authorize third party applications interacting with the user's cloud storage, Digi.me uses authentication components from the OAuth 2.0 protocol.

Encryption. Besides authenticating the user, the password also unlocks a secret master key, which acts as a gateway to all encryption keys stored in an internal vault on the user's device. Each of these keys correspond to a single encrypted file within the user's personal cloud storage. Password vaults are protected by asymmetric encryption mechanisms based on the RSA cryptosystem including keys of 2048-bit length and Optimal Asymmetric Encryption Padding (OAEP)[48]. Data inside the personal cloud storage is encrypted using symmetric cryptography, namely AES-256 or AES-128. Even if an attacker should succeed in unlocking an encryption key, which would take months or years depending on the chosen password and the cryptographic algorithm used to generate the key (i.e. the key length), one would only have access to a single file protected by the given key.

5.2.2 Consent Access Service

The following two sections provide an overview of how the core components of the Digi.me ecosystem interact to enable a secure and privacy-aware data exchange between users and third party services. The overall architecture of Digi.me's consent access service (shown in *Figure 6*) is based on vari-

[47] https://digi.me/finsights/

[48] https://medium.com/blue-space/improving-the-security-of-rsa-with-oaep-e854a5084918

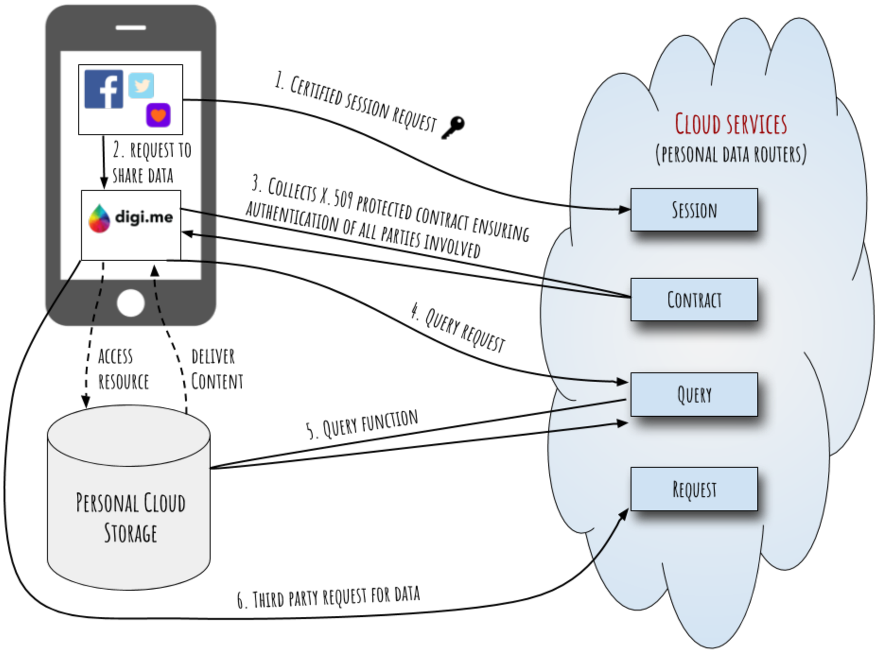

Figure 6: Consent Access Service on Digi.me

ous software processes, which use cryptographic keys to provide end-to-end confidentiality and are protected by a set of security protocols. Multiple protection layers of firewalls, API filters and rate limiters are implemented to prevent different classes of vulnerabilities such as brute-force, replay or zero-day attacks. The following steps describe the underlying consent access service, which provides users with full control over how and where their personal data is processed and stored[49].

1 Whenever a user accesses an external third party service over the Digi.me app, the external application starts the consent access process by sending a certified session request via an API call. This request is authenticated by TLS certificates and OAuth 2.0 components, which ensure that the given application is a trustworthy part of the Digi.me ecosystem.

2 Once the app is authenticated and the session is identified, the third party application sends a request to the Digi.me app, which asks the user for the approval to share their data with the third party app.

[49] The numbers correspond to those in Figure 6

3 Accessing an API exposed by the cloud service provider, the Digi.me app collects a machine readable contract. This contract is protected within a X.509 certificate or a JWS wrapper and ensures that all involved parties have been authenticated and the contract has not been tampered with. The contract is stored by a cloud service and specifies for every single data exchange, what data is sent, what this data is used for, for how long the third party application will retain the data and whether they will share it with other parties[50].

4 After the user has accepted the terms obtained in the contract, the Digi.me app asks them for their approval to fetch the required data for them and sends a query request to the cloud service component. On top of that, a log message containing information about the user's approval (i.e. time of approval, etc.) is generated, serving as a mutual trust mechanisms between third party services, Digi.me and users.

5 The crucial part from a security point of view is the query function triggered by the cloud service. As the query function is only fetching encrypted data from the secured cloud storage, Digi.me can neither see or use personal data. In addition, the cloud service, which stores data temporarily in encrypted form, re-encrypts the headers of the data containing the encryption keys, so that only third party applications can access it. Thus, no matter what happens to the data stored for a short period of time in the cloud service environment, no party except the application that required it can decrypt and read it.

6 Finally, the third party application can fetch the requested data via an API call. Given that the master key is unlocked by the user's password, Digi.me can access the file keys and present the data to the application in an unlocked format.

5.2.3 Synchronization service and data storage

In order to store user data generated from external sources (e.g. Facebook, health services, etc.), Digi.me relies on a cloud based synchronization service to fetch and move data into the personal cloud storage [47]. Even though cloud storage services are run by centralized parties, the trust one can have in cryptographic cloud storages nearly equates to the trust one can have in the reliability of cryptography [66]. Thus, data integrity can be taken for granted. Furthermore, the fact that data is only passed from one place to

[50] https://developers.digi.me/sample-sharing-contracts.html

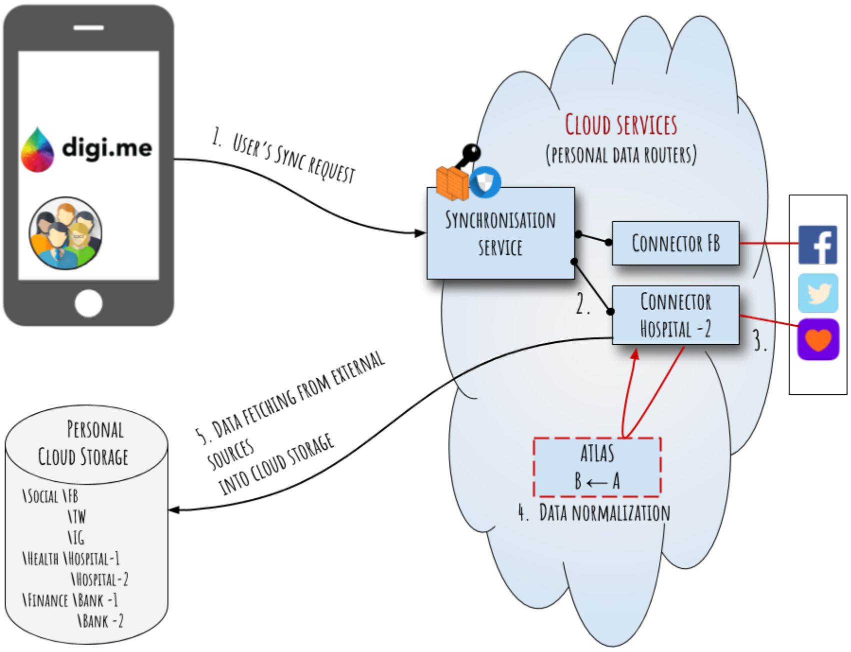

Figure 7: Data synchronization on Digi.me

another in cipher instead of unencrypted plain-text provides utmost confidentiality and transparency in terms of who and how someone can access personal information. Besides the fact that users can enjoy end-to-end confidentiality due to cryptographic mechanisms, cloud storage services are also highly scalabe, easy to deploy and configure, and reliable [139]. The following steps describe the data synchronization service as featured in *Figure 7*. The underlying mechanisms ensure not only modification detection but also simplify the use of third party applications, which operate with personal data in various data formats[51].

1 Whenever a user wants to store application data from an external source, a synchronization request is triggered via the Digi.me app. The synchronization request function is highly protected and comes with zero-day attack mitigation, Distributed Denial of Service (DDoS) prevention and rate limiting. These protection mechanisms guarantee that only the authenticated and authorized user is able to request data synchronization.

2 Once the request function was proven valid, the synchronization service

[51] The numbers correspond to those in Figure 7

establishes a connector function to an external application's API. SSL connection requirements and Swagger definitions[52] ensure that all APIs that somehow are publicly exposed via internet access act in a secure way. Whenever a new external data source is added, a new connector function can be written efficiently based on a common code foundation.

3 The external application receives the synchronization request and sends the required data back to the connector.

4 With multiple data sources, the variety of occurring data formats increases. In order to simplify the use of personal data for other third party applications which may use user data generated by other sources, data should be normalized into a common global format before it is stored in users personal cloud storage. This is done by an internal cloud service named ATLAS. Providing a flexible and powerful object transformation service, ATLAS converts data that comes in the specific format used from external sources into a standardized and consistent format irrespectively from its source.

5 Once data formalization is achieved, the data can be fetched from the connector and can be moved to the cloud storage where it is stored in encrypted files. Thus, Digi.me and all involved cloud storage services only have access to file systems full of encrypted data, instead of any database on their servers allowing them for disclosure or other types of misuse of sensitive user information.

5.3 Mastodon

Mastodon is a community-driven, open-source software platform for hosting social media sites. As a federated network, Mastodon does not run a social media site by themselves, but instead offers a way for everyone to host their own microblogging server (called instance) on the domain of their choice. As shown in *Figure 8*, each instance is hosted on different servers by an independent group of admins or developers. Since the launch in 2016, Mastodon aggregated more than 2700 instances (e.g. with *mastodon.social*[53] being the largest one) and almost two million active accounts. Unlike in the cases of Solid and Digi.me, the technical documentation regarding Mastodon and especially their approach to authentication, access control, and encryption

[52] https://docs.swagger.io/spec.html

[53] https://mastodon.social/about

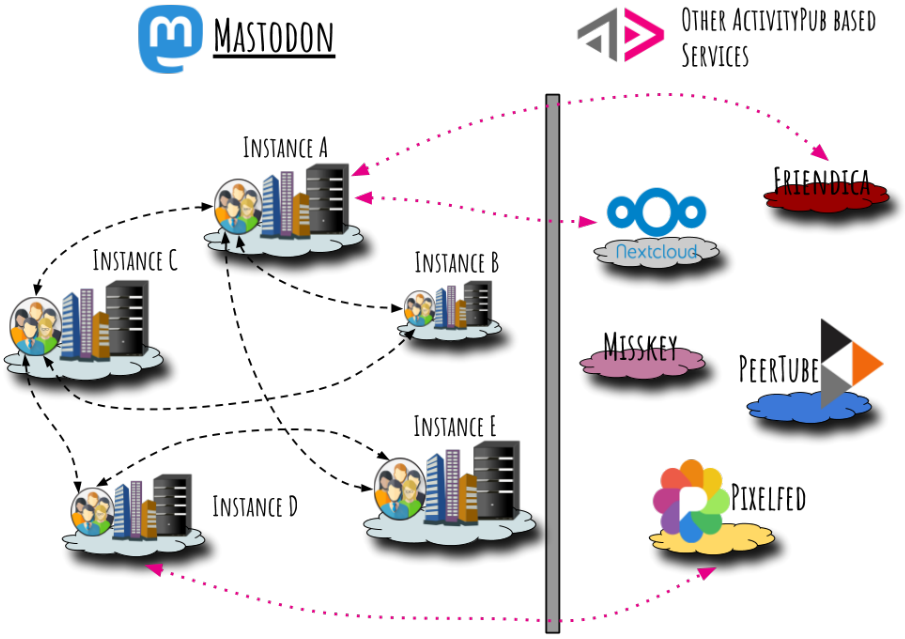

Figure 8: Mastodon Fediverse based on ActivityPub

mechanisms is limited. As a consequence of the lack of information and academic papers, the general informative value of the following sections is somewhat restricted.

5.3.1 ActivityPub

The Mastodon platform is built upon the ActivityPub protocol[54], providing interoperability between various servers and allowing users on one server to seamlessly talk to users from a different server. ActivityPub in the Mastodon ecosystem acts similar to the Simple Mail Transfer Protocol (SMTP) in the E-Mail ecosystem. While SMTP allows different mail providers (such as Gmail or GMX) to interact, the ActivityPub protocol enables instances to communicate with each other while still being hosted and maintained independently. Thus, users on one instance can follow users from another one, building a social graph over multiple independent networks. This social graph is not restricted to the Mastodon ecosystem. In fact, every service or platform that implements the ActivityPub protocol can interoperate with Mastodon instances, allowing users from Mastodon to interact with people from e.g.

[54] https://www.w3.org/TR/activitypub/

Misskey[55], Pixelfed[56] or PeerTube[57]. In additon, the Activity Streams 2.0 specification[58] provides a JSON-based serialization syntax to describe actions (such as updating a blog post or following an user) in a semantic and machine-processable manner.

5.3.2 Authentication

When no central authority is given in a network, the verification of the user's identity is not straightforward and thus, impersonation attacks are a major challenge. Mastodon uses a validation method based on the "rel=me" hyperlink extension. The idea is to add one or more links (e.g. personal homepage, twitter account etc.) to your Mastodon profile metadata and prove your identity by cross-referencing these links[59]. Verifying my identity by proving the ownership of my personal homepage would thus require two steps: (i) adding *<a href="https://mastodon. social/@stefan" rel="me">@mastodon</a>* to my personal homepage; and (ii) adding *<a href="https://myhomepage.example" rel="me">My personal homepage</a>* to my Mastodon profile metadata.

This approach, however, requires trust and the reliability surely depends on the recognition factor of the linked account or website[60]. With version 2.8.0 released in the mid of 2019, Mastodon implemented Keybase[61] identity proofs, which provide a way to link user's identities from multiple services with their encryption keys. The ownership of Mastodon accounts can be validated by posting cryptographically signed statements that link to the user's keybase account[62].

The ActivityPub's core specification does not include an official mechanism to authentication. However, there are some common practices the developer community has converged on[63]. To enable users to interact on a multitude of Mastodon instances (client to server authentication), ActivityPub implemented the OAuth 2.0 protocol[64]. In order to authenticate servers

[55] https://misskey.io/
[56] https://pixelfed.org/
[57] https://joinpeertube.org/
[58] https://www.w3.org/TR/activitystreams-core/
[59] https://docs.joinmastodon.org/usage/decentralization/
[60] https://blog.joinmastodon.org/2018/10/mastodon-2.6-released/
[61] https://keybase.io/
[62] https://keybase.io/blog/keybase-proofs-for-mastodon-and-everyone
[63] https://www.w3.org/wiki/SocialCG/ActivityPub/Authentication_Authorization
[64] https://docs.joinmastodon.org/api/authentication/

to each other in a federated environment, they implemented: (i) HTTP Signatures [27] in conjunction with the actor's public key; and (ii) Linked Data Signatures[65]. However, considering the technical documentation, it is not clear how Mastodon handles server to server authentication at this point.

5.3.3 Access Control

In the federated network of Mastodon, access control should primarily specify: (i) which actors (i.e. users hosted on a given instance) can read an user's post (called toot); (ii) whether or not one can respond to a toot; and (iii) how actors can be described by properties in order to set fine-grained permissions. Access control and permissions on a user level are applied locally (i.e. from the specific instance). Therefore, users on one particular instance cannot be restricted by server admins or moderators from a different instance. Users in all instances can set privacy preferences for their posts so that only their followers or the tagged users can see it. The inbox stream includes all activities received by a given user (see also[66]). The instance server is then supposed to filter these activities depending on the requester's permission.

However, one may argue that this approach seems limited as it does not allow for granting access to specific instances or groups of users which are aligned with one's moderation policies or desired properties. The ability to create addressable meta-collections that contain specific properties an instance or user must provide, and allow for object fetching based on these collections, is currently missing. Several suggestions (see for instance [67], [68] or [69]) have been made to address the authorization issues which ActivityPub-based platforms face by considering a capability-based authorization mechanism for Linked Data (such as OCAP-LD).

5.3.4 Encryption

At the point of writing, Mastodon does not provide end-to-end encryption for user interaction. Besides the addressed recipient, the instances (i.e. admins of the sender's and recipient's server) can see, access and store private

[65] https://w3c-dvcg.github.io/ld-signatures/

[66] https://www.w3.org/TR/activitypub/

[67] https://wordsmith.social/falkreon/securing-activitypub

[68] https://socialhub.network/t/sending-ocap-ld-invocation-in-activitypub/424

[69] https://blog.dereferenced.org/what-would-activitypub-look-like-with-capability-based-security-anyway

	Solid	Digi.me	Mastodon
Identification and authentication	WebID for decentralized user identification; WebID-TLS for decentralized authentication	Username and password for users; OAuth 2.0 for third party applications	Username and password for users; OAuth 2.0 for third party applications
Access Control	WAC for decentralized, cross-domain user authorization	Cloud-based consent access service for app authorization	Applied locally on instance level
Data storage	PODs hosted by POD providers (with the option to self-host server and run POD on your own)	Personal cloud storage hosted by centralized providers	Centralized data storage on separated instance servers
Data encryption	Not implemented at the moment	AES-256 for file encryption within cloud storage	Not implemented at the moment
Confidentiality	Digital signatures; X.509 certificates; TLS	TLS; RSA encryption for password vaults with additional OAEP padding; X.509 certificates	Keybase proofs for identity verification; TLS connection in most instances (not sure if required)
Interoperability	WebID for Auth interop; REST API + Linked Data principles for API and schema interop	Data normalization via internal cloud service ATLAS for schema interop	ActivityPub + Activity Streams 2.0 for API and schema interop

Table 6: Comparison of web platforms based on key indicators

messages as well[70]. This also applies if the user sets the privacy preference for their post so that only followers or tagged users are addressed. While the encrypted transport layer (using TLS) in most of the Mastodon servers protects against vulnerabilities such as man-in-the-middle attacks, the lack of end-to-end encryption in message exchange amplifies the trust, users must have in the chosen instance[71].

5.4 Discussion

Based on the outcomes of the previous chapter, *Table 6* presents a comparative analysis of Solid, Digi.me and Mastodon in regards to several key indicators. Each of the discussed platforms follows a different strategy to enforce a decentralized, privacy-preserving social web. This chapter provides an evaluation of the status quo and further outlines some of the most challenging, non-technical barriers, which should serve as an impulse for further research.

[70] https://github.com/tootsuite/mastodon/issues/9004

[71] https://2ality.com/2017/04/mastodon.html

5.4.1 Solid

Solid has a strong focus on providing interoperability between apps and thus avert enforced data lock-in and account proliferation. While blockchain-based alternatives such as Substratum[72] may focus on architectural decentralization in a puristic way, Solid aims to solve the core challenge of centralized data aggregation. Even though there are open questions on encryption and end-to-end confidentiality when it comes to POD and identity providers, Solid offers users the choice to choose and switch between various providers and still retain access to their data and their social graphs. A choice that users often do not have in todays fragmented web. When consumers have full control over their personal information, they can independently decide where they want to store the data, which provider they trust to store it, and further who can access the data. Thus, the better understanding of who gets access to what information can potentially result in greater transparency as it clarifies the way, decisions are made about the user.

5.4.2 Digi.me

Digi.me is on a very similar path aiming for data freedom in a decentralized web, but follows a slightly different strategy. While Solid fosters interoperability by building on open standards and Linked Data principles, Digi.me acts as a transparent data broker between third party applications and consumers. Fine-grained encryption mechanisms on file level increase user privacy by minimizing the risk for disclosure or misuse of sensitive information down to a minimum. Moreover, the vocabulary interoperability based on data normalization capabilities facilitates the processing of various data formats from multiple sources. Such an automated schema migration and conversion could also enhance the interoperability within the Solid ecosystem by facilitating the transformation of various ontologies[73].

5.4.3 Mastodon

Mastodon as a federated network deals with the 'winner takes all' capitalism model formed by closed silos. By offering everyone the ability to host their own microblogging instance, Mastodon empowers competition between providers as well as providing choice for consumers. Besides the lack of end-

[72] www.substratum.net

[73] https://forum.solidproject.org/t/digi-me-on-the-same-path-to-data-freedom/1185/2

to-end confidentiality in private message exchange, one of the most significant issues of Mastodon was the synergy between the lack of distributed identities and the moderation policies of single instances. As a consequence, if the user decided to leave a given instance for whatever reason, or the instance got shut down from the admins, the user would have lost their complete social graph built on that specific instance. With version 3.0[74] Mastodon implemented an account migration system that provides the ability to transfer followers from an account hosted on one instance to an account hosted on a different instance.

Raman et al. [100] recently discovered apparent forms of centralization within the Mastodon ecosystem. According to their results, almost half of the Mastodon users are hosted on 10% of the instances, which may constitutes the risk for converging towards a semi-centralized-like network. Regular instance outages and the fact that almost half of the toots are published on only ten instances furthermore emphasize the importance of replication strategies (such as copying posts onto a secondary instance) to improve the availability.

If we think back to the use case scenario in *Section 4.1*, each of these platforms tackles the issue of centralized services from a slightly different angle. All of them aim to replace the current data lock-in by interoperable services, however, only Solid and Digi.me enable independence from service providers by decoupling data from applications. As for Mastodon, it is not fully clear how user privacy within the federated ecosystem is promoted, leaving open the questions of content moderation, decentralized identity and end-to-end confidentiality. While the previous discussion points are platform-specific, several common challenges exist that all decentralized social web platforms currently face:

5.4.4 User and developer adoption

One of the most substantial barriers is at the same time probably the most obvious one - the classical 'chicken and egg' problem when it comes to adoption. Developers are not motivated enough to build applications if there is no wide user base, and unless there are developers pushing forward applications, there will be not enough competition to drive attractive alternatives which lead users to switch from their current data silos. Especially considering Solid there is still no such breakthrough application that could potentially drive users to enter into the Solid ecosystem. The lack of adoption of existing, decentralized, privacy-preserving Facebook alternatives may indicate

[74] https://blog.joinmastodon.org/2019/10/mastodon-3.0/

that privacy as the only feature is not enough. On the one hand not for the average user to overcome inertia and compensate trade-offs regarding usability and functionality. And on the other hand not for SME and providers who are currently locked out of the market to be empowered enough to compete with market leaders. The success of new technologies often depends on the new abilities provided to the consumer. Therefore, these platforms must offer user experience that is able to compete with the current status quo in regards to usability and speed, and functionalities that existing social web applications cannot deliver due to their siloed nature [129].

5.4.5 Open standards

Another issue that comes along with adoption is the importance of open standards and protocols. Especially more complex standards regarding the Semantic Web often face a long and difficult road until they reach widespread adoption [8]. The success of the responsible parties such as the W3C to pass open standards for the decentralized web will determine whether or not developers will use these standards to build applications. In the case of Solid, engagement of the developer community is facilitated by providing `solid.js`[75]. `Solid.js` comes as a browser library for client-side scripting and includes all of the relevant Solid protocols to support developers in writing Solid applications [107]. On top of that, library extensions such as `solid-auth-client`[76] accelerate the development lifecycle of Solid applications by allowing for a smooth and secure way to log in to an user's data pod and to perform efficient reading and writing of the data.

5.4.6 Monetization and decentralized business models

If the technical implementation is the groundwork, then the economics, marketing, user centered design and coordination can be the driving forces for the successful adoption of decentralized web platforms. It will need a fundamental shift away from current digital business models which are based on data lock-in and the monetization of user data as they will not work the same way they do for centralized, corporate-controlled platforms. The investigation of alternative revenue models such as subscriptions, reverse licensing models or micropaymnets based on cryptocurrencies will be an important aspect for the re-decentralization of the social web. Amongst many other efforts, the

[75] https://github.com/solid/solid-client

[76] https://solid.github.io/solid-auth-client/

Web Monetization[77] specification proposed by the Web Incubator Community Group (WICG), an API that allows for micropayments and is intended to be an alternative to advertisements, recently caught attention[78]. Ng [94] examines the changes in business models that result from a connected, digital world and the availability of data as an economic resource. Tumasjan and Beutel [125] discussed Blockchain-based decentralized business models in the sharing economy and further outlined relevant parameters of user adoption of these kind of models.

5.5 Future work

Platforms such as Solid, Digi.me and Mastodon delivered the technological proof that social web applications can be built upon a more decentralized and privacy-preserving approach. Although research on Semantic Web technologies and Linked Data initiatives that underpin the vision of a decentralized web makes constant progress, many challenges seem to be ahead of us. Based on our analysis we have identified a number of possible directions for future research:

Interoperability & schema migration.

In order to improve vocabulary interoperability and to deal with different data models, it will need tools for automated schema migration and conversion. A full data normalization capability as it is provided by Digi.me could also leverage platforms such as Solid or Mastodon in terms of vocabulary interoperability[79]. An interesting direction for future research would thus be to investigate how the mentioned platforms intersect in terms of automated schema migrations and examine how the could leverage each other.

Decentralized identity management systems.

Earlier we highlighted how the platforms of Solid, Digi.Me and Mastodon approach the identification and authentication of their users. We identified that the lack of a truly decentralized identity management system (in the case of Mastodon) can have an adverse effect on the overall performance. Therefore, we believe it would be useful to investigate alternative approaches, such as the DLT-based identity management

[77] https://adrianhopebailie.github.io/web-monetization/

[78] https://techcrunch.com/2019/09/16/100m-grant-for-the-web-fund-aims-to-jump-start-a-new-way-to-pay-online/

[79] https://forum.solidproject.org/t/digi-me-on-the-same-path-to-data-freedom/1185

schemes evaluated by Dunphy & Petitcolas [41] or the WebID protocol implemented in the Solid ecosystem, for decoupling user identities from Mastodon instances.

Economic models & monetization strategies.

A shift away from centralized services, who control and systematically monetize personal user information by placing individualized advertisements, raises also the need for alternative digital business models. The economic models based on which the incumbent firms on the market of social web applications generate their main revenue streams from would not be fully consistent with the vision of a more decentralized and self-determined web. Although we have seen research and standardization efforts (such as the Web Monetization[80]) pushing in this direction, we believe it would be interesting and beneficial to further investigate new revenue models on which decentralized platforms can built their services.

Content curation & community governance.

The disappearance of a single party controlling the digital service also entails new challenges to content curation and community governance especially in regards to decentralized social media sites. There will be a need for legal frameworks for establishing consumer rights over the content and data created on decentralized platforms. Therefore, an interesting direction for future work would be, to examine tools to effectively moderate content and governance of speech in federated networks such as Mastodon.

6 Conclusion

Since its inception in 1989 the internet took a remarkable development. This resulted in several billions of people spread across the globe benefiting from the economic possibilities, simplified collaborations and the tremendous amounts of shared knowledge. As an integral part of this ecosystem, the social web gradually penetrated into our everydays life. But large scale data scandals and privacy breaches changed public awareness on how much of our control over personal information we are willing to hand over to corporate-controlled platforms. Some may argue that we needed this unprecedented

[80] https://adrianhopebailie.github.io/web-monetization/

centralization to realize how we want our systems to be designed. However, to say decentralization will be the solution to every single problem we face would be counterproductive and detrimental. Freedom, competition, autonomy, experimentation, different people associate different values with decentralization. But all of these values are said to enhance innovation. We need to foster this innovation as it will provide more choice, which is ultimately benefiting our society as a whole.

In this thesis, we aimed to identify: (i) the key challenges as well as opportunities with respect to the development of decentralized social web platforms; (ii) techniques that can be used to facilitate authentication, access control and encryption in decentralized platforms; and (iii) currently available platforms that offer users more control with respect to personal data processing as well as their approaches to authentication, access control and encryption. We presented a taxonomy of network architectures and differentiated between centralized, decentralized and distributed systems to exemplify what decentralization in information systems means, offers and takes. We provided an overview of what it means to protect user privacy and furthermore explained the concepts of authentication, access control and encryption, and how these techniques can enhance privacy. Furthermore, we looked at the technical functionalities as well as at some of the key challenges and opportunities of Solid, Digi.me and Mastodon. Finally, we outlined directions for future research in this field. Our intention was clearly not only to deliver answers, but also to raise questions and provide an impulse for further discussions and research regarding the decentralization of the social web.

To conclude in the words of Sir Tim Berners-Lee: *"It has taken all of us to build the web we have, and now it is up to all of us to build the web we want - for everyone."* [13].

References

[1] Jiyad Ahsan. Centralized vs. decentralized: The best (and worst) of both worlds. https://hackernoon.com/centralization-vs-decentralization-the-best-and-worst-of-both-worlds-7bfdd628ad09, 2018. [Online; accessed 2019-07-12].

[2] Luca Maria Aiello, Marco Milanesio, Giancarlo Ruffo, and Rossano Schifanella. Tempering kademlia with a robust identity based system. In *2008 Eighth International Conference on Peer-to-Peer Computing*, pages 30–39. IEEE, 2008.

[3] Luca Maria Aiello and Giancarlo Ruffo. Lotusnet: Tunable privacy for distributed online social network services. *Computer Communications*, 35(1):75–88, 2012.

[4] Hunt Allcott and Matthew Gentzkow. Social media and fake news in the 2016 election. *Journal of economic perspectives*, 31(2):211–36, 2017.

[5] Mohamed Almorsy, John Grundy, and Ingo Müller. An analysis of the cloud computing security problem. *arXiv preprint arXiv:1609.01107*, 2016.

[6] Jamila Alsayed Kassem, Sarwar Sayeed, Hector Marco-Gisbert, Zeeshan Pervez, and Keshav Dahal. Dns-idm: A blockchain identity management system to secure personal data sharing in a network. *Applied Sciences*, 9(15):2953, 2019.

[7] Randy Baden, Adam Bender, Neil Spring, Bobby Bhattacharjee, and Daniel Starin. Persona: an online social network with user-defined privacy. In *ACM SIGCOMM Computer Communication Review*, volume 39, pages 135–146. ACM, 2009.

[8] Chelsea Barabas, Neha Narula, and Ethan Zuckerman. Defending internet freedom through decentralization: Back to the future. *The Center for Civic Media & The Digital Currency Initiative MIT Media Lab*, 2017.

[9] John Perry Barlow. A declaration of the independence of cyberspace. https://projects.eff.org/~barlow/Declaration-Final.html, 1996. [Online; accessed 2019-05-26].

[10] Jamie Bartlett. Soon, the internet will be impossible to control. https://www.telegraph.co.uk/technology/internet/11284538/Soon-the-internet-will-be-impossible-to-control.html, 2014. [Online; accessed 2019-09-23].

[11] Juan Benet. Ipfs-content addressed, versioned, p2p file system. *arXiv preprint arXiv:1407.3561*, 2014.

[12] Tim Berners-Lee. Linked data - design issues. http://www.w3.org/

DesignIssues/LinkedData.html, 2006. [Online; accessed 2019-05-25].

[13] Tim Berners-Lee. Tim berners-lee: I invented the web. here are three things we need to change to save it. https://www.theguardian.com/technology/2017/mar/11/tim-berners-lee-web-inventor-save-internet, 2017. [Online; accessed 2019-05-28].

[14] Rohit Bhadauria and Sugata Sanyal. Survey on security issues in cloud computing and associated mitigation techniques. *arXiv preprint arXiv:1204.0764*, 2012.

[15] Ames Bielenberg, Lara Helm, Anthony Gentilucci, Dan Stefanescu, and Honggang Zhang. The growth of diaspora-a decentralized online social network in the wild. In *2012 Proceedings IEEE INFOCOM Workshops*, pages 13–18. IEEE, 2012.

[16] Christian Bizer, Tom Heath, and Tim Berners-Lee. Linked data: The story so far. In *Semantic services, interoperability and web applications: emerging concepts*, pages 205–227. IGI Global, 2011.

[17] Piero Bonatti, Sabrina Kirrane, Axel Polleres, and Rigo Wenning. Transparent personal data processing: The road ahead. In *International Conference on Computer Safety, Reliability, and Security*, pages 337–349. Springer, 2017.

[18] Russell Brandom. Will third-party plugins survive the tech backlash? https://www.theverge.com/2018/7/6/17538400/gmail-plugin-privacy-app-developers-google-facebook, 2018. [Online; accessed 2019-08-22].

[19] Dan Brickley and Libby Miller. Foaf vocabulary specification 0.99. http://xmlns.com/foaf/spec/, 2014. [Online; accessed 2019-05-25].

[20] Sonja Buchegger and Anwitaman Datta. A case for p2p infrastructure for social networks-opportunities & challenges. In *2009 Sixth International Conference on Wireless On-Demand Network Systems and Services*, pages 161–168. IEEE, 2009.

[21] Sonja Buchegger, Doris Schiöberg, Le-Hung Vu, and Anwitaman Datta. Peerson: P2p social networking: early experiences and insights. In *Proceedings of the Second ACM EuroSys Workshop on Social Network Systems*, pages 46–52. ACM, 2009.

[22] Glauber C Batista, MaurÃcio A Pillon, Guilherme P Koslovski, Charles Miers, Nelson Mimura Gonzalez, and Marcos Simplicio. Using externals idps on openstack: A security analysis of openid connect, facebook connect, and openstack authentication. 05 2018.

[23] Carole Cadwalladr. The great british brexit robbery: how our democracy was hijacked. *The Guardian*, 7, 2017.

[24] Carole Cadwalladr and E Graham-Harrison. The cambridge analytica

files. *The Guardian*, 21:6–7, 2018.

[25] Mariana Carroll, Alta Van Der Merwe, and Paula Kotze. Secure cloud computing: Benefits, risks and controls. In *2011 Information Security for South Africa*, pages 1–9. IEEE, 2011.

[26] Jason Catlett. Panel on infomediaries and negotiated privacy techniques. In *Computers, Freedom and Privacy: Proceedings of the tenth conference on Computers, freedom and privacy: challenging the assumptions*, volume 4, pages 155–156, 2000.

[27] Mark Cavage and Manu Sporny. Signing http messages. 2018.

[28] Antorweep Chakravorty and Chunming Rong. Ushare: user controlled social media based on blockchain. In *Proceedings of the 11th international conference on ubiquitous information management and communication*, page 99. ACM, 2017.

[29] Usman W Chohan. The concept and criticisms of steemit. *Available at SSRN 3129410*, 2018.

[30] Shihabur Rahman Chowdhury, Arup Raton Roy, Maheen Shaikh, and Khuzaima Daudjee. A taxonomy of decentralized online social networks. *Peer-to-Peer Networking and Applications*, 8(3):367–383, 2015.

[31] Konstantinos Christidis and Michael Devetsikiotis. Blockchains and smart contracts for the internet of things. *Ieee Access*, 4:2292–2303, 2016.

[32] Ian Clarke, Oskar Sandberg, Brandon Wiley, and Theodore W Hong. Freenet: A distributed anonymous information storage and retrieval system. In *Designing privacy enhancing technologies*, pages 46–66. Springer, 2001.

[33] Lorrie Faith Cranor. Agents of choice: Tools that facilitate notice and choice about web site data practices. *arXiv preprint cs/0001011*, 2000.

[34] Leucio Antonio Cutillo, Refik Molva, and Thorsten Strufe. Privacy preserving social networking through decentralization. In *2009 Sixth International Conference on Wireless On-Demand Network Systems and Services*, pages 145–152. IEEE, 2009.

[35] Leucio Antonio Cutillo, Refik Molva, and Thorsten Strufe. Safebook: Feasibility of transitive cooperation for privacy on a decentralized social network. In *2009 IEEE International Symposium on a World of Wireless, Mobile and Multimedia Networks & Workshops*, pages 1–6. IEEE, 2009.

[36] Gabriel Dance, Michael LaForgia, and Nicholas Confessore. As facebook raised a privacy wall, it carved an opening for tech giants. https://www.nytimes.com/2018/12/18/technology/facebook-privacy.html, 2018. [Online; accessed 2019-09-19].

[37] Anwitaman Datta, Sonja Buchegger, Le-Hung Vu, Thorsten Strufe,

and Krzysztof Rzadca. Decentralized online social networks. In *Handbook of Social Network Technologies and Applications*, pages 349–378. Springer, 2010.

[38] Primavera De Filippi. The interplay between decentralization and privacy: the case of blockchain technologies. *Journal of Peer Production, Issue*, (7), 2016.

[39] Andrea De Salve, Barbara Guidi, Paolo Mori, Laura Ricci, and Vincenzo Ambriola. Privacy and temporal aware allocation of data in decentralized online social networks. In *International Conference on Green, Pervasive, and Cloud Computing*, pages 237–251. Springer, 2017.

[40] Ben Dodson, Ian Vo, TJ Purtell, Aemon Cannon, and Monica Lam. Musubi: disintermediated interactive social feeds for mobile devices. In *Proceedings of the 21st international conference on World Wide Web*, pages 211–220. ACM, 2012.

[41] Paul Dunphy and Fabien AP Petitcolas. A first look at identity management schemes on the blockchain. *IEEE Security & Privacy*, 16(4):20–29, 2018.

[42] Matthew English, Sören Auer, and John Domingue. Block chain technologies & the semantic web: A framework for symbiotic development. In *Computer Science Conference for University of Bonn Students, J. Lehmann, H. Thakkar, L. Halilaj, and R. Asmat, Eds*, pages 47–61, 2016.

[43] Joerg Evermann and Henry Kim. Workflow management on the blockchain—implications and recommendations. *arXiv preprint arXiv:1904.01004*, 2019.

[44] José G Faísca and José Q Rogado. Decentralized semantic identity. In *Proceedings of the 12th International Conference on Semantic Systems*, pages 177–180. ACM, 2016.

[45] Shehroze Farooqi, Fareed Zaffar, Nektarios Leontiadis, and Zubair Shafiq. Measuring and mitigating oauth access token abuse by collusion networks. In *Proceedings of the 2017 Internet Measurement Conference*, pages 355–368. ACM, 2017.

[46] Luciano García-Bañuelos, Alexander Ponomarev, Marlon Dumas, and Ingo Weber. Optimized execution of business processes on blockchain. In *International Conference on Business Process Management*, pages 130–146. Springer, 2017.

[47] Ray Gavin. Digi.me security intro. `https://digi.me/downloads/download-security-presentation.pdf`, 2019. [Online; accessed 2019-08-24].

[48] Robert Gellman. Privacy in the clouds: Risks to privacy and confi-

dentiality from cloud computing. In *World privacy forum*, volume 23, 2009.

[49] Kalman Graffi, Christian Gross, Dominik Stingl, Daniel Hartung, Aleksandra Kovacevic, and Ralf Steinmetz. Lifesocial. kom: A secure and p2p-based solution for online social networks. In *2011 IEEE Consumer Communications and Networking Conference (CCNC)*, pages 554–558. IEEE, 2011.

[50] Ralph Gross and Alessandro Acquisti. Information revelation and privacy in online social networks. In *Proceedings of the 2005 ACM workshop on Privacy in the electronic society*, pages 71–80. ACM, 2005.

[51] Shivam Gupta, Arpan Kumar Kar, Abdullah Baabdullah, and Wassan AA Al-Khowaiter. Big data with cognitive computing: a review for the future. *International Journal of Information Management*, 42:78–89, 2018.

[52] Stuart Haber and W Scott Stornetta. How to time-stamp a digital document. In *Conference on the Theory and Application of Cryptography*, pages 437–455. Springer, 1990.

[53] Irit Hadar, Tomer Hasson, Oshrat Ayalon, Eran Toch, Michael Birnhack, Sofia Sherman, and Arod Balissa. Privacy by designers: software developers' privacy mindset. *Empirical Software Engineering*, 23(1):259–289, 2018.

[54] John Hagel, John Hagel 3rd, and Marc Singer. *Net worth: shaping markets when customers make the rules*. Harvard Business Press, 1999.

[55] Harry Halpin. Decentralizing the social web. In *INSCI'2018-5th International conference'Internet Science'*, 2018.

[56] Dick Hardt. The oauth 2.0 authorization framework. 2012.

[57] Justus Haucap and Ulrich Heimeshoff. Google, facebook, amazon, ebay: Is the internet driving competition or market monopolization? *International Economics and Economic Policy*, 11(1-2):49–61, 2014.

[58] Tom Heath and Christian Bizer. Linked data: Evolving the web into a global data space. *Synthesis lectures on the semantic web: theory and technology*, 1(1):1–136, 2011.

[59] Julian PT Higgins, Sally Green, et al. Cochrane handbook for systematic reviews of interventions. 2008.

[60] Giles Hogben. Security issues and recommendations for online social networks. *ENISA position paper*, 1:1–36, 2007.

[61] Lin Shung Huang, Alex Rice, Erling Ellingsen, and Collin Jackson. Analyzing forged ssl certificates in the wild. In *2014 IEEE Symposium on Security and Privacy*, pages 83–97. IEEE, 2014.

[62] Sally Hubbard. Facebook's new plan doesn't protect your privacy, and neither does the ftc. https://edition.cnn.com/2019/05/

03/perspectives/facebook-data-privacy-ftc/index.html, 2019. [Online; accessed 2019-09-19].

[63] Syed Asad Hussain, Mehwish Fatima, Atif Saeed, Imran Raza, and Raja Khurram Shahzad. Multilevel classification of security concerns in cloud computing. *Applied Computing and Informatics*, 13(1):57–65, 2017.

[64] Luis-Daniel Ibáñez, Elena Simperl, Fabien Gandon, and Henry Story. Redecentralizing the web with distributed ledgers. *IEEE Intelligent Systems*, 32(1):92–95, 2017.

[65] Jim Isaak and Mina J Hanna. User data privacy: Facebook, cambridge analytica, and privacy protection. *Computer*, 51(8):56–59, 2018.

[66] Seny Kamara and Kristin Lauter. Cryptographic cloud storage. In *International Conference on Financial Cryptography and Data Security*, pages 136–149. Springer, 2010.

[67] Alex Kantrowitz. Here's how facebook tracks you when you're not on facebook. https://www.buzzfeednews.com/article/alexkantrowitz/heres-how-facebook-tracks-you-when-youre-not-on-facebook, 2018. [Online; accessed 2019-09-19].

[68] Pavan Kapanipathi, Julia Anaya, Amit Sheth, Brett Slatkin, and Alexandre Passant. Privacy-aware and scalable content dissemination in distributed social networks. In *International Semantic Web Conference*, pages 157–172. Springer, 2011.

[69] Moon Kim and Jee Chung. Sustainable growth and token economy design: The case of steemit. *Sustainability*, 11(1):167, 2019.

[70] Sabrina Kirrane. *Linked data with access control*. PhD thesis, 2015.

[71] Sabrina Kirrane, Alessandra Mileo, and Stefan Decker. Access control and the resource description framework: A survey. *Semantic Web*, 8(2):311–352, 2017.

[72] Sabrina Kirrane, Serena Villata, and Mathieu d'Aquin. Privacy, security and policies: A review of problems and solutions with semantic web technologies. *Semantic Web*, 9(2):153–161, 2018.

[73] Barbara Kitchenham and Stuart Charters. Guidelines for performing systematic literature reviews in software engineering. 2007.

[74] David Koll, Jun Li, and Xiaoming Fu. Soup: an online social network by the people, for the people. In *Proceedings of the 15th International Middleware Conference*, pages 193–204. ACM, 2014.

[75] Yann Krupa and Laurent Vercouter. Handling privacy as contextual integrity in decentralized virtual communities: The privacias framework. *Web Intelligence and Agent Systems: An International Journal*, 10(1):105–116, 2012.

[76] John Kubiatowicz, David Bindel, Yan Chen, Steven Czerwinski,

Patrick Eaton, Dennis Geels, Ramakrishna Gummadi, Sean Rhea, Hakim Weatherspoon, Westley Weimer, et al. Oceanstore: An architecture for global-scale persistent storage. In *ACM SIGARCH Computer Architecture News*, volume 28, pages 190–201. ACM, 2000.

[77] Fen Labalme and Jad Duwaik. An infomediary approach to the privacy problem. *Feb*, 9:24, 1999.

[78] Christopher Lemmer Webber, Jessica Tallon, Erin Shepherd, Amy Guy, and Evan Prodromou. Activitypub, w3c recommendation 23 january 2018. https://www.w3.org/TR/2018/REC-activitypub-20180123/, 2018. [Online; accessed 2019-05-25].

[79] Jinyang Li and Frank Dabek. F2f: Reliable storage in open networks. In *IPTPS*. Citeseer, 2006.

[80] Q Vera Liao, Wai-Tat Fu, and Markus Strohmaier. # snowden: Understanding biases introduced by behavioral differences of opinion groups on social media. In *Proceedings of the 2016 CHI conference on human factors in computing systems*, pages 3352–3363. ACM, 2016.

[81] Natasha Lomas. Digi.me bags 6.1m to put users in the driving seat for sharing personal data. https://techcrunch.com/2016/06/30/digi-me-bags-6-1m-to-put-users-in-the-driving-seat-for-sharing-personal-data/, 2016. [Online; accessed 2019-08-26].

[82] Nate Lord. What is data encryption? definition, best practices and more. https://digitalguardian.com/blog/what-data-encryption, 2019. [Online; accessed 2019-09-05].

[83] Luís Miguel Oliveira Machado, Renato Rocha Souza, and Maria da Graça Simões. Semantic web or web of data? a diachronic study (1999 to 2017) of the publications of tim berners-lee. 2018.

[84] Essam Mansour, Andrei Vlad Sambra, Sandro Hawke, Maged Zereba, Sarven Capadisli, Abdurrahman Ghanem, Ashraf Aboulnaga, and Tim Berners-Lee. A demonstration of the solid platform for social web applications. In *Proceedings of the 25th International Conference Companion on World Wide Web*, pages 223–226. International World Wide Web Conferences Steering Committee, 2016.

[85] Theresa M Marteau, David Ogilvie, Martin Roland, Marc Suhrcke, and Michael P Kelly. Judging nudging: can nudging improve population health? *Bmj*, 342:d228, 2011.

[86] Brad McCarty. Facebook connect, oauth and openid: The differences and the future. https://thenextweb.com/socialmedia/2010/11/04/facebook-connect-oauth-and-openid-the-differences-and-the-future/, 2010. [Online; accessed 2019-08-22].

[87] Charles McMellon. Privacy-enhanced business: Adapting to the online environment. *Journal of Consumer Marketing*, 19:445–447, 09 2002.

[88] Marino Miculan and Caterina Urban. Formal analysis of facebook connect single sign-on authentication protocol. In *SOFSEM*, volume 11, pages 22–28. Citeseer, 2011.

[89] Satoshi Nakamoto et al. Bitcoin: A peer-to-peer electronic cash system. 2008.

[90] Arvind Narayanan and Jeremy Clark. Bitcoin's academic pedigree. *Communications of the ACM*, 60(12):36–45, 2017.

[91] Arvind Narayanan, Vincent Toubiana, Solon Barocas, Helen Nissenbaum, and Dan Boneh. A critical look at decentralized personal data architectures. *arXiv preprint arXiv:1202.4503*, 2012.

[92] Rammohan Narendula, Thanasis G Papaioannou, and Karl Aberer. A decentralized online social network with efficient user-driven replication. In *2012 International Conference on Privacy, Security, Risk and Trust and 2012 International Confernece on Social Computing*, pages 166–175. IEEE, 2012.

[93] Lily Newman. Facebook stored millions of passwords in plaintext—change yours now. https://www.wired.com/story/facebook-passwords-plaintext-change-yours/, 2019. [Online; accessed 2019-09-19].

[94] Irene CL Ng. New business and economic models in the connected digital economy. *Journal of Revenue and Pricing Management*, 13(2):149–155, 2014.

[95] Shirin Nilizadeh, Sonia Jahid, Prateek Mittal, Nikita Borisov, and Apu Kapadia. Cachet: a decentralized architecture for privacy preserving social networking with caching. In *Proceedings of the 8th international conference on Emerging networking experiments and technologies*, pages 337–348. ACM, 2012.

[96] Chris Oakes. Privaseek seeks attention. https://www.wired.com/1999/08/privaseek-seeks-attention/, 1999. [Online; accessed 2019-08-29].

[97] Steve O'Hear. Digi.me and personal merge to put you in control of the nascent 'personal data ecosystem'. https://techcrunch.com/2017/08/17/digi-me-and-personal-merge/, 2017. [Online; accessed 2019-08-26].

[98] Carlos Pedrinaci, Dong Liu, Maria Maleshkova, David Lambert, Jacek Kopecky, and John Domingue. iserve: a linked services publishing platform. In *CEUR workshop proceedings*, volume 596, 2010.

[99] Gareth W Peters and Efstathios Panayi. Understanding modern banking ledgers through blockchain technologies: Future of transaction processing and smart contracts on the internet of money. In *Banking beyond banks and money*, pages 239–278. Springer, 2016.

[100] Aravindh Raman, Sagar Joglekar, Emiliano De Cristofaro, Nishanth Sastry, and Gareth Tyson. Challenges in the decentralised web: The mastodon case. In *Proceedings of the Internet Measurement Conference*, pages 217–229. ACM, 2019.

[101] Shaan Ray. The difference between blockchains and distributed ledger technology. https://towardsdatascience.com/the-difference-between-blockchains-distributed-ledger-technology-42715a0fa92, 2018. [Online; accessed 2019-07-12].

[102] Elissa M Redmiles, Jessica Bodford, and Lindsay Blackwell. "i just want to feel safe": A diary study of safety perceptions on social media. In *Proceedings of the International AAAI Conference on Web and Social Media*, volume 13, pages 405–416, 2019.

[103] Erich Rickens. What is a distributed ledger? https://blog.blockport.io/what-is-a-distributed-ledger/, 2018. [Online; accessed 2019-07-11].

[104] Owen Sacco and Alexandre Passant. A privacy preference ontology (ppo) for linked data. In *LDOW*. Citeseer, 2011.

[105] Nat Sakimura, John Bradley, Mike Jones, Breno de Medeiros, and Chuck Mortimore. Openid connect core 1.0 incorporating errata set 1. *The OpenID Foundation, specification*, 2014.

[106] Andrei Vlad Sambra, Sandro Hawke, Tim Berners-Lee, Lalana Kagal, and Ashraf Aboulnaga. Cimba-client-integrated microblogging architecture. In *International Semantic Web Conference (Posters & Demos)*, pages 57–60, 2014.

[107] Andrei Vlad Sambra, Essam Mansour, Sandro Hawke, Maged Zereba, Nicola Greco, Abdurrahman Ghanem, Dmitri Zagidulin, Ashraf Aboulnaga, and Tim Berners-Lee. Solid: A platform for decentralized social applications based on linked data, 2016.

[108] Lorenz Schwittmann, Matthäus Wander, Christopher Boelmann, and Torben Weis. Privacy preservation in decentralized online social networks. *IEEE Internet Computing*, 18(2):16–23, 2013.

[109] Rajesh Sharma and Anwitaman Datta. Supernova: Super-peers based architecture for decentralized online social networks. In *2012 Fourth International Conference on Communication Systems and Networks (COMSNETS 2012)*, pages 1–10. IEEE, 2012.

[110] Sybil Shearin and Pattie Maes. Representation and ownership of electronic profiles. In *CHI 2000 Workshop Proceedings: Designing Interactive Systems for 1-to-1 E-commerce*. Citeseer, 1999.

[111] Ryan Singel. Facebook's gone rogue; it's time for an open alternative. https://www.wired.com/2010/05/facebook-rogue/, 2010. [Online; accessed 2019-07-08].

[112] Daniel A Smith, Max Van Kleek, Oshani Seneviratne, Mc Schraefel, Re Bertails, Tim Berners-lee, Wendy Hall, and Nigel Shadbolt. Webbox: Supporting decentralised and privacy-respecting micro-sharing with existing web standards. 2012.

[113] Daniel J Solove. A taxonomy of privacy. *U. Pa. L. Rev.*, 154:477, 2005.

[114] Sarah Spiekermann, Rainer Böhme, Alessandro Acquisti, and Kai-Lung Hui. Personal data markets. *Electronic Markets*, 25(2):91–93, 2015.

[115] Dominic Spohr. Fake news and ideological polarization: Filter bubbles and selective exposure on social media. *Business Information Review*, 34(3):150–160, 2017.

[116] Thorsten Strufe. Safebook: A privacy-preserving online social network leveraging on real-life trust. *IEEE Communications Magazine*, 95, 2009.

[117] Hassan Takabi, James BD Joshi, and Gail-Joon Ahn. Security and privacy challenges in cloud computing environments. *IEEE Security & Privacy*, 8(6):24–31, 2010.

[118] Mike Thelwall. Can social news websites pay for content and curation? the steemit cryptocurrency model. *Journal of Information Science*, 44(6):736–751, 2018.

[119] Allan Third and John Domingue. Linked data indexing of distributed ledgers. In *Proceedings of the 26th International Conference on World Wide Web Companion*, pages 1431–1436. International World Wide Web Conferences Steering Committee, 2017.

[120] Amin Tootoonchian, Stefan Saroiu, Yashar Ganjali, and Alec Wolman. Lockr: better privacy for social networks. In *Proceedings of the 5th international conference on Emerging networking experiments and technologies*, pages 169–180. ACM, 2009.

[121] Reza Tourani, Satyajayant Misra, Travis Mick, and Gaurav Panwar. Security, privacy, and access control in information-centric networking: A survey. *IEEE communications surveys & tutorials*, 20(1):566–600, 2017.

[122] Dinh Nguyen Tran, Frank Chiang, and Jinyang Li. Friendstore: cooperative online backup using trusted nodes. In *Proceedings of the 1st Workshop on Social Network Systems*, pages 37–42. ACM, 2008.

[123] Carmela Troncoso, Marios Isaakidis, George Danezis, and Harry Halpin. Systematizing decentralization and privacy: Lessons from 15 years of research and deployments. *Proceedings on Privacy Enhancing Technologies*, 2017(4):404–426, 2017.

[124] Daniel Trottier. *Social media as surveillance: Rethinking visibility in a converging world.* Routledge, 2016.

[125] Andranik Tumasjan and Theodor Beutel. Blockchain-based decentral-

ized business models in the sharing economy: A technology adoption perspective. In *Business Transformation Through Blockchain*, pages 77–120. Springer, 2019.

[126] Max Van Kleek, Daniel A Smith, Dave Murray-Rust, Amy Guy, Kieron O'Hara, Laura Dragan, and Nigel R Shadbolt. Social personal data stores: the nuclei of decentralised social machines. In *Proceedings of the 24th International Conference on World Wide Web*, pages 1155–1160. ACM, 2015.

[127] Max Van Kleek, Daniel Alexander Smith, Nigel Shadbolt, et al. A decentralized architecture for consolidating personal information ecosystems: The webbox. 2012.

[128] HS Venter and Jan HP Eloff. A taxonomy for information security technologies. *Computers & Security*, 22(4):299–307, 2003.

[129] Ruben Verborgh. Re-decentralizing the Web, for good this time. In Oshani Seneviratne and James Hendler, editors, *Linking the World's Information: Tim Berners-Lee's Invention of the World Wide Web*. ACM, 2019.

[130] Serena Villata, Nicolas Delaforge, Fabien Gandon, and Amelie Gyrard. An access control model for linked data. In *OTM Confederated International Conferences" On the Move to Meaningful Internet Systems"*, pages 454–463. Springer, 2011.

[131] Nick Vogel. The great decentralization: How web 3.0 will weaken copyrights. *J. Marshall Rev. Intell. Prop. L.*, 15:136, 2015.

[132] David Vorick and Luke Champine. Sia: Simple decentralized storage. *Nebulous Inc*, 2014.

[133] VoxMedia. Facebook-cambridge analytica scandal. https://www.theverge.com/2018/4/10/17165130/facebook-cambridge-analytica-scandal, 2019. [Online; accessed 2019-08-22].

[134] Marko Vukolić. The quest for scalable blockchain fabric: Proof-of-work vs. bft replication. In *International workshop on open problems in network security*, pages 112–125. Springer, 2015.

[135] Daniel J Weitzner. Whose name is it, anyway? decentralized identity systems on the web. *IEEE Internet Computing*, 11(4):72–76, 2007.

[136] Zack Whittaker. A huge database of facebook users' phone numbers found online. https://techcrunch.com/2019/09/04/facebook-phone-numbers-exposed/, 2019. [Online; accessed 2019-09-19].

[137] Wikisource. Index:intelligence community assessment - assessing russian activities and intentions in recent us elections.pdf — wikisource,, 2018. [Online; accessed 25-May-2019].

[138] Shawn Wilkinson, Tome Boshevski, Josh Brandoff, and Vitalik Buterin. Storj a peer-to-peer cloud storage network. 2014.

[139] Jiyi Wu, Lingdi Ping, Xiaoping Ge, Ya Wang, and Jianqing Fu. Cloud storage as the infrastructure of cloud computing. In *2010 International Conference on Intelligent Computing and Cognitive Informatics*, pages 380–383. IEEE, 2010.

[140] Chi Zhang, Jinyuan Sun, Xiaoyan Zhu, and Yuguang Fang. Privacy and security for online social networks: challenges and opportunities. *IEEE network*, 24(4):13–18, 2010.

[141] Zibin Zheng, Shaoan Xie, Hong-Ning Dai, and Huaimin Wang. Blockchain challenges and opportunities: A survey. *Work Pap.–2016*, 2016.

[142] Matteo Zignani, Sabrina Gaito, and Gian Paolo Rossi. Follow the "mastodon": Structure and evolution of a decentralized online social network. In *Twelfth International AAAI Conference on Web and Social Media*, 2018.

[143] Guy Zyskind, Oz Nathan, et al. Decentralizing privacy: Using blockchain to protect personal data. In *2015 IEEE Security and Privacy Workshops*, pages 180–184. IEEE, 2015.