



Privacy-aware Biometric Blockchain based e-Passport System for Automatic Border Control

Bing Xu, BSc (Hons), PGCert, MA
School of Computing and Communications
Lancaster University

A thesis submitted for the degree of
Doctor of Philosophy

September, 2023

I would like to dedicate this thesis to my principal supervisor Richard Jiang Min for his long-last support and patient guidance along the way. He is not only an academic role model but also a life guide. Also, my parents and my favourite twin sister, I am extremely grateful for having them to support and encourage me all the way through.

Declaration

I declare that except where specific reference is made to the work of others, the work presented in this thesis is, to the best of my knowledge and belief, original and my own work. The material has not been submitted, either in whole or in part, for a degree at this, or any other university. This thesis does not exceed the maximum permitted word length of 80,000 words including appendices and footnotes, but excluding the bibliography.

Privacy-aware Biometric Blockchain based e-Passport System for Automatic Border Control

Bing Xu, BSc (Hons), PGCert, MA.

School of Computing and Communications, Lancaster University

A thesis submitted for the degree of *Doctor of Philosophy*. September, 2023

Abstract

In the middle of 1990s, World Wide Web technology initially steps into our life. Now, 30 years after that, widespread internet access and established computing technology bring embodied real life into Metaverse by digital twin. Internet is not only blurring the concept of physical distance, but also blurring the edge between the real and virtual world.

Another breakthrough in computing is the blockchain, which shifts the root of trust attached to a system administrator to the computational power of the system. Furthermore, its favourable properties such as immutable time-stamped transaction history and atomic smart contracts trigger the development of decentralized autonomous organizations (DAOs).

Combining above two, this thesis presents a privacy-aware biometric Blockchain based e-passport system for automatic border control(ABC), which aims for improving the efficiency of existing ABC system. Specifically, through constructing a border control Metaverse DAO, border control workload can be autonomously self-executed by atomic smart contracts as transaction and then immutably recorded on Blockchain. What is more, to digitize border crossing documentation, biometric Blockchain based e-passport system(BBCVID) is created to generate an immutable real-world identity digital twin in the border control Metaverse DAO through Blockchain and biometric identity authentication. That is to say, by digitizing border crossing documentation and automatizing both biometric identity authentication and border crossing documentation verification, our proposal is able to significantly improve existing border control efficiency.

Through system simulation and performance evaluation by Hyperledger Caliper, the proposed system turns out to be able to improve existing border control efficiency by 3.5 times more on average, which is remarkable. What is more, the dynamic digital twin constructed by BBCVID enables computing techniques such as machine learning and big data analysis applicable to real-world entity, which has a huge potential to create more value by constructing smarter ABC systems.

List of Publications

Contributing publications

B. Xu, Q. Ni, R. Jiang, *et al.*, “Biometric blockchain (bbc) based e-passports for smart border control,” in *Big Data Privacy and Security in Smart Cities* (Advanced Sciences and Technologies for Security Applications), Advanced Sciences and Technologies for Security Applications. 2022, ch. Chapter 13, pp. 235–248, ISBN: 978-3-031-04423-6 978-3-031-04424-3. DOI: 10.1007/978-3-031-04424-3_13

B. Xu; T. Agbele, Q. Ni, *et al.*, “Biometric blockchain a secure solution for intelligent vehicle data sharing,” *Springer Cham.*, 2020. DOI: 10.1007/978-3-030-32583-1_11

B. Xu; T. Agbele, and R. Jiang, “Biometric blockchain a better solution for the security and trust of food logistics,” *IOP Conference Series: Materials Science and Engineering*, vol. 646, 2019. DOI: 10.1088/1757-899X/646/1/012009

B. Xu; R. Jiang; and Q. Ni, “Privacy-aware biometric blockchain based e-passport system for automatic border control,” *IEEE Transactions on Information Forensics and Security*, 2023 *Under Review*

Acknowledgements

Without the support of my family, friends, and supervisor team, I would not have come this far in my academic journey. First and foremost, I greatly appreciate and sincerely thank my principal supervisor, Richard Jiang. My gratitude for him regarding this academic achievement is profound. Additionally, I would like to express my gratitude to Prof. Ni Qiang and Dr. Leandro Soriano Marcolino for their supervision during my studies at Lancaster University.

To my dear mum, dad, and twin sister, I always find it impossible to find the right words to express how fortunate and happy I am to have them in my life. Their presence means the world to me, not to mention their enormous love and unwavering support.

Last but certainly not least, I want to thank my fiancé, Matthew S. Bradbury, for always being there for me whenever I needed you.

Contents

1	Introduction	1
1.1	Problem Overview	1
1.2	Research Question	2
1.3	Thesis outline	3
2	Related Work	5
2.1	Biometric IdM System	5
2.1.1	Preserving biometric IdM privacy	6
2.1.1.1	Biometric data protection regulations	6
2.1.1.2	Biometric data privacy preserving methods	7
2.1.2	Protect Biometric IdM security	13
2.2	Established IdM Systems	15
2.2.1	Fast Identity Online(FIDO)	15
2.2.2	Public key infrastructure(PKI)	16
2.3	Self Sovereign Identity(SSI) and SSI IdM	16
2.4	Biometric Blockchain based SSI IdM	21
2.5	E-Passport Systems and Smart Border Control	28
2.5.1	Smart Border for Immigrant Control	29
2.5.2	Smart Border for Customs Control	31
2.6	Chapter Summary	33
3	Identity and Identifier in the Metaverse	35
3.1	Digital Twin in the Metaverse	35
3.2	Identity in the Metaverse	36
3.2.1	Identity definition and identity category	37
3.2.2	Biometric identity	42
3.3	Identifier in the Metaverse	53
3.3.1	Identifier definition and generation	55
3.3.2	Uniform resource identifier (URI) and URI generic syntax	58
3.3.3	Decentralized identifier (DID)	61

3.4	Identity Credentials	63
3.4.1	Digital Verifiable Credentials	65
3.5	Chapter Summary	66
4	Identity Management System	67
4.1	Identity management(IdM)	67
4.1.1	IdM principles	69
4.1.2	Digital identity authentication	71
4.2	Established IdM Protocols	75
4.2.1	HTTP authentication	75
4.2.2	Single Sign On(SSO)	76
4.2.2.1	Open Authorization(OAuth 2.0)	77
4.2.2.2	OpenID Connect(OIDC)	78
4.3	IdM Risk Management	79
4.3.1	Level of assurance(LOA)	81
4.3.2	Risk assessment	82
4.4	IdM Security and Data Privacy	83
4.4.1	Preserving identity privacy	84
4.4.2	Protect IdM security	85
4.4.3	Network security	86
4.5	Chapter Summary	86
5	Blockchain	88
5.1	Blockchain System Overview	88
5.1.1	Decentralized distributed system	97
5.1.2	Decentralized Autonomous Organizations (DAOs)	99
5.1.3	Microservice	102
5.2	Key Components in Blockchain	105
5.2.1	Transactions	105
5.2.1.1	Merkle Tree	114
5.2.1.2	Smart Contracts	116
5.2.1.3	Digital Signature	122
5.2.2	Consensus rule	124
5.2.3	Wallets	130
5.3	Bitcoin, Ethereum, and Hyperledger	132
5.3.1	Bitcoin	132
5.3.2	Ethereum	133
5.3.3	Hyperledger	134
5.4	Blockchain Major Issue and Threats	136
5.4.1	Major issue	136

5.4.2	Security threats	138
5.5	Chapter Summary	139
6	Cloud, Edge and Fog Computing	140
6.1	Internet of Thing(IoT)	140
6.1.1	IoT features	142
6.1.2	Pros and Cons	143
6.2	Cloud	144
6.3	Edge	148
6.4	Fog Computing	149
6.5	Chapter Summary	150
7	Privacy-aware Biometric Blockchain-based e-Passport System for Automatic Border Control	152
7.1	System Overview	152
7.2	BBCVID for e-Passport	154
7.2.1	Main IdM Components in BBCVID	157
7.2.2	Main Participants in BBCVID	158
7.2.3	Biometric Identity Authentication and BBCVID Main Operations	159
7.3	VS for Automatic Border Control	168
7.3.1	Main Components in VS	170
7.3.2	Main Participants in VS	172
7.3.3	System Modelling	175
7.3.4	VS for Automatic Immigrants Control	177
7.3.5	VS for Automatic Customs Control	178
7.4	Chapter Summary	180
8	Implementation and Performance	181
8.1	Implementation	181
8.1.1	Decentralized Application(DApp): Android Mobile Application	187
8.2	System Simulation	194
8.2.1	BBCVID: Biometric E-passport	195
8.2.2	VS: Virtual Stamping for Border Crossing Events	202
8.3	Performance Evaluation	204
8.3.1	Performance of the BBC ABC System	204
8.3.2	Performance as Biometric Blockchain-based Identity System . .	209
8.4	System Security Assessment	213
8.4.1	System Security Evaluation	213
8.4.2	Anti-Impersonation Capability Discussion	216
8.5	Chapter Summary	218

9	Conclusions and Future Work	219
9.1	Summary of Work	219
9.2	Research Questions Revisited	221
9.3	Contributions	224
9.4	Future Work	225
	References	227

List of Figures

2.1	A facial image is simple encrypted which annihilates image neighbourhood feature completely [195]: left: original image, right: encryption image	8
2.2	Linear single point value distortion of facial image: left: original picture, middle: distortion value 7, right: distortion value 15	10
2.3	A demonstration of possible attacks or threats in biometrics identity systems [111].	14
2.4	A flowchart of data integrity check-up via PKI certificate.	17
2.5	Border e-gate facilitation in London Heathrow airport: swinging door, screen device, passport reader, and biometric identity reader [555] . . .	30
3.1	A demonstration of identity category structure.	40
3.2	A demonstration of fingerprint level 1, 2, and 3 features [20].	48
3.3	a demonstration for face image collection, detection, and standardization.	53
3.4	An example of U.S. Food & Drug Product codes and product code builder [186].	56
3.5	URI syntax diagram and Examples of URI [524].	60
3.6	A sample of DID document[460].	62
3.7	ID credential classification based on where identity attribute comes from [111].	64
3.8	A demonstration of verifiable credential's basic structure [121].	66
5.1	Block chain structure in detail.	91
5.2	Hard fork: Diverging a chain between new version consensus and old version consensus.	94
5.3	Soft fork: Stale block is still maintained by old consensus non-updated nodes.	95
5.4	A demonstration of centralized, federated, and decentralized distributed network structure.	98
5.5	Transaction scripting: a demonstration of scripting signature when spending a pay-to-public_key-hash output [70].	109

5.6	Raw transaction: an example of Bitcoin simple raw transaction [70]. . .	111
5.7	Merkle tree: An example of Merkle tree structure and the bottom-up construction process[70].	115
5.8	Smart contract processing: A demonstration of smart contract processing procedure [503].	119
6.1	A demonstration of IoT features.	141
6.2	A demonstration of Cloud service models.	146
6.3	Information flow in Cloud environment.	148
7.1	Full workflow of the proposed BBC ABC system.	155
7.2	Enrolment private data exchange through Biometric Open Protocol [244].	160
7.3	VS overview	176
8.1	Hyperledger Fabric test network local builds.	182
8.2	User Login.	189
8.3	New user registration.	189
8.4	New user registration success.	190
8.5	User database in Google Firebase.	190
8.6	Google Firebase email authentication scheme.	191
8.7	Ubuntu command line logs for acknowledging bbcvid channel is created.	193
8.8	Docker image lists for all peer nodes in BBCVID.	193
8.9	BBCVID HMRC and DVLA tree structure after certificate authority configuration: tls-cert.pem, ca-crt.pem, IssuerPublicKey, IssuerRevocationPublicKey, and msp are generated.	194
8.10	Biometric e-passport application form.	196
8.11	Facial biometric identity authentication and applicant’s current location.	196
8.12	Biometric DID document.	197
8.13	The proposed e-passport representation in mobile application.	197
8.14	The proposed e-passport DID documentation.	198
8.15	Vehicle DID application form.	199
8.16	The proposed vehicle DID document representation in mobile application.	199
8.17	The vehicle DID documentation.	200
8.18	Commercial border crossing permit application form.	201
8.19	The proposed VS border crossing records representation in mobile application.	201
8.20	BBCVDI: Average latency comparison among 1, 4, 7, and 10 orderer nodes.	213
8.21	VS: Average latency comparison among 1, 4, 7, and 10 orderer nodes. .	213
8.22	BBCVID:Min. and max.latency comparison among 1, 4, 7, and 10 orderer nodes for 100 byte blind write.	214

8.23	VS:Min. and max.latency comparison among 1, 4, 7, and 10 orderer nodes for 100 byte blind write.	214
9.1	A demonstration of an initial idea for the BBC ABC Metaverse DAO interface.	226

List of Tables

2.1	uPort, Sovrin, and ShoCard feature comparison.	18
3.1	Extracted feature for fingerprint template	51
3.2	Extracted final facial identity feature vector	54
5.1	A raw transaction follows the top-level format [70].	110
8.1	A comparison between existing and the proposed border control system in the regard of border crossing duration.	209
8.2	Performance comparison of blind write 1000 and 100 byte key value on BBCVID	211
8.3	Performance comparison of blind write 1000 and 100 byte key value on VS	212
8.4	Blind write packet loss rate comparison between BBCVID and VS. . .	213

Chapter 1

Introduction

1.1 Problem Overview

During the COVID-19 pandemic in 2021 and very recent war-affair between Russia and Ukraine, border control becomes an issue. It is not only because there is a sharp increase in border crossing demand but also a shortage in border control officers due to sick leave. Plus, in accordance with [389], the border crossing demand in 2025 will reach 887 million for land, air, and sea travellers in the EU, compared with 722 million in 2020 with an increase rate of about 5 percent per year. Similarly, the estimated number of individual(non-commercial) files to be stored in the border control agent system will be 128 million at 2025 compared with 104 million at 2020, which the annual growth rate is about 4.6%. That is, existing border agent capability is capped; therefore, the border control efficiency needs to be improved.

What is more, passport as legal identity credential is requested to present at border checkpoint all the time; however, existing centralized surrogate e-passport identity management system bears enormous risks. Like for instance, exposing itself as a single point of malicious attack, system administrator functional creep for user footprint tracking, system breakdown, and user private data breaching etc.

Most important, since current e-passport still requires a passport book to carry a microchip so that biometric identity data can be embedded on it, biometric identity authentication is only able to conduct at where the passport book is. Even though existing e-passport is portable, it suffers all the difficulties embedded with physical distance.

To add on, border crossing involves a large number of documentation verification, especially for commercial business entity who is required to declare customs. It normally is the most time consuming procedure among the entire border crossing workflow, which the existing method is to conduct the documentation verification manually.

Last but not least, in theory, efficient smart border control depends on constructed digital twin of every real-world individual person and the logistic transportation vehicle so that supervision upon all border crossing events can be effectively and efficiently conducted; However, a digital twin like that is still not yet to come. The reason of that can be summarised into two folds. First, individual digital twin requires fidelity. It means and requires that the individual's corresponding digital identity is able to be authenticated by real world legal identity credential. In the case of border control, it refers to use digitized passport to authenticate the individual's digital twin identity. By the same token, logistic transportation vehicle digital twin requires vehicle's entire attributes to be mapped into its digital twin but not just driving number plate. That is to say, vehicle's ownership has to be able to be authenticated in its digital twin. Second, verifying the present of a bodily human over open internet. That is to say, constructing individual bodily person's digital twin is difficult due to the fact that verifying the present of a bodily person over open internet is problematic, and digital identity is forgeable by identity thief through computational manipulations.

1.2 Research Question

In 2019, W3C has already published the “verifiable credentials data model 1.0: expressing verifiable information on the web[121]”, which is deemed as the standard data model of creating digital verifiable credential. It means creating a verifiable digital twin for real world identity credential such as university degree certificate and birth certificate is solvable. However, creating an individual bodily person's digital twin is still hard to fulfil. The difficulty mainly comes from that it is extremely hard to verify the present of a human being over open internet. As Peter claims “no one knows you are a dog if you are on internet [525],” it points the same problem out. Therefore, digital identity authentication is always a reasonable risk-based assurance process[204], [386].

To be more precise, even though the digital twin of the real world identity credential is solvable, the mapping between the digital identity credential and the bodily human person is still difficult. Indeed, that difficulty is derived from the mechanism of the digital identity being authenticated, which normally is either knowledge-based or challenge-based scheme. That is to say, either the knowledge or the challenge nowadays is easily to be stolen or sorted by identity thief through computational manipulations.

In real world border control scenario, regardless of how the border crossing legitimacy is verified, it has to be guaranteed that border crossing permit is only given to the lawful right person. Therefore, it is required that when border control efficiency is improved, the border crossing accuracy cannot be sacrificed. To sum it

up, in this thesis, there are three principal research questions(RQ) are addressed:

RQ 1: How to create a digital twin for individual bodily person in the Metaverse?

RQ 2: How to effectively improve the border crossing efficiency overall?

RQ 3: How secure and robust is the proposed system in terms of preserving identity privacy and protecting system security?

1.3 Thesis outline

To answer RQ 1, the main obstacle of verifying the present of a bodily person over internet has to be conquered. In current literature, the optimal answer of that is biometric identity authentication. Therefore, in **Chapter 2 Related Work**, biometric identity management(IdM) system is well reviewed, including detailed and dominated data privacy preserving methods and system security protecting methods. Most important, it has been mentioned in ‘Problem Overview’ that existing e-passport system suffers criticism of being a centralized system and bears huge risk. Therefore, both fast identity online(FIDO) and public key infrastructure(PKI) are reviewed as two established authentication system in current literature that have the most potential to replace the centralized IdM systems. Apart from that, the new trending decentralized self sovereign identity(SSI) and SSI IdM are reviewed as well. In that subsection(chapter 2.4), the research focus is given to “how to preserve biometric data privacy in SSI”. At the end of chapter 2, in order to answer RQ 2 well, existing e-passport and smart border control system are reviewed, which aims to identify the main obstacles and challenges in existing smart border control system.

To add more onto RQ 1, further investigation about identity and identifier in the Metaverse(**Chapter 3**) is conducted. That is, we clarify and distinguish the three separate concept: real life identity, identity in the Metaverse, and the digital twin identity in the Metaverse. Particularly, by giving a clear and well-defined explanation about identity and identifier at the very beginning, properties and functions of digital twin identity in the Metaverse can be well understood. Plus, decentralized identifier and identity credentials are also well researched in this chapter, which includes decentralized identifier and the digital verifiable credentials.

In order to answer RQ 2 and RQ 3, the proposed automatic border control solution is constructed as an integration of two Hyperledger Fabric Blockchain. Specifically, one chain is a biometric Blockchain based IdM system, which is able to generate the digital twin of a real world bodily person, vehicle, and commercial business entity. The other chain is a Blockchain Metaverse Decentralized Autonomous Organization DAO

for the border control agency, which is also where the IdM Blockchain generated digital twin being projected to. Therefore, **Chapter 4 Identity Management System** and **Chapter 5 Blockchain** are well researched.

In **Chapter 6**, the privacy-aware biometric Blockchain based e-passport system for automatic border control solution is fully proposed, and then in **Chapter 7** the system implementation and performance evaluation is offered. Finally, it comes to **Chapter 8** where a summary of this thesis, research question revisit, contributions, and future work recommendations are made.

Chapter 2

Related Work

2.1 Biometric IdM System

Digital identity management system is always problematic in open internet. In general, traditional digital identity authentication mechanism is dominated by knowledge-based credentials like password or PIN number, which nowadays is not considered secure enough any more [162]. To be more precise, the digital identity normally is a combination of a digital identity(identifier) and identity attributes(credentials) [437]. Specifically, a user's identity is represented by a digital identifier which can be identified and called remotely over internet. The digital identity attributes normally is a proof of a user has full control over a digital identity [162].

The straightforward issue in these traditional digital identity authentication methods is that passwords are way easy to breach and forget. Password fatigue and password thieves cause digital identity misuse and bring huge risk to online service provider [291]. They are getting more and more insecure about who on earth they are providing their service to. To tackle that, biometric identity authentication sheds its light on digital identity authentication solution as biometric identity data is globally unique, persistent, and intrinsic to one and only real human being. By deploying biometric identity authentication, it does not only authenticate a digital identity but also proves the present of a real bodily human being over the thin air [210].

From traditional centralized or federated surrogate digital identity system such as password-based systems, to nowadays popular biometric-based system, completely fool proof authentication is still not yet to come. On one hand, internet service provider requests higher and higher user identity assurance especially in the scenarios where there is a request for legal enforcement. On the other hand, robust quantum computing algorithms make spoofing attacks and data breaches way easier than before. Admittedly, biometric identity authentication has higher assurance and is

user friendly, but its application to the general public is hindered because of biometric data storage and transmission security concerns.

To be more precise, biometric identity management system has some significant challenges due to the nature of biometric data. That is,

1. **Publicly collectable.** human face identity, voice, gaits, and even fingerprint, they are private information, but it is publicly collectable from cameras, CCTVs, scanners and recorders. Even though both governmental and official organizations are trying the best to secure biometric identity information usage and preserve the privacy of biometric data by imposing biometric data storage and transmission regulations, biometric data publicly collectable nature is eternally contradictory with the requirement of confidentiality and privacy [110], [111].
2. **High intra- and inter-personal variance.** Due to the fact that biometric data collection process contains noises from both outside environment such as illumination, background, angles, and inside variances such as scars, dirt, pressure, and position etc, biometric data has high intra- and inter- personal variance [45]. It indeed draws a major shortcoming for verifying biometric identity data integrity after transmission over open internet.
3. **Permanency.** In accordance with [48], most biometric identity traits are stable and persistent, which means they are permanent and does not change over a person's life time. Therefore, once biometric identity is leaked or stolen, it is impossible for biometric identity owner to change to a new one. That is, permanency in biometric data becomes a serious drawback.

Therefore, a good practise of biometric IdM system has to be able to preserve privacy and protect system security.

2.1.1 Preserving biometric IdM privacy

Biometric identity information as intrinsic private data is required to preserve its privacy by both government and internet communities. There are few established regulations and standards to regulate biometric identity data conducts in the open internet.

2.1.1.1 Biometric data protection regulations

- Biometric Database and online transfer protocols – IEEE 2410-2019 - IEEE Standard for Biometric Open Protocol(BOPS) [244] defines the framework and specifications for multi-level access control and assurance for identity

management protocols. It adopts homomorphic encryption and a simple API to facilitate the application of this protocol, which is software supported mechanism [206].

- UK Confidentiality and Privacy Law of online personal data storage and transmission - General data protection regulations (GDPR) [402]. It profoundly and comprehensively defines a standardized mechanism for general data protection method, which principles, lawfulness, fairness and transparency are all well defined in the regard of conducting operations to access, rectification, erasure, restrict processing, data portability and objects.
- The ISO/IEC Standard 24745 on Biometric Information Protection - [86] provides a general guidance for the protection of biometric information. According to this standard, a protected biometric reference is typically divided into two parts, namely, pseudonymous identifier (PI) and auxiliary data (AD). Depending on how these two components are generated, biometric template protection schemes can be broadly categorized as:
 1. Feature transformation approach.
 2. Biometric crypto-systems.

A detailed review of biometric template protection approaches is available in [29], [199], [428] for in-depth analysis.

- Trusted execution environment(TEE) and secure element (SE) – TEE [417] is an isolated area in a hardware’s main processor, which is completely separated from the hardware device main operation system and designed to compute, store and process high security data, such as password, security generation, key distribution and verification etc [159]. To put in more detail, the applicable devices are smart phones, tablets, and set-top boxes etc, which the security of connected devices and the device itself are all required to have regulated security. Whereas as a contrast, a secure element is a processor that mainly aims to protect the device against unauthorized access and confidential data[404].

2.1.1.2 Biometric data privacy preserving methods

As in the regulations listed above, the transmission and storage of biometric data is not allowed to be *raw plain-text*, but rather has to be fully encrypted [86], [244], [402]. Since biometric IdM system is a very established subject, there are quite a few established and well-known biometric identity preserving methods available in the current literature. Even though the privacy-preserving methods are encryption

dominated, it turns out very effectively and robust[111].

A. Symmetry Encryption

Symmetry encryption refers to use the same key to encrypt and decrypt information, such as simple encryption and homomorphic encryption. Simple encryption generally refers to straightforward same key-based encryption and decryption, which is normally used to secure biometric identity information *locally* such as smart cards, mobile phone, and laptops. Therefore, a system is impossible to decrypt an encrypted biometric data without user vulnerably gives the encrypt key out; therefore, the data(encryption key and/or biometric identity information) exchange environment is deemed very secure.

In simple encryption, it normally requires a local server to host encryption key collection from user. Like for instance, biometric identity like facial image can be encrypted but annihilates neighbourhood features completely, which the result image is presented as above in **Figure 2.1** (left: original image, right: encrypted image). That is, to secure local biometric identity authentication or secure the biometric identity template storage at local device, simple encryption is always the first choice when the data exchange environment is very secure.

Most important, in simple encryption, the encryption key normally is very simple as well, such as PIN numbers. It is very easy to implement and has very low cost as well. However, in the case of biometric identity authentication online, simply encryption is not deemed as strong enough to protect biometric identity template.

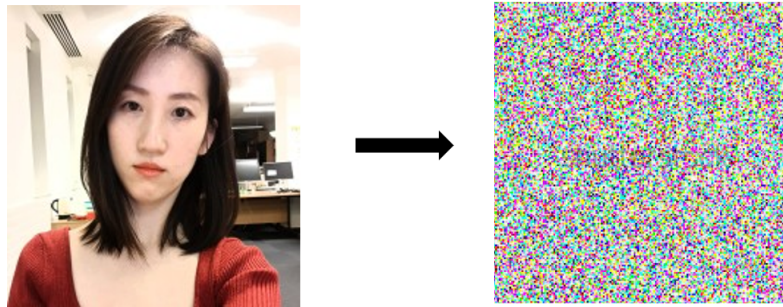


Figure 2.1: A facial image is simple encrypted which annihilates image neighbourhood feature completely [195]: left: original image, right: encryption image

By similar token, homomorphic encryption can be used to performed operations on cipher-texts directly without knowing its original plain-text at all [75], [111], [538]. That is,

$$E(x) \otimes E(y) = E(x \otimes y) \quad (2.1)$$

Where

$$E(x) = x^g \{key : (n, g)\} \quad (2.2)$$

And the homomorphic multiplication is

$$E(x) \cdot E(y) = x^g \cdot y^g = (x \cdot y)^g = E(x \cdot y) \quad (2.3)$$

However, the major drawback of homomorphic encryption once to be that it cannot perform simple addition, and it is semantically non-secure [111]. Later for the same public key $key : (n, g)$, Paillier [324] switched the exponential base from the data to be encrypted to a random base, which makes additions possible and way more efficient as in below formula

$$E(x) \cdot E(y) \propto g^x \cdot g^y = g^{x+y} = E(x + y) \quad (2.4)$$

Homomorphic encryption indeed is ideal, as verifier can authenticate any user's identity by simple operations on cipher-text directly without any knowledge on user's secret nor the plain-text of the transmitted raw data, which above function is the fundamental principle of zero knowledge proofs [183]. However, homomorphic encryption in biometric identity authentication normally very expensive in the regard of computational cost and system encryption library set ups which is not seen practical especially interoperability is considered as well [111].

B. Cryptography: Asymmetry Encryption

Similar with symmetry encryption, cryptography-based methods aim to encrypt original biometric identity data into cipher. However, the major difference between cryptography and symmetry encryption is cryptography has two different keys: one is for encryption and the other is for decryption. Plus, classical asymmetric cryptography has an irreversible relationship between the two keys (private key and public key). That is, even though public key is made publicly available, private key is still mathematically non-revertible and non-deducible from the public key.

Generally, an ideal cryptography system requires two elements [111]:

1. Irreversible encryption without the key.
2. Semantic security in the cipher. That is, two different encrypted ciphers should have no mutual information at all.

One significant drawback of cryptography solution is the cipher completely lost the feature of the original biometric data, which the biometric traits are covered by cryptography techniques [111].

C. Transformation

Consider the complexity of encryption key transfer in remote biometric identity authentication, it is possible to match biometric data directly in encrypted data domain, which eliminates the necessity of decryption and key transmission from the root. The obvious benefit of matching biometric identity in encrypted domain is the biometric identity turns to be revocable [180] as the identity feature extracted is derived from encrypted data but not user's original biometric features.

To add on, to further protect the channel of biometric data transmission, distortion-based transformation can still keep some of the topological structure of the original image. One demonstration is illustrated above as in **Figure 2.2** by linear single point value distortion method (left: original picture, middle: distortion value 7, right: distortion value 15).

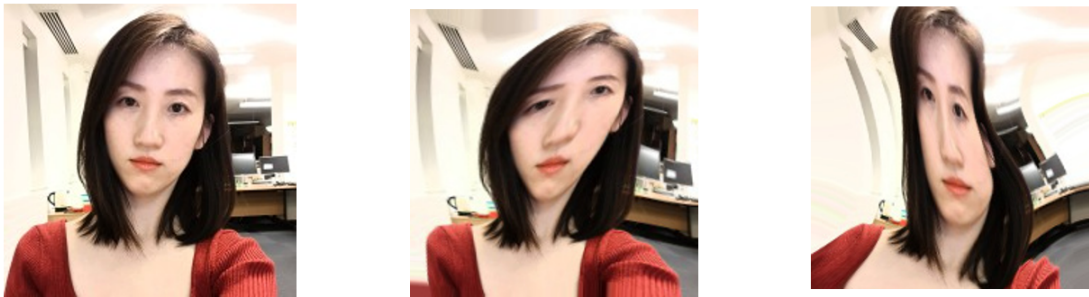


Figure 2.2: Linear single point value distortion of facial image: left: original picture, middle: distortion value 7, right: distortion value 15

What is more, non-reversible transformation on biometric data is commonly named as *cancelable biometrics*[372]. Compared with biometric encryption, biometric transformation has some generic rule which retains original data feature after transformation. Plus, transformation significantly involves in parameter pre-settings; therefore, it gives user priority to set a user-specified key/parameter. Most important, transformation-based approach gives verifier a flexibility to conduct the verification based on either the after-transformation data itself or features extracted from transformed data [345].

Due to transformation has to input parameters, biometric verification based on transformation method is commonly seen as a two-factor verification. That is, both

the parameter/key knowledge based secrets and the biometric identity itself. To put in more detail, below are conventional transformation methods:

- **Parameters/PIN** - Transformation depends on a user-specified parameter or key, which is

$$y = Transformation(biometricrawdata, user_specifiedkey/PIN) \quad (2.5)$$

- **Filters** - Random filters, PIN-based random kernel, which is theoretically based on minimum average correlation energy theorem as below

$$y = quantization(filter\ function(biometricrawdata)) \quad (2.6)$$

However, it has been proved that the adversarial system is able to guess the transformation function based on large enough training set. Therefore, recent research point is to make the transformation function into a computational intractable problem [111].

- **Cryptographic transform** - Shuffling or bio-hashing are two classical methods for biometric cryptography transformation [345]. Specifically, a bit-block shuffling key ‘1011001’ is the generic shuffling rule for all biometric data, and the Hamming distance is not affected by shuffling at all. In general, shuffling transformation is revocable, template diversity (not applicable to cross-database matching scenarios), protected against stolen biometric data, and privacy protection. Plus, de-shuffling, even with a prior information, is a nondeterministic problem [248]. However, in biometric IdM system, if the shuffling key is changed, the old biometric identity template has to be changed according as well. It requires the shuffling key exchange if the biometric identity authentication is conducted cross-platform, which hinder the application of this technique [267].

D. Biometric Encryption

As emphasized before, biometric raw data contains large inter- and intra-personal variance due to a lot of factors, which makes it especially difficult to input biometric raw data directly to get one identical fix string by cryptography. Luckily, there are some established algorithms in currently literature can fulfil that task, which they are in general called biometric encryption. The aim of biometric encryption is to extract a fixed length and persistent string from biometric data (a.k.a. biometric key), which at the same time allow in-exact inputs for encryption.

- **Error correction code** – initially used in communication system to reduce communication data variances caused by transmission channel and environmental hazards. The main concept behind an error correction code is to add large enough redundant information into contaminated raw data, so that error can be corrected accordingly. However, there is an eternal conflict in error correction algorithm. That is, the more redundant information that can be added to the contaminated raw data, the correction accuracy will go up accordingly. However, the redundant information is limited by the internet bandwidth which has been proved that there is a limit for the maximum amount of data it can transferred over the internet [424], [510].
- **Fuzzy extractor** – fundamentally a probability function model. In accordance with [102], [270], the supporting knowledge behind the fuzzy extractor is secure sketch and strong extractor, which in together makes noisy input exact reconstruction possible. That is, M is a strong randomness extractor, and its function is expressed as $Ext : M \rightarrow \{0, 1\}^l$ with r randomness. (m, l, ϵ) is called a strong extractor if for all m -sources W on $M(Ext(W, I), I) = \epsilon(U_l, U_r)$, and $I = U_r$ is independent of W . As for secure sketch, suppose any noisy input raw data w and its corresponding sketch s , given s and a value w' . If w' is close enough to w , w can be exactly recovered. However, one strong requirement in secure sketch is s can not expose any information about w at all. For a mathematical expression, secure sketch is:
 1. Suppose the noisy input is w , and the sketching procedure is S . S operates on w , and generate a helper string $s \in \{0, 1\}^*$.
 2. To exactly recovery input w , element w' is collected. If $dis(w, w') \leq t$, then $Rec(w', s(w)) = w$ and Rec is a recovery correctness function.

Instead of exactly recover the noisy input, fuzzy extractor [541] is to precisely recovery an input data generated data string. For a detailed explanation of fuzzy extractor, for any biometric raw data set B , sets a uniform distributed random numeric string R (PIN/parameters/secrets). As long a fresher collected biometric raw data set B' is close enough to B , R can be exactly recovered from B' . To put in more detail, a fuzzy extractor (m, l, t, ϵ) :

1. Generate (Gen). Suppose $w \in M$, which M is a metric space with distance function 'dis'. Gen outputs an extracted string $R \in (0, 1)^l$ and a helper string $P \in (0, 1)^\epsilon$.
2. Reproduce (Rep). If $dis(w, w') \leq t$ and $(R, P) \leftarrow Gen(w)$, then $Rep(w', P) = R$.

The output of above is R , which is uniform random sequences of bits and commonly is used as cryptography secret keys. Most important, fuzzy extractor can be very insecure if the fuzzy random secret is re-used after a few times. If attacker knows about the distribution of input data, the secret string key can be guessed [191]. Plus, fuzzy extractors in general suffers low-entropy issue, which the scheme itself requires more than one biometric type as input [102].

To put more in, [60] uses CNN deep learning technique to extract fixed string from facial image, so that transaction signatures can be signed by facial image. However, it needs to be fused with fixed RSA private key, and the security of that fusion is not discussed at all.

2.1.2 Protect Biometric IdM security

Communication over internet thin air makes distance invisible, which fundamentally conquers the difficulty embedded with distance, such as physical representation of an entity and information transmission. With the rapid usage of internet, biometric identity verification gradually replaces conventional knowledge-based or token-based identity authentication methods for extra security. However, biometrics data has to be used with extra caution because of the nature of biometric data itself. As discussed, biometric identity is password-less and persistent; therefore, it well tackles password fatigue issue and improves identity veracity significantly.

However, some major security concerns and risks hinder the generalization of biometric identity authentication solutions. To put in more details, the straightforward problem of biometric data is [110]:

1. Biometric data are public available like facial image and gait.
2. There is an established user-side attacks.
3. System challenges in the regard of biometric template protection, and transmission channel etc.
4. There is a rigour requirement for setting up or collect biometric data in both device and environment wise.
5. Authentication or verification performance is not stable.

Figure 2.3 is a demonstration of possible attacks or threats in biometric identity management systems [110]. To put in more detail, attacks to biometric identity authentication system can be derived from *synthetic biometric representations and brute-force trail*. Specifically, synthetic biometric representations are to use synthetic material to spoof a biometric identity feature so that a match score can

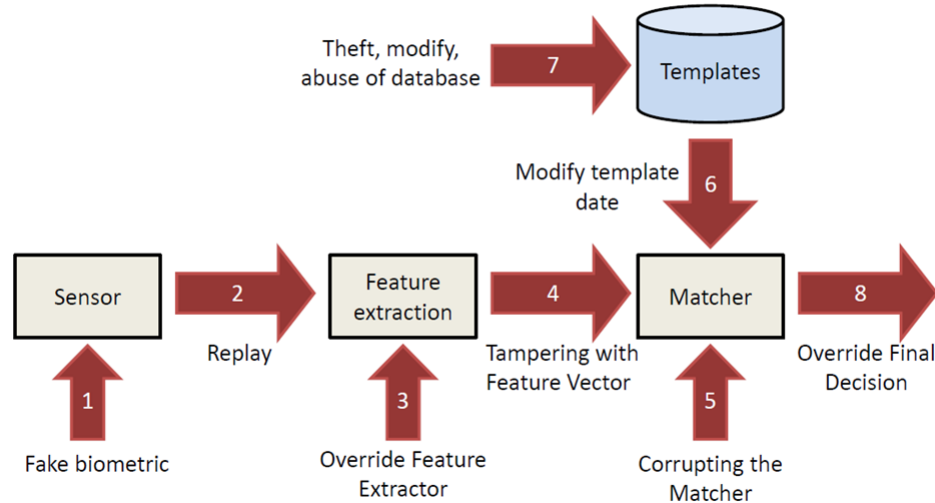


Figure 2.3: A demonstration of possible attacks or threats in biometrics identity systems [111].

be reached; brute-force trail attack refers to submitting a random biometric identity to consistently submit it until a false match is accepted [7], [35].

Similar with another critical attack derived from user interface is *spoofing attack*. That is, using face mask, contact lenses, and man-made material (rubber or art glue) to fake biometric trait like face image, iris and fingerprint etc. One classical countermeasure of spoofing is liveness detection.

To put more in detail, fingerprint liveness detection has very poor performance back the time of year 2003, which finger films are accepted by most of the tested sensors with an acceptance rate of 67% [539] only. However, strenuous work and efforts has been made to improve the liveness detection in recent years. For example, hardware based liveness detection methods [434] is much more expensive than software-based solutions that is to make the most use of already captured biometric data to detect liveness, such as perspiration phenomenon [478], dynamic features [117], CNN [263], and texture descriptor [287]. By 2018, a highest correct classification rate between a fingerprint film and a genuine fingerprint reaches 98.60% [263]. By similar token, a face biometric implementation might be fooled by family members who resemble the user or a 3D mask of the user as in impersonation. A fingerprint biometric implementation could potentially be bypassed by a spoof made from latent fingerprints of the user. Although anti-spoofing or Presentation Attack Detection (PAD) technologies [395] have been actively developed to mitigate such spoofing attacks, it belongs to mitigation but not prevention.

2.2 Established IdM Systems

2.2.1 Fast Identity Online(FIDO)

Fast identity online (FIDO) is a profitable industry association who maintains a standardized biometric identity authentication protocol for fast and stronger authentication solution [15]. Specifically, the protocol encourages password-less, phishing-resistant, and multi-factor authentication by employing public key cryptography, which is strongly resistant to phishing. Plus, USB, near field communication(NFC), and Bluetooth low energy(BLU) communication channels are used in FIDO mechanism to enable biometric data collection and transmission while initial enrolment [326]. FIDO also has established device onboard protocol for IoT install authentication secrets so that a more secure communication channel can be established [227]. However, FIDO is not decentralized identity, but rather is an established solution for biometric identity authentication solution that can be accomplished remotely over internet thin air by registering user's authenticator with remote authentication service.

The essential principle behind FIDO password-less identity authentication is the presentation of a live biometric sample results in unlocking access to a private key that can be used for authentication [226]. That is, FIDO authenticator contains two keys, an authentication key (private key) and an attestation key(public key, and it will be sent to service provider), which has a classic cryptography relation. Authenticator is registered at service provider, and user's identity is authenticated based on a cryptography challenge response. It offers the service provider with a very high assurance that the user being authenticated is indeed the same user who originally registered with the site. As part of the service, each public key is linked with an AppID, which is a URL carried user credentials and is assigned by service provider as well [227].

FIDO server is responsible for interacting with relying party web server to transfer messages between a FIDO universal authentication framework (UAF) client via a device user agent; validating FIDO UAF authentications against the configured authenticator metadata; manage user registered account for the relying party; evaluating user authentication and transactions confirmation responses to determine their validity [326].

To preserve user privacy in biometric identity data, FIDO only occasionally associates server-assigned username between a user and a relying party [15]. Whereas as a contrast, conventional IdM system requires a central authority certificate, which introduces a trusted third party. Third party normally mixes identity assertions with key cryptographic proofs, and may even be sent in the plaintext to initiate an internet conversation during handshake.

Most important, FIDO protocol keeps biometric identity is authenticated at where

the biometric identity template is stored. Therefore, the service provider is still totally ignorant about user's biometric identity at all. That is to say, it is still possible for one user to claim multiple identities in one FIDO IdM system(e.g. purchase another FIDO authenticator and register at the same service provider), which is seriously violates identity persistent rule.

2.2.2 Public key infrastructure(PKI)

Public key infrastructure(PKI) is verifiable digital identity endorsed by certificate authority(CA)'s digital signature and certificate [230], which the verifiable digital identity specifically is anyone's public key. To put more in detail, in PKI, a PKI based digital identity is a public- and private-key pair. Public key and private key are in conventional cryptography relation. That is, the public key is used as user's identity identifier, and the corresponding private key is user's identity credential that can be used to prove the ownership over the identity [230].

To add on, see **Figure 2.4** for a demonstration of how to use PKI certificate to prove the authenticity of a user's public key by RSA signature algorithm. To be more precise, PKI certificate(e.g. x.509 certificate [231]) contains user's identity information, user's public key, and CA's information. CA will hash those three components of information first, and then digitally sign a signature on that hash data, and finally attached it to the end of the PKI certificate. PKI indeed means to build a tie between user's information with user's public key through CA's certificate. Anyone who is interested in verifying the authenticity of user's public key, they can hash certificate information by himself first, and then decrypt CA's certificate to retrieve the CA authenticated user information. two hash outputs finally are revealed and can be matched to verify the authenticity.

Particularly, by deploying digital signature in CA's digital certificates, data veracity, ownership and integrity can all be evaluated very efficiently [230]. PKI is widely used for national ID system such as e-passport, which PKI specifically is used to proof the authentication of a digital identity credentials. Currently, the main challenging in PKI system is user's private key protection scheme. As user's private key is very difficult to remember, which it normally is stored in a microchip or token.

To put PKI in another way, PKI indeed is to *build up a tie* between user's information with a public key that is endorsed by a trusted CA. Therefore, anyone is available to user's PKI certificate is able to authenticate that *tie*.

2.3 Self Sovereign Identity(SSi) and SSI IdM

Conventional centralized or federated IdM system cannot meet user's expectation any more due to the fact that it exposes more and more shortcomings. Centralized

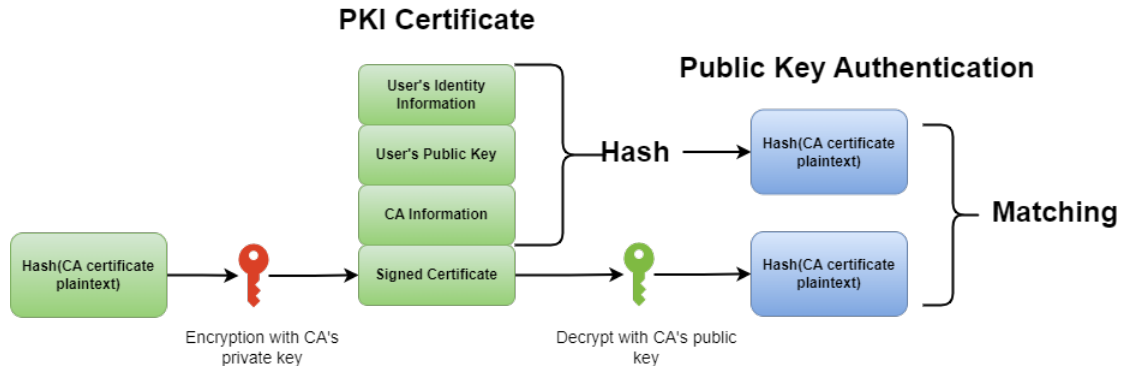


Figure 2.4: A flowchart of data integrity check-up via PKI certificate.

IdM suffer few major pain: First, central system administrator functional creep for data breach and user footprint tracking. Second, centralized IdM exposes itself as an attacking point for malicious attack, and the lose of one system will cause millions identity lose. Third, user has no control over the identity at all. To tackle all those three drawbacks, decentralized and distributed Blockchain technology sheds its light on decentralized SSI IdM which draws exclusive attention.

SSI is also known as self-managed identity or user controlled identity [457], which user has broader control over his own identity. [449] has defined some important features of a self-sovereign identity(SSI), which include:

- **Independent existence.** The existence of SSI is not dependent on any outsider nor any other entities.
- **Control.** User of SSI should has full control over the SSI, like storage, and CRUD operations.
- **Access.** User's access to his own SSI should never be denied.

Survey paper [457] claims that the privacy root of SSI is the introduction of Diffie-Hellman key-exchange system which allows individual users to securely exchange private data in open internet based on public key cryptography, which faces quite a few challenges. To be more precise,

- **Root of trust.** SSI IdM gives individual rights to issue digital identity on user's own behalf, which means every participants in SSI IdM is able to issue any digital identity to himself. Therefore, it leads to a question of how reliable of these digital identities are. As per request, all IdM systems should maintain identity persistence property, which means one person should only be able to

IdM	Network		Security		Privacy	
	Permissioned	Public	Basic Security	Multi-Lateral	Anonymity	Remote Admin
uPort	x	✓	✓	x	✓	✓
Sovrin	✓	x	✓	✓	✓	x
ShoCard	✓	x	✓	✓	✓	✓

Table 2.1: uPort, Sovrin, and ShoCard feature comparison.

claim one identity only in one IdM system. In SSI IdM, it seems impossible to stop one user from claiming multiple digital identity to the same service provider, which causes significant security hazard.

Therefore, current SSI IdM system are still dominated by the existence of a central authority who takes the responsibility to supervise the whole system, or a certificate authority who endorses SSI as in a decentralized PKI to add credibility to it.

- **Management.** As an IdM system, encryption key exchange or private data(credential) exchange seems to be necessary. The backbone of SSI IdM is Blockchain which is peer-to-peer organized; therefore, it lacks of a host and management for key exchange. Plus, if every user’s digital identity is issued from an SSI IdM, none is confident about who they are really exchange key with, which is another serious security hazard.
- **Accountability and governance.** In a well maintained IdM system, accountability and governance is always crucial as any behaviour leading to security hazards and threats should be effectively detected and governed. However, in SSI IdM, the lack of central authority makes the accountability and governance difficult.

Even though Blockchain does not have a long history, there are some very established blockchain based SSI IdM system in current literature already, which most of them are semi-decentralized and managed in a self-sovereign manner, like Sovrin, Uport, and ShoCard. The aim is to build a digital IdM system that is a proxy of physical world identity credentials to authenticate who you are in the digital world. Specifically, Uport [320], ShoCard [446], and Sovrin [459] are three of the most established decentralized IdM applications in current market. They all build upon decentralized network and generate user fully self-sovereign identities. See **Table 2.1** presents a condensed comparison among these three Blockchain based IdM systems.

- **Uport** [320] is open source framework for decentralized identity. It is built on top of Ethereum network and maintained by smart contracts, which aims to create decentralized identities for banking and emailing services. Smart contracts take responsibility to maintain all identities, which all contracts’

addresses are stored in a token. Specifically, a proxy smart contract will act as a permanent identifier for a Uport user. Identity attributes are encrypted and stored in Interplanetary File System(IPFS); however, IPFS content address registry is centralized. Even though Uport further constrains read/write function to the registry, centralized registry still bears more risks with regard to data breach. Distinctively, Uport offer identity recovery mechanism. That is, each Uport ID is linked with at least one delegate who has the right to function on behalf of that user, which puts delegates at the targets spot for attackers. Most important, Uport deploys API to transfer topic-based information (json object) from one end to another, which is especially useful when Uport identity communicates with a browser.

Major drawbacks in Uport are two folds. That is, first, data stored and processed on server is not encrypted, which draws a privacy concern. Second, the delegates are public available with privilege voting rights to uport identity; therefore, it exposes delegates at an attacking spot.

- **Sovrin** [459] is open sourced and decentralized identity. Minders are limited to trusted institutions(service providers), and each identity has one pair of public and private key. The identity is issued, distributed, and replicated by those qualified miners, and a redundant Byzantine fault-tolerant protocol is imposed for keeping the consensus in the network. Sovrin build upon a permissioned Hyperledger Indy network and uses cryptography zero-knowledge proofs, which gives user freedom to decide what identity attribute they intend to share with the third party. Sovrin aims to uses one identity to achieve multi-platform login, and deploys two blockchain rings to improve scalability. Specifically, one ring is used by minders nodes only, and the other ring is used by general public who only has read function.

Major drawbacks in Sovrin is the identity is not fully self-sovereign by identity user, but identity user has the full control over what identity attribute to share with identity verifier through zero-knowledge proofs.

- **ShoCard** [446] A token-based and decentralized travel identity with built in facial recognition mechanism. ShoCard aims to enhance air travel customer experience and improve efficiency at traveller checkpoints. Specifically, a mobile application is developed to upload traveller travel documents and self-portrait. A fresher new facial image will be captured at the airport kiosk to complete facial recognition. If all information is authenticated, a digital single travel token is issued to the traveller to pass-through all gates in airports. ShoCard in general is a public key infrastructure where the ShoCard server takes the responsibility to transfer messages between ShoCard user and airport agencies

(public key certifier). Major drawbacks in ShoCard is facial image still needs to transmit between certifier and user. Even though the data is encrypted in envelope, there is still a risk for man-in-the-middle attack. Plus, if certifier stop issue certificates to user, the corresponding identity loses credibility completely.

For another example, [207] proposes a blockchain based identity-as-a-service system in Cloud network which is named as BIDaaS. Cloud reduces system infrastructure costs with integrated management and is easier to use. Specifically, user firstly submits his identity credentials to BIDaaS provider and then a verifiable identity transaction is created by BIDaaS provider on a private blockchain that is maintained by BIDaaS provider as well. Service provider can verify user's identity through verifying the transactions signature.

What is more, [19] proposes use domain name system-like mechanism to generate decentralized identifier for identity user, which accomplishes the goal of self-sovereign identity management. To the best efforts being made to improve the system security level and performance, a private Ethereum network is deployed. Specifically, the decentralized identity identifier is generated by calling built-in smart contract. Similar with Uport [72], a proxy contract is used to complete required IdM task(CRUD) on behalf of user. Distinctively, the system implements a validate function to valid identity attributes before adding it to the decentralized identifier. All functional smart contracts are indexed by a similar domain name system mechanism so they can be called precisely when they are in need to fulfil a function.

For a short summary, for decentralized application(dApp) based IdM system, the major challenge lies in how to make identity issuer reliable and trust worthy as everyone is able to issue a digital identity for themselves. A lot of researches such as [19], [207], [320], [446], [459] recommend either a central authority(server) or a certificate authority(server) who play the role as a trusted entity within SSI system so that the trust entity can endorse SSI to gain extra credit.

Plus, securely linking identity attributes with decentralized identifier, and authenticating the decentralized identity effectively should also be taking care of. Traditionally, DID document is used to package all verification material, evidence and mechanism in itself to prove DID controller's ownership over corresponding DID, which requires the DID is resolvable to DID document. For a secure connection between identity attribute and identifier, there is a firm trending for deploying decentralized identifier. That is due to the fact that DID has been labelled as a standard for decentralized identity by W3C, which has benefits of improving interoperability, stronger anti-threats model, and low implementation cost. Most important, the decentralized identifier specifically can be managed either by self-executed smart contracts.

2.4 Biometric Blockchain based SSI IdM

In some more rigid context to authenticate a user's identity, biometric information normally will be required for extra assurance. As biometric identity information like fingerprint and voice is harder to fake than normal identity credentials like driving license or birth certificates, biometric-aware Blockchain based IdM gains more credibility as it is more reliable in confirming a person's identity. Especially when user's identity may incur legal disputes, biometric identity information is able to clear liability and legitimacy. Most important, the combination of Blockchain and biometrics brings biometric identity authentication system some desirable features like immutability, availability, and universal access [135].

To built a biometric IdM system on Blockchain network, three main things need to be tackled. That is,

1. Biometric identity template storage.
2. Biometric identity collection.
3. Authentication mechanism.

Biometric identity template storage

To start with, biometric identity template can be stored on Blockchain in three ways:

1. **On-Chain storage.** The benefits of storing user's biometric identity template on-chain are two folds:
 - (a) The universal availability of user's biometric identity template is guaranteed as long as the Blockchain is available. That is, the biometric identity template can be fetched directly from the Blockchain without re-directing to another location at all.
 - (b) As Blockchain transaction records are immutable, the recorded biometric identity template is immutable so the template integrity can also be guaranteed.

The main challenge of recording user's biometric identity template on-chain is to preserve user's privacy [196]. Even though biometric identity template has been encrypted, user's public footprint may be tractable for outsiders which is seen as a privacy hazard. Plus, if large training set is available for particular user's transaction, the biometric identity template may be able to be forged. Plus, on chain storage is often very expensive if it is directly storing encryption biometric identity template [135]. Therefore, storing (encrypted)biometric identity template directly on-Blockchain is not recommended [196].

2. **Off-chain storage.** Regardless of Blockchain, there are four main biometric identity template storage locations in general, which include IPFS [53], Cloud, smart cards, and user's end device. To store biometric identity template off-Blockchain, biometric identity template normally is transformed into a hash string and then stored the hash string on the Blockchain as either a transaction content [136] or part of the Blockchain Merkle tree data structure [364]. In later section 5.2.1.1, there is a detailed introduction about Merkle tree.

To add more details in, biometric identity template stored in IPFS and Cloud is required to define a template recollection and an indexing mechanism. That is to say, recording the hash string of the biometric identity template on Blockchain and then use it as an index to redirect identity authenticator to where the biometric identity template is stored. Therefore, the biometric identity template can be called at later time when authentication request is raised. However, the availability of the biometric identity template is significantly depends on the availability of Cloud and IPFS, which is not seen completely reliable [196].

[491] is a good example of combining Ethereum and IPFS to store biometric identity template off-chain, which fulfils the task of biometric identity authentication without any central authentication server. However, the drawback of that is the user has to generate multiple keys to encrypt multiple biometric identity templates so that privacy can be preserved. Plus, the encryption key has to be transmitted to the remote server to conduct authentication, which put the biometric identity template at risk.

Similarly, [384] stores user's biometric identity template in a centralized database and then hash it into a hash string. Hash string is recorded on-Blockchain as an index to call the biometric identity template from the database. However, the system is nearly centralized system as database administrator maintains all user's biometric identity templates and pushes its corresponding hash string to Blockchain. [325] separates biometric identity templates into n pieces, and each pieces is stored at a trusted entity in Blockchain system. It is similar with IPFS storage but preserves template privacy even more as one piece of template data lose does not lose any identity information at all especially when the user's identity is completely anonymous.

By similar token, biometric identity template stored in user's end device either supports local identity authentication only, or requires to register the user's authenticator at authentication server which enable remote biometric identity authentication. Last but not least, biometric identity template stored in smart cards are portable, which enables biometric identity authentication at where the smart card is; however, a smart card reader is required to read the smart card data.

Comparing storing whole biometric identity template on-chain, off-chains storage normally saves a lot of space and cost [135]. That is, an android mobile phone camera captured facial image normally has a size of 100KB to 200KB [467], but a SHA256 hash string or Merkle tree leaf has only 256 bits only. Most important, Blockchain in most cases has a storage limits which is about 200GB [364].

3. **Biometric as digital signature.** Transforming user's biometric identity template into Blockchain private key to sign Blockchain transactions like digital signature ECDSA[33] or Schnorr[433]. However, there are two major challenges have to be conquered. That is,

- (a) Inexact input and inexact output.
- (b) Size of the signature and/or the key.

That is, since biometric identity credential has inexact input and inexact output, it is very difficult to extract one persistent and constant alphanumeric string(key) from biometric identity template. Luckily, algorithms like fuzzy extractors can extract fixed string from noisy input through helper data's help [191]. However, it suffers some drawbacks like low entropy [191], massive input size [270], [541], and helper data management [102] etc. [46] also found that extracting fixed string from biometric data requires a lot of memory and has to well protect the helper data for security concerns.

Apart from the main blockchain, biometric identity template can be stored in a private side chain to secure the access of the template and preserve the privacy [196]. In accordance with [135], biometric identity template storage is a trade-off between cost and performance, which they conclude that storing biometric identity template in Blockchain Merkle tree is the best performance with lowest cost.

Preserving user's biometric information on Blockchain

Preserving user's private and biometric data through Blockchain generally has three different mainstreams. That is,

1. **Private or permission Blockchain.** Private Blockchain has very few and limited number of controller who takes the responsibility of maintaining the Blockchain system security and consensus rule. A private or permissioned Blockchain suffers similar criticism that is similar as a centralized system, it essentially down to a closed decentralized database [312]. To put more in detail, [249] uses private Hyperledger Blockchain to secure user's biometric identity information by adopting private ledger and IPFS. By similar token, [403] also

adopts similar architectures to secure forensic information on private Hyperledger Blockchain and forensic videos and images off chain in IPFS, which achieving fast transaction speed at 11.99 seconds per transaction. [101] also proposes to use private and permissioned Blockchain to secure visa and digital passport, which a government issued e-passport is scanned into PDF file and then the PDF file is transformed into an URL. The URL will be recorded and pushed onto permissioned Blockchain.

2. **Multiple chains(rings).** Even though multiple chains(rings) in one Blockchain in most cases are used for improve scalability, multi-rings can be used to secure biometric identity information on Blockchain. Particularly, there should be two chains(rings) altogether: one ring is to record immutable URI for the biometric identity template, and the other ring is the private or permissioned Blockchain that is particularly for closed database to securely store user's biometric identity information, such as [384].
3. **Divide user's biometric identity data into n pieces.** That is, dividing strongly encrypted biometric identity template into n equal pieces and nodes in Blockchain only hold one share only. Like [92], it proposes trusted authority server to hold a share of encrypted biometric identity template. That is, [92] separates encrypted biometric identity template into two shares specifically. One share is stored in user's end device and the other share is stored off-Blockchain in DID document, which also compiles Biometric online protocol standard(BOPS) protocol [244]. By similar token, [491] creates multiple different encrypted biometric identity templates in smart contracts and then stored in IPFS. One template is deleted straightaway after being used; however, user is required to store lots of encryption keys as well.

Biometric identity collection and authentication on Blockchain

To collect fresh biometric identity credential from user to conduct biometric identity authentication, data integrity after transmission and security while transferring on open internet are the two main concerns [196]. Since biometric identity data has large inter- and intra-variance, using checksum or digital signatures to check the data integrity is impossible. That is, two biometric identity credentials that are collected at different times are not the same even they are derived from the same person. Therefore, biometric identity integrity is difficult to verify after transmission online. In this case, the alternative solution is to add a clear identity to who the biometric identity credentials are collected from through either public key infrastructure(e.g. x.509 certificates) or digital signatures as in [49] and [135].

Indeed, biometric identity credential collection for biometric Blockchain IdM system has no much differences compared with the rest biometric IdM systems, which is through sensor or cameras mostly. Since Blockchain IdM identity authentication is normally conducted by smart contracts, collected biometric identity credentials arrives at user's digital wallets first or corresponding application clients, and then encrypted before transmitting to Blockchain smart contract for authentication.

The biometric identity authentication in Blockchain is normally conducted by smart contract, which is atomic, self-execute, and irreversible. Most important, the authentication performance is significantly linked with the performance of the biometric blockchain, which major factors that affects Blockchain performance include network latency(bandwidth), block size, and scalability [196]

Last but not least, biometric identity authentication has costs, which is transaction fee that Blockchain system full nodes charge for transaction validation fee and resources fee such as gas in Ethereum.

More literature review about Biometric Blockchain SSI IdM

To add on, below reviews biometric Blockchain based IdM system, which specifically include:

- **Biometric Blockchain.** Survey paper [363] surveys the methods of storing biometric template in Blockchain and compares the cost with performance evaluation. The criteria being used to evaluate the blockchain performance is latency, processing time, economic cost, and biometric performance. They conclude a good trade-off method is to store biometric template on Merkle trees.
- **Smart contracts enabled biometric authentication in Blockchain.** Smart contract as in computing self-execution code which automatically processes its target data once the contract is triggered. The existence of built-in smart contract in Blockchain enables its non-intermediation feature and infer the transaction security from its own intrinsic working mechanism rather than rely on an external protection. [491] discusses the trend of using multi-modal biometric authentication before access a permissioned Blockchain through smart contract based biometric-as-a service system; however, the design is rough and there is not much details and neither has performance evaluation.
- **Biometric in IPFS.** Interplanetary File System (IPFS) is distributed, cryptography content-addressable storage (CAS) network, which enables decentralized data storage. IPFS address vary with its stored contents. [491] use Ethereum smart contract and interplanetary file system (IPFS) to build up a biometric

authentication system and encrypt user biometric data to store in IPFS. Whenever user requests access to online service, user sends the encrypt key and encrypted new biometric data sample to service provider. IPFS address and encrypt key is stored in Ethereum smart contract, which is available to service provider so that user biometric authentication can be remotely fulfilled. There are some established filing database system protocols, which are also generally used in decentralized file storage system. Specifically,

1. Domain name [338]: a string that defines which organization or whom takes control over the domain name specified web resources within the internet. Domain name usually identifies a network domain, such as an internet connected personal computer, or a server computer that is used to host a website.
 2. Domain Name System (DNS) [338]: hierarchical and decentralized naming system for both internet and private network connected devices, services, and resources. Remarkably, DNS transforms a domain name to its corresponding digital IP address for locating and identifying an online device or computer services. Name Server is the server part of DNS, which is to translate the specific human-memorable domain names or hostnames into respective numeric internet protocol addresses, which it is also the first principal of the internet namespace. The second principal of internet namespace is to locate and identify internet computer system and resources.
 3. Directory services [292]: maps the name of network resources to their respective network address. It defines a namespace for the network, and the namespace is used to assign a name(unique identifier) to each of the objects. Shared information infrastructure for directory information base, directory information tree, entry, attribute, distinguish name, relative distinguishing name, directory access protocol.
- **Biometrics as a signature.** Classic Blockchain system suffers private key management issue, as all assets will be lost once the private key is lost. [344] suggests to use biometric data to make a fuzzy biometric signature so that there is no risk at all for the private key to be lost. Particularly, the secret key that is used to sign a signature in the blockchain system is generated by user' biometric data, and will be deleted once being employed. The drawbacks in [344]is the public key file size is ten times larger than default classic blockchain system and the signature generation is around 16 times slower. [265] suggests a practical user biometric data signature scheme in blockchain, where user biometric data is used as a user corresponding private key. The benefit of this proposal is that the private key is recoverable, which solves blockchain private key lose

problem. However, as a compensation of achieving these merits, user's identity is not anonymous any more, and the cost of collecting and authenticating user signature becomes too high to be adopted at a general public scale.

[146] proposes to use biometric data as a password to get confidential access to sensitive information. Specifically, user biometric data like fingerprint data is captured by a CMOS light-sensitive scanner, and then MD5 encryption algorithm is employed to turn global unique features in user fingerprint data into a binary code. The binary code is recorded in blockchain and used as password for authentication. This proposal sounds very promising; however, it does not give clear solution about how to extract global unique features from fingerprint data nor how to authenticate a user specifically.

[287] suggests to store user's biometric template into a one-time token by employing Shamir's secret sharing algorithm [440] and blockchain technology, which aims to secure multi-party blockchain transaction or digital service entails multiple stakeholders. Specifically, the key used to encrypt and decrypt user biometric data in the token is distributed to multi-nodes in accordance with Shamir's threshold secret sharing algorithm in a permissioned or private blockchain network. This method guarantees that there are only sufficient number of blockchain nodes can redeem the decryption key to complete the transaction. However, the drawback of this proposal is it needs extra middleware to issue and verify the token, and the user encrypted biometric token is recorded on the blockchain.

- **Fuzzy biometric signature.** Fuzzy signature is a signature generation mechanism that allows noisy data(e.g. Biometrics data) to be the signature input [60]. As introduced in [332], the generic construction of a fuzzy signature is derived from a combination of both a common signature scheme and homomorphic encryption, which the major drawback of that is the assumption of fuzzy data is uniformly distributed [60]. [265] is also based on biometric fuzzy signatures for blockchain based IoT system, which user's biometric data is used as a private key.
- **ZKP-based biometric blockchain to preserve privacy.** Zero knowledge proof is derived from cryptography technique. Similar with the logic of knowing a Merkle tree root can be equal to claim knowing every single member of that Merkle tree. ZKP proves user's ownership without providing any witness. Classical ZKP algorithms includes Zero-Knowledge Succinct Non-Interactive Arguments of Knowledge (zk-SNARKs) [158], which has established toolbox ZoKrates to run on Ethereum. To de-link from Blockchain public key transaction history, [291] further hashes user's public key into a 'hashclaim', and

identity of the referent is a set of attributes bond to its identifier. By deploying ZoKrates [158], it fulfils a processing model of off-chain computation and on-chain verifying on Ethereum, and accomplish ZKP key exchange in a black-box. Because ‘hashclaim’ does not disclose user’s public key at all and identity is authenticated by zk-SNARK, transaction becomes completely untraceable. Most important, user’s ‘hashclaim’ is maintained by five smart contracts, which registration, identity attribute repository, and knowledge management are all accomplished.

What is more, [234] combines certifying authority(CA) in public key infrastructure with Blockchain to store user identity information like biometrics, civil identity, or digital account identity on the Blockchain. As user public key is certified by an appropriate CA, stored identity information on the Blockchain is signed by corresponding private key so that the validity of the identity information can be authenticated by anyone. The employment of CA leads to a centralized management system, which limits Blockchain transparency feature applications. Whereas as a contrast, [49] adopts a totally decentralized and transparent design which involves a web-of-trust model and smart contracts, which is more practically acceptable in general public scenarios. However, [49] does not suggest an idea on biometric identity but only digital identity.

2.5 E-Passport Systems and Smart Border Control

A country’s border is a geographical physical line which separates two adjacent countries in accordance with it and distinguishes their administrative divisions at the same time. People and products in different countries are subject to different law regulatory; therefore, promoting lawful border crossing indeed is very important, especially in the regard of protecting homeland security, national sovereignty, and nature resources.

Border control is a very complex workflow, but it mainly involves in immigration and customs control for people and goods entering a specific country. Typical border control method can be divided into two major streams: either hard or soft border control. To put in more detail, a hard border control regime is the legal scenario where all people and goods entering a specific country are subject to immigration and customs checking and inspections at the border checkpoints. To avoid land boundary disputes, human trafficking and contraband smuggling, physical boundary like fence or wall are often set up across the border and equipped with heavy military force.

Whereas as a contrast, a soft border regime removes all tangible barriers at the border, which allows people and goods from different countries to move freely without any inspections and checking. A true soft border system requires very strong

cooperation and trust among all countries involved in the soft border system. A good example of that is the European Union(EU). By deeply cooperate among all union countries, EU member citizens and products are free to move within EU zone without any immigration and customs control at all. Under soft border regime, a country's border is only a concept on the paper.

Hard border control regime is beyond doubt to claim the strongest way to prevent illegal border-crossing, but at the same time it is the least efficient. That is, hard border system requires heavy human intervention in nearly all its workflow, and the whole procedure is so time consuming that always causes jam. Plus, for political reasons, there is scenario where hard border regime is completely impossible, such as the Northern Ireland and Republic of Ireland border. Therefore, because of both efficiency and implementation constraints, soft border solution appears to be a strong trending.

To put more in detail, current smart soft border system is to take advantages of smart IoT to replace human border guards to remotely deliver border control task so that the workflow efficiency and accuracy can be enhanced without extra human labour inputs. Especially nowadays Edge and Fog computing technology is getting more established and emerging robust computing algorithm keep coming up, smarter IoT are capable of completing even more rigorous and challenge task. That is, as in the relevant scenario of border control where real time big data analysis in short latent time is required, IoT devices now are able to perform these tasks in a much easier, securer, and more accurate manner.

The concept of smart border control system indeed starts in 2013 [118] when EU zone faces increasing number of illegal border crossing. The Schengen area is an established soft border zone where member countries' internal borders are all abolished. It booms EU zone economy but somehow becomes a cost of security as Non-EU country citizens and criminals are very easy to get across the EU border without any legitimacy at all. Plus, when the number of both legal and illegal immigrants goes up very quickly, the number of border guards have to go up correspondingly to ease the congestion around the border. To tackle these issues, European Commission starts to build up smart border control systems to speed up passenger clearance to enhance border control.

2.5.1 Smart Border for Immigrant Control

Existing smart border system for *immigrant control* can be roughly summarized as a smart electronic-gate(e-gate) system, which has already been broadly deployed in many airports, such as Norway Avinor Oslo Airport, Germany Berlin Tegel Airport, Ireland Dublin Airport and UK London Heathrow Airport etc. As seen in **Figure 2.5**, the e-gate system usually consists of a swinging door to give access control, a screen

device to capture passenger photography, a passport reader to collect travel document information, and a fingerprint reader to collect passenger fingerprint in real time [555]. Specifically, the gate facilitation aim to collect passenger biometric finger-print and photography data while passenger walks through the channel, and automatically authenticate passenger biometric identity and border crossing legitimacy.



Figure 2.5: Border e-gate facilitation in London Heathrow airport: swinging door, screen device, passport reader, and biometric identity reader [555]

Indeed, there is one necessary premise for deploying e-gate system, which is the application of machine readable biometric electronic passports(e-passports). Malaysia in 1998 was the first country which issues biometric passports, but now there are more than 85 countries issue it [172]. To put more in detail, the major difference between e-passport and normal passport is e-passport has a micro-chip that contains the passport holder's encrypted fingerprint, photography image and soft identity data, but normal passport can not be read directly by a machine at all but only a human person.

From the technical side, the current e-passport is a combination of RFID, biometric identity, and public key infrastructure technology [288], [383]. That is, passport book carries RFID chip that contains the biometric identity information of the passport holder with only basic access control. That is to say, in case of RFID chip forgery, only simple encryption scheme is introduced [389]. Most important, current e-passport still depends on physical counter-forgery applications such as laser perforated number, UV reaction passport pages and hologram patches etc to protect the e-passport from data copy [383].

The state of art generation of e-passport is dominated by fully digital non-tangible formatted passports for more flexibility and efficiency, which is still not yet

in production but under proposing and evaluating. For e-passports, [500] suggests to put scanned passport as PDF file and then push it onto Blockchain after further encryption. The new e-passport can be used through mobile phone generated QR code. The benefit of that is fast retrievable of the original passport data; however, the drawback of that is it cannot be used for online identity authentication. Similarly, [500] indeed is a digital non-tangible transformation solution for a tangible passport book. [101] uses permissioned Blockchain to digital passport and visa to make identity authentication at passport and visa office particularly, which is very similar with [500]. However, it does not mention biometric identity is authenticated by the passport office and the passport is issued to user directly by passport office. [40] aims to improve adaptability and interoperability of the e-passport among different countries, which is to use more advanced data-carrier in the passport book such as electronic ink. [369] uses permissioned Blockchain to construct an international e-passport system for all countries around the world; however, it gives a clear communication solution about how two countries exchange and authentication passport information but not give any clear information about how the e-passport is constructed nor any architecture design is available.

2.5.2 Smart Border for Customs Control

As for smart border system in *customs control*, Norway–Sweden 1600 kilometers high-tech land border is the one that should not be forgot to mention [282]. Both Norway and Sweden are within Schengen area but Norway is not an EU member. Therefore, people are free to travel across the border but goods and products are subject to customs inspections and checking. To enhance border custom control and improve efficiency over cargo and lorry, Norway-Sweden smart border uses x-ray to scan lorry so that there is no need for unloading products. Plus, an automatic number plate recognition system is applied all around the border. The system can automatically recognize all pass-through vehicle plate number so that the vehicle pass-through legitimacy can be automatically verified at the real time.

Smart border customs control indeed is an intelligent transportation system(ITS), which aims for speeding up both commercial and private vehicle border crossing at border checkpoints [276]. It requires efficient data collection and exchange among vehicle, driver, and carrier(customs clearance, licensing, and registration etc.) [276]. ITS improves border crossing efficiency by remote monitor and intelligent supervise over vehicle transportation status and sending feedback and updates to corresponding entity in case of delay and rejection risk. Like for instance, the UK Single Trade Window(STW) as part of smart border system is built to share shipment data with multiple governmental organizations to reduce the necessity of data duplicate entry [103].

Main challenges

To put it clear, the main task in smart border customs control is closely related with (1)remote identity(driver and vehicle) authentication over open internet and (2) different platforms issued data(documentation) authentication and then reconciliation it with real time border checkpoint in-situ live data.

To put more in detail, [302] points out that the key challenge in cross border smart customs control is the reliability of product loaded on the commercial vehicle, which normally requires enterprises to submit full documentations and evidences that are issued by different organizations to verify the product information. Similarly, [24] argues that the main challenge is the interoperability among different organizations within the customs control system. They both suggest Blockchain can be a good solution; However, none of them offer any specific Blockchain solution. Plus, [266] also claims that documentation is the 'backbone' of the customs procedure.

The State of Art of the Smart Customs Control Solution

With known main challenges in smart border customs control, some state of art smart solutions are proposed. To emphasize, customs workload does not only process data to reconcile it but also takes the responsibility to collect and monitor it. To put more in detail,

- **Smart documentation.** In border customs clearance scenario, it has been discussed that a lot of different documentations are involved ,such as import/export permit, product origin certificate, special licensing related with the product such as high precision high tech research equipment's, and product safety certificates etc. They all issued from different organizations; therefore, authenticating the documents becomes part of border officer's workload. In smart border solution, UK single trade window [103] has been proposed to tackle duplicate input issue, which let user input in-use data once only by share the STW with different organizations and allow them to take the data they required from the STW. Similarly, [439] proposes a data sharing solution that is Blockchain built on top of global Cloud. Global Cloud receives data access requests and then forward the request to the security gateway where the request belongs to, finally the requested data will be sent as a Blockchain transaction with sender's signature.
- **Smart transportation.** Smart transportation for capture vehicle knowledge remotely through computing is widely used in smart border system, such as [282] uses x-ray to scan loaded products on vehicle so that there is no need to

unload product at the border checkpoints any more. Similarly, gamma-ray/X-ray scanning is also adopted in [472] to eliminate security risk of products loaded on vehicle. Plus, capturing vehicle number plate through CCTV camera is a division of computer vision, like [348] use deep learning algorithms to improve the computer vision to detect vehicle number plate and doubtful objects. Most important, RFID customs seal reader is also applied to customs control to inspect vehicle borer crossing legitimacy.

- **Smart remote surveillance.** FOLDOUT [382] smart sensor platform is particularly designed for border crossing surveillance, which a high resolution camera, a thermal camera, and a low-light camera are introduced into the system to enable 24/7 non-stop workflow. It is highly weather-resistant and the highest sensor range can reach 200 meters. By similar token, [3] uses IoT devices set up a border protection system, which the IoT are equipped with RFID tags to alarm intrusion with graph based model.

Some currently still under researching smart customs control solutions are also worth mentioning. That is, one project in relevance is the TradeLens developed by Maersk and IBM [4]. By logging in commercial invoice and packing lists in TradeLens, all customs clearance required documentations will be submitted by professional customs broker. The platform is based on permissioned Blockchain for permission-based data sharing. For another example, Samsung and Kcnet [268] also builds up the customs logistic platform on Blockchain, which is particularly for exporting service. It is built upon private Blockchain with auto information consolidation and sharing. Similarly, Microsoft and inter-American development bank (IADB) [425] also work on a cross border customs clearance platform based on Blockchain, which aims to enable secure, efficient and immediate data sharing.

2.6 Chapter Summary

In this chapter, background and related work are introduced and reviewed. E-passport as biometric identity credential, which is closed linked with **Chapter 2.1 Biometric IdM**. Traditional biometric IdM system are dominated by centralized or federated mechanism, which suffer significant security and privacy risks, such as single point of attack, system break down, data breach, and administrator function creep etc. Plus, biometric identity data itself has very distinguishing features like publicly collectable, high inter- and intra-variance, and permanency, preserving biometric identity information privacy and protect system security is an indispensable and crucial part of all biometric IdM system. Therefore, in subsection **Chapter 2.1.1 Preserving Biometric IdM privacy** and **Chapter 2.1.2 Protect Biometric**

IdM Security are reviewed. To preserve biometric IdM privacy, official regulations has to be fulfilled and introduced, which is in **Chapter 2.1.1.1**. After that, specific biometric dtaa privacy preserving methods are introduced and reviewed in **Chapter 2.1.1.2**.

To the relevancy of our research topic, **Chapter 2.2 Existing Established IdM Authentication Manager** is reviewed, which include **Chapter 2.2.1 Fast Identity Online(FIDO)** and **Chapter 2.2.2 Public Key Infrastructure(PKI)**. That is, FIDO is the state of art of online biometric identity authentication protocol, and PKI is the state of art of (centralized) public key oriented digital identity system. Comparing with our research of decentralized identifier(public key) and digital biometric credential(biometric e-passport), FIDO and PKI are of strong relevancy.

Followed by a review of **Chapter 2.3 self Sovereign Identity(SSi) and SSI IdM** that contains SSI features and current state of art SSI IdM like Sovrin, ShoCard, and Uport, **Chapter 2.4 Biometric Blockchain based IdM** is also reviewed which is biometric Blockchain based SSI IdM.

At the end of the chapter, **Chapter 2.5 Current e-Passport Systems and Soft Border Control System** is reviewed.

Chapter 3

Identity and Identifier in the Metaverse

3.1 Digital Twin in the Metaverse

The Metaverse has been defined as “the concept of a fully immersive virtual world where people gather to socialize, play and work [290].” It transforms existing 2D computing user interface into 3D, and imitates the real world into virtual simulations for users to make immersive interactions. To add on, the Facebook further describes the Metaverse as a place where you can present yourself in a 3D digital space where you are inside of [290]. That is, Metaverse is blurring the edge between real and digital world.

To put more details in about the Metaverse, the word Metaverse is origin from Neal Stevenson’s science fiction novel, Snow Crash, in 1992. Recently, Facebook changed its name to Meta in 2021 draws exclusive attention and pushes the concept of Metaverse famously known. Specifically, Metaverse is the new evolution of computing technology in terms of offering an immersive user experiences through cross reality(XR), which XR includes virtual reality(VR), augmented reality(AR), and mixed reality(MR) [343]. This revolution transforms the traditional computing 2D image based user interface into 3D, which conquers the major limitations and inefficiencies in 2D environment. For example, [343] has argued that the maim limitations in 2D education is the high drop-out rate because of Zoom fatigue and emotional isolation. It draws a call on the necessary of 3D user interface, which the advantages include immersive interaction and optical fidelity of the reality thanks to XR [410].

As discussed above, Metaverse has a focus on immersive interaction over open internet. Indeed, does not only the application of 3D user interface but also the fidelity and veracity mapping from the real world to the digital Metaverse world, they both are able to bring user immersive experiences. Specifically, the real fidelity

mapping is called digital twin[178], which normally the digital twin is constructed through Blockchain technology in the Metaverse[236]. The reason of that are mainly because Blockchain is able to construct an immutable tie between a real world object to its projection in the Metaverse digital world [236].

For a formal definition of digital twin, it is a digital representation of an actual real world object that serves as the indistinguishable digital counterpart of it for the purpose of simulation, testing, and monitoring etc [296]. That is to say, the digital twin constructs the mapping from real world object to the Metaverse with high fidelity and consciousness, which requires existing attributes of the real world object to be mapped to the Metaverse as much as possible[517]. what is more, through enabling this mapping, digital twin also enables first-hand data collections from digital twin devices so that artificial intelligence and computing assisted management, predictions and simulations can be enabled as well [178].

In relevant of our research, Metaverse application in logistics and individual identity in the Metaverse are of interests. To put more in detail, digital twin based logistics and transportation system has significant impact on the visibility of a vehicle, which enables machine learning and other computing methods applicable to the logistics system overall [341]. It requires surveillance system such as sensors, actuators, and CCTV cameras etc to collect live data from the real world and then constantly sends to the digital Metaverse to reflect the current status of the object [341].

3.2 Identity in the Metaverse

Metaverse is a fully digitized world where its existence is completely depends on computing technology. It combines the concept of digital twin, computing virtual reality(VR), and augmented reality(AR), which constructs a digital world for entertainment and efficient management of the real world. Indeed, digital twin is a very crucial part in Metaverse, which is to construct a real mapping between the real world and the digital world by digitizing the real world entity. To put more in detail, digitizing a real world entity is to use computing technology to symbolize the real entity's attributes and identity in digital world, which can be summed up as creating digital identity for the real world's tangible object and use it in Metaverse.

Indeed, identity in Metaverse can be completely made-up, like identity in Metaverse gaming or media channels etc. However, digital twin identity in Metaverse creates more value, which enables personal attributes and digital belongs to be attached to it. To put more in detail, digital twin identity in Metaverse emphasize the real mapping and veracity, which has duplication effects of knowing of one world status is equal to knowing of the other. Most important, there are a lot of limitations of learning real world without computing. Digital twin in Metaverse is able to conquer these limitations and enable computing applicable to the real world [516].

In this section, the concept and definition of both identity and identifier are introduced in detail and how to separate identity into different categories in accordance with identity credential format and identity nature is explained. That is, in accordance with identity credential format, identity can be classified into paper-based identity, digitized identity, and digital identity [436]. However, in accordance with the nature of identity, identity can be separated into intrinsic identity like biometric identity and linked or linkable identity like gym member identity [213]. Most important, we discuss the stability and persistence in an identity significantly depends on how strong the tie is between the identity identifier with that person. As biometric identity identifier come from human body measurement that is readable, measurable, and intrinsic to a person, we claim biometric identity system has the highest assurance and credibility in the regard of identity identification and authentication [7].

3.2.1 Identity definition and identity category

In current era, securely authenticate a person's identity is brutal in security. By definition, a person's identity is 'who a person is, or the qualities of a person or a group that make them different from others [93].' In accordance with that definition, two major functions can be perceived in identity. First, differentiation. A person or a group of people can be differentiated through identifying or calling those distinguishing qualities in identity [84]. To put in more detail, in modern society, resources are very limited like welfare benefits and healthcare. To keep resources distribution fair and equal, effectively differentiating every individual is crucial so that everyone can get an equal share of the public resource. For another example, criminals receive penalty and punishment upon their wrong doings, and one person can only vote once in general election. Precisely distinguishing a person from anyone else is also beneficial for maintaining a safe and secure social order if one person is completely prevented from claiming multi-identity in one identity system. Second, consistency [297]. In sociology, the part of 'who a person is' in identity definition is defined by 'how we see ourselves' [84]. That is, a person's self-identity, social identity and role identity, like social position in community, personal character, beliefs, values, and role in family etc. these identities form a person's mindset and spirit which has significant and consistent effect on a person's behaviour. Therefore, the consistency in a person's identity refers to a psychologically consistent self-identified image of how a person see himself, which is not used for differentiation but self-identification [408].

Normally, one identity consists of two parts: identity identifier and its corresponding credential [94]. Specifically, identity identifier is the specific syntax of we are who we claim to be, and credentials are the evidence or mechanism that we use to prove we are the owner of that identity. Therefore, in the process of identity authentication that is to prove a person is who they claim to be, it normally involves two procedures [294].

First, declare the identity as they claims who they are. Second, prove the ownership of that identity declaration. Therefore, the assurance of identity authentication is judged by how strong of the identity ownership is approved [16].

As both the identity identification criteria and the distinguishing precision request vary a lot in different scenarios, the attribute of identity comes into different categories too. However, identity in general can still be separated into two major categories: intrinsic identity and linked or linkable identity [94].

To put in more detail, identity that is intrinsic to a person refers to a person's biometric identity, genetic identity, and personal identity, which is the identity completely belongs to and self-sovereign by identity user [319]. That is, intrinsic identity is an intrinsic character that a person born with, developed and fully controlled by that identity user. Some intrinsic identity is also called legal identity, which the information declared in legal identity can precisely identify one single person only with legal enforcement function and legal ground [7]. To be more precise,

- **Biometric identity** [210]. Biometric identity comes from body measurements that can be calculated and extracted from a human body, which the feature of a person's biometric character remains unchanged during that person's lifetime. Most important, the undeniable and immutable tie between biometric data with one and the only person makes biometric identity distinctively special and highly assured, especially in the scenario where authentication of a person's identity has legal concern. Plus, in accordance with [50], there are no two people by far have been found had identical fingerprint yet. Therefore, one outstanding benefit of using biometric identity to authenticate a person's identity is that it has legal enforcement function, which is biometric authentication of a person's identity can enforce that person's behaviour to be governed at jurisdiction and law legitimacy level [251]. One classical example of biometric identity application is police officer collects biometric identity information like fingerprint at crime site to identify a criminal [17].

For another example, personally identifiable information (PII) like photography, fingerprint, voice, iris, palm print, Deoxyribonucleic acid (DNA), gender, and place of born etc, they all belong to intrinsic identities [251]. Indeed, human biometric character has been classified into two types, and specifically they are physiological and behavioural biometrics [83]. That is, physiological biometrics are like fingerprint, photography, iris, palm, and voice etc. Those biometric identity characters come from our own body, which is measurable, readable, globally unique and immutable. Whereas as a contrast, behavioural biometrics is the feature of our body movement or gesture, like the way how we walk or how we take mobile phone out of our pocket etc [447]. They are established patterns of our behaviour which are formed by our behavioural habits. The authentication of behavioural biometrics identity requires

to collect a larger size of sample data to extract identity feature, but the major drawbacks of behavioural biometric identity feature is they are not stable and very easy to duplicate; therefore, in most cases behavioural biometrics identity will be used as an add-on to strength a multi-factor biometric authentication system [50].

Apart from biometric identity, personal identity is an intrinsic identity to a person as well, but the major difference is biometric identity is eternally stable, globally unique, and immutable [210]. However, as a comparison, it is generally acceptable to claim that a person's personal identity character will change over time and personal identity is very hard to quantify and measure. Like for instance,

- **Personal identity.** Personal identity refers to beliefs, values, habits, and personal characters etc. which is more about a person's self-image and mindset [84]. Personal identity character like extrovert is an identity, which distinguishes extrovert people from introvert people. In that case, the distinguishing character 'extrovert' represents a group of people, which identifies people in a relaxed precision manner. An identity like that can only differentiate people in a vague manner, which is more appropriate to separate a group of people. As a result of that, even though personal identity is an identity, it is rarely used in real practical scenario to authenticate individual's identity. That is, even though personal identity is an identity, it is commonly used as a descriptor but not an identity identifier [408].

As a comparison to intrinsic identity, linked or linkable identity is an identity that is issued by an identity issuer, and then the identity issuer links that identity with one specific person to fulfil a function [294]. Indeed, the aim of issuing linked or linkable identity is the identity issuer attempts to distinguish one person from anyone else for issuer's own purpose [294]. In most cases, the identity issuers are either governmental organizations or service providers, and they issue an identity to all general public or their service users to fulfil either cohesive governmental functions or business functions. Like intrinsic identity, the linked or linkable identity also can be divided into two major sub-classes, that is, linked or linkable legal identity, and linked or linkable functional identity [37], [544]:

- **Linked or linkable legal identity.** Linked or linkable legal identity is the identity itself has a legal ground, which is generally used to cohesively fulfil a governmental function that is applied to all general public, like taxation, border control, and vehicle control etc. Therefore, the issuer of legal identity in most cases are governmental department like Driving and Vehicle Licensing Authority (DVLA), Her Majesty's Revenue and Customs (HMRC), and UK Border Agency(UKBA) etc [297]. For one specific example, government issues birth certificate to new born baby for vital record the birth of a person so that government can tell if that person is legitimate for governmental benefits. Most

important, birth certificate contains gender, place of birth, and date of birth intrinsic identity information as well, which strengthen the legal ground of birth certificate. To point it out, biometric identity belongs to legal identity, but it is not linked by any identity issuer nor linkable to any individual person, but it is intrinsic to one specific individual who born with these biometric identities characters [544].

Similarly,

- **Linked or linkable functional identity.** Linked or linkable functional identity is service provider issues an identity to fulfil a function in accordance with their business service provider’s requirement [23]. One good example of linked or linkable identity is digital identity, which is issued and managed by online service providers, like email address, IP address, online shopping account, Instagram account, Facebook account and digital newspaper account etc, these identities are linked with one user and gradually become an indispensable part of our daily life. For another example of linked identity in real tangible world, that is gym issues gym card to all users who have paid for gym service [23].

Most important, the obvious feature of linked or linkable identity is the identity itself totally controlled by the identity issuer, so issuers can revoke and modify user’s identity as much as they want. See below **Figure 3.1** for a demonstration of identity category structure.

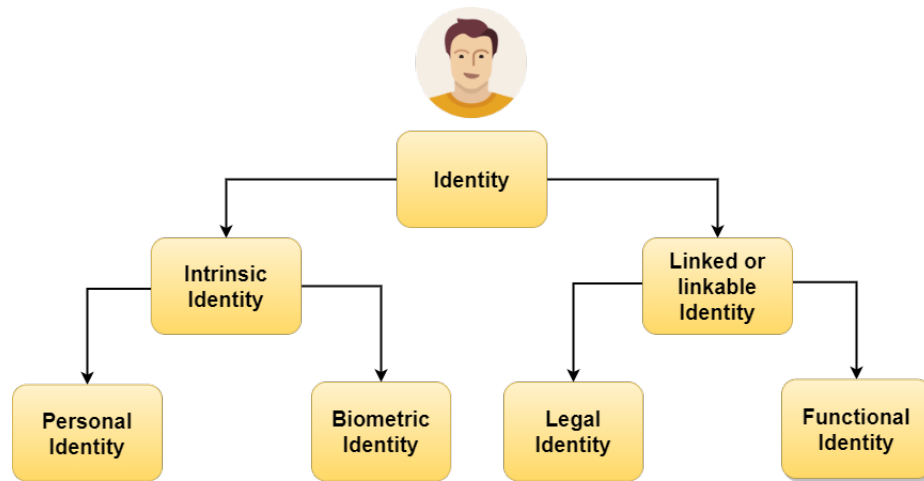


Figure 3.1: A demonstration of identity category structure.

In accordance with the format of identity credential, identity has three major formats. That is, paper-based identity, digitized identity, and digital identity [94].

- **Paper-based identity.** In general, paper-based identity refers to the identity that we use to interact in the tangible world, which the identity issuer issues paper-based identity credentials like residential card, passport book, gym card, birth certificate, and bank card etc to prove the ownership of a paper-based identity. Paper-based identity is issued to get access the resources and services that are supplied in the real tangible world, which the paper-based credentials enable face-to-face identity authentication.
- **Digitized identity.** Digitized identity is to transform a paper-based identity and its corresponding credential into a digital documentation like a scanned copy or photography so that user's paper-based identity can be authenticated online and remotely. In most cases, digitized identity is requested by online service provider to prevent their service user from claiming multi-identities in their service identity system.
- **Digital identity.** Apart from the identity that we have or are linked with real tangible world, a new form of identity is imposed with the arrival of internet. That is, the digital identity that we use in digital world. Digital identity is the identity that we use to interact over internet, which allows the access to computer and online services to be mediate and managed by computer programs rather a real human being. To be more precise, a digital identity can be interpreted as a bundle of digital data, which contain the identity attribute data so that identity authentication can be conducted, and user interactions can be controlled by verifying those attributes.

Most important, one outstanding feature of digital identity is anonymous [94]. That is, user name in most cases are pseudonyms rather than user legal name, like email address, online shopping account user names, and gaming user names etc. As a consequence of that, online service providers are ignorant about most of their service users' legal identity at all.

Anonymity in identity indeed can be an advantage if the service provider does not require any legal enforcement or the digital service provider only requests weak identity proof [16]. That is, user's privacy can be securely protected by naming all user anonymously. However, at the same time, anonymity may cause sever security risk like one person can easily claim multi-identity in one identity system. Therefore, service provider usually will request their service user's identity to be matched with their service security level [289]. For instance, in the case of the online service has legal concern like money transfer in banking, border crossing application to border agency, or access to company intranet for confidential business data etc. user digital identity in those cases will be linked with user legal identity to improve identity credibility, and make user digital identity more reliable and trustworthy. That is to say, identity has credibility [309].

Another thing can never be forgotten to mention about identity is identity credibility, which refers to how trustworthy it is about an identity claim, like our Facebook account identity does not have as much credibility as our legal name identity as in residential card [309]. The higher credibility an identity has, the more acceptable for an identity to be adopted by service providers. To put in more detail, identity credibility significantly depends on how strong the tie is between a person and an identity identifier [275]. Specifically, as biometric identity identifier is intrinsic to a person, the tie between biometric identity and a person is eternally stable and immutable; therefore, biometric identity has the highest credibility among all the rest identities [83]. Followed by legal identity, the tie between the legal identity identifier like legal name and the person is cohesively composed by governmental organization with legal ground [251]. Therefore, whenever service provider requires higher level identity credibility to secure their identity system, service provider will compulsory link user legal identity or biometric identity with their service user's account identity.

For a short summary, identity is who we are and what make us different from anyone else. Biometric identity is intrinsic to one globally unique person only, which the identity is extracted or measured from human body so biometric identity has legal enforcement function. Legal identity refers to identity that is issued and cohesively linked by a governmental organization to fulfil a governmental function, and linked functional identity is issued and linked by general business service provider to fulfil their business function. Most important, the reason why a person's biometric identity has distinctive legal enforcement function is because the strong undeniable and immutable tie between biometric data to one and only specific nature person.

3.2.2 Biometric identity

By definition, biometrics is ‘...the measurement and statistical analysis of people's biological and behavioural characteristics, which can be used to recognize or identify individuals [210]’. As discussion above, biometric identity belongs to the category of a person's intrinsic identity, which comes from human physical characteristics like body measurements and materials. The intrinsic feature of biometric identity is derived from the immutable and undeniable tie between the extracted biometric data and one and only person at a global scope [127]. The origin of biometric identity application can be tracked back to the 17th century in Paris when anthropometry was initially used in forensics science to identify criminals based on evidence collected at crime site. Due to the fact that biometric data has to be extracted from human body, the tool and technology can be used to extract biometric data from human body significantly affect the generalization and deployment of biometric identity applications. From measuring tape, inked fingerprint on paper, to today's high precision sensor and laboratory equipment, more and more biometric information can be extracted from human body

[250]. Plus, computing algorithm is getting more and more robust to extract features from big raw biometric datasets, which also helps the generalization of biometric identity application. Like nowadays, electroencephalographic brain activity can also be used to identify a person [366].

Indeed, there is no such a formal definition or concept about what a novel biometric trait is; however, to count for a biometric identity feature to be able to function as an identifier to distinguish a person from anyone else, then certain criteria need to be fulfilled [250]. *First, universality.* The biometric identity traits to be extracted have to be universality available to extract from all target population. *Second, uniqueness.* Biometric identity feature of two different people has to be different as well. *Third, stable.* A good identity feature is persistent over time, and one person can only occupy one identity only in one biometric system [163], [450].

In accordance with the origin of biometric raw data, biometric identity can be further classified into another three groups [210]. That is, (1) genetic identity. (2) phenotypic biometric identity. (3) behavioural identity. Genetic identity raw biometric data are collected at molecular level, which in general is used in forensics science, species classification, and kinship identification. Physiological phenotypic and behavioural identity in general refer to the identity raw data are collected and derived from human physiological body. Specifically, phenotypic identity is a measurement or character of our physiological body, whereas behavioural identity is a stable and established pattern of our physiological body interact with outside environment [210]. To put in more detail,

1. **Genetic identity** like DNA and blood type can identify a person at molecular level, which a person is identified by human physiological body material as what a person's body is consists of. DNA can be extracted from a tiny amount of blood, hair, bones, teeth, saliva, or semen, but the equipment and skill required to extract DNA is complex and very high [161]. To put in more detail, organic polymer DNA indeed exists in every human organism cell, which is consist of the phosphate backbone, the deoxyribose sugar, and the nitrogenous base [17]. Within one cell, there are approximately three billion bases long nucleotides string. Therefore, even though 99 percent of human DNA is exactly the same among different people, one percent variance is enough to precisely distinguish a person from anyone else at a global scope. For instance, traditional short tandem repeats analysed by capillary electrophoresis is not able to distinguishing Monozygotic twins at DNA level in the past; however, very recent the discovery of ultra-deep massively parallel sequencing method can fulfil the task of classify identical twins by DNA [494].

Plus, because the process of genetic identity identification is very time consuming and it is also very difficult to extract raw genetic data from human

body, genetic identity is generally applied to forensics science and parenting identification but not general practice applications. Most importantly, genetic identity has a significant security concern which is more than a loss of assets or data. [319] discusses genetic identity security concern at the regard of human reproduction at law level. As reproductive technology like clone and genomes extraction technology get more and more established, it brings a significant risk for new born baby to inherit genes substantially different from common human reproduction forms and endanger human species as a whole.

2. **Behavioural identity** refers to established movements or habits when a person's body interacts with environment, such as gait, keystroke, and signature [428]. Due to the fact that human behaviour varies a lot in different environmental conditions, it is also imperative to decide what to extract from human behaviour to count as a stable identity feature [163]. [380] has investigated in behavioural biometrics in VR, body segments are arbitrarily combined into body relation, which concludes the pattern is useful to identify users. To put in more detail about behavioural identity,

- **Signature.** At the very beginning, signature can only be authenticated by how much two signatures are alike to each other in the regard of appearance only. That is, Ascertain similarities between biometric raw data sample and template. However, as now computing algorithms and high precision camera can capture more information from a signature. Signature identity authentication through both 2-Dimension and 3-Dimension are made possible. That is, signatures are conventionally used in the case of signing a business contract to secure a transaction, cashing a cheque, or reaching agreement between multi-party. Therefore, it is very important to tell the difference between an authenticate signature and a forged one. In the case of a skilled forgery, it is extremely difficult to tell the differences just by human eyes. Even though a signature sample will be collected before verification, it requires a lot more sample if more stable features to be extracted in a signature. For example, thickness of a stroke, pressure, and the speed of pen during signing can be taken as a stable feature for authenticating a signature [246]. Deep learning algorithms like Restricted Boltzmann Machine (RBM) [2], embedded high-dimension Euclidean distance [397], and Siamese recurrent neural networks [488] are applied to extract signature features with significantly high accuracy. In accordance with [248], the lowest equal error rate of signature identity verification can be as low as 2.63 percent.
- **Gait.** Gait identity is a typical pattern recognition issue which draws a very wide range of community's attention like computer vision, machine

learning, biomedical, forensics science and robotics. Gait identity feature can be extracted either by model-based method or appearance-based method [50], [447]. That is, model-based method is to learn the pattern of gait in the regard of a person's whole body, which is how legs moves in response to the rest of our body. Whereas as a contrast, appearance-based model refers to leg movement pattern 'in the image'[45]. Very recently, some new emerging deep learning algorithms have been applied to gait identity learning which shows very good results. For example, 3D convolutional neural networks can learn a gait patter from multiple viewing angles [526]. [50] uses recursive long short-term memory (LSTM) algorithm to learn skeletons joint features.

There are some major pros and cons in deploying behavioural identity to authenticate a person's identity [112], [358], [529]. Specifically,

- **Pros.** Behavioural identity in general is publicly collectable; therefore, a person's behavioural identity can be verified even without the notice of the subject [358]. That is, behavioural identity verification is a good extra factor to further assurance the identity of the user, especially when knowledge-based credentials are stolen or surveillance is vitally required. One classical behavioural identity application is to use keystroke to secure mobile access. In that case, even knowledge-based PIN is lost, further keystroke identity verification can terminate illegal access. Plus, some behavioural identities are extremely hard to spoof. Again, like keystroke identity, raw data has to be professionally captured before an identity feature can be extract, which make the identity impossible to spoof.
- **Cons.**
 1. Time consuming. Due to the fact that behavioural identity feature is impossible to extract from only one instance of movement, it requires a massive amount of raw data collection and computation before a behavioural identity feature can be extracted and fixed from a person [229].
 2. Behavioural identity is publicly available, which makes it impossible for deploying behavioural identity as one single factor to authenticate a person's identity. Like for instance, a person's gait can be captured by video cameras without any requirement from that person's cooperation [289], [498].
 3. There are too much external factors affecting the accuracy of behavioural identity identification and authentication. That is, gait pattern is significantly affected by leg injuries, walking speed, clothing, and weather

conditions etc. Similarly, signature may have a major change when pen and paper-material is changed [50].

Apart from above biometric identities that are being discussed, phenotypic biometric identity like fingerprint, iris, palmprint, ear shape, facial appearance, voice, and retina also belong to biometric identity. In the very past, face and fingerprint can only be verified by man-power with bare human eyes, which has a big chance to get a false verification result. With the arrival of information digitalization and deep learning applications, phenotypic biometric identity authentication also steps into a new era. Replacing ink on paper, phenotypic biometric identity is digitalized into raw data, which makes biometric identity traits way easier and more accurate to be extracted from raw data. Indeed, the uniqueness of phenotypic biometric identity has only been statistically proved, and the research is still not yet established [7]. According to [7], U. S. National Institute of Standards and Technology (NIST) reported 1 false match in 40 billion iris comparisons; therefore, the uniqueness of iris identity is statistically admissible. Even though there are some court cases doubt the uniqueness of biometric identity, fingerprint and face identity are generally admissible with acknowledged legal enforcement.

In a vague manner, biometric identity also can be divided into *soft biometrics* and *hard biometrics*. That is, soft biometric identity [129], [354] are identity features that can be semantically and precisely expressed by human, such as gender, age, ethical group, and bearded etc. Hard biometric identity is like fingerprint, DNA, and voice etc, which is impossible to express by words but is able to precisely distinguish one person from another. To put in more detail, soft identities are commonly used in identity authentication system to fusion with hard biometrics to make a quicker and more accurate decision about the subject's identity, especially when biometrics data are captured under un-constraint environment [200]. Even though soft identity on its own can also distinguish a person's identity, such as bags of soft biometrics, the accuracy compared with hard biometrics is much lower [355], [414]. Deep learning methods are also good at finding correlations between each factor [468].

Recently, new form of identity keeps coming up with the development of new data collection equipment and data analysing knowledge, such as finger vein identity [245], [540], ear [519], and lip movement [529] etc. [398] deploys cancellable iris recognition to secure iris identity authentication system, which reaches a 99.9% recognition rate and 0.97 second processing time. Plus, revolutionary new data collection equipment drastically improves the biometric identity authentication accuracy as well. For example, remote identity verification by gait and face [447] with a high detection accuracy of 95.2%. Super resolution [21], [30] images significantly improve performance in face image recognition and fingerprint especially when image collection environment is poor. Plus, industrial biometric identity authentication mechanism is getting into the market as well. HSBC mobile banking application offers fingerprint

or voice biometric identity verification to secure the access to their application to their 15 million customers [7].

Fingerprint

A fingerprint raw data is commonly collected by a sensor, scanner or camera to capture an image of an impression that is left by the friction ridges of the subject's finger. A fingerprint consists of ridges and valleys on the surfaces of the finger, which is a statistically and globally unique human physiology feature and persistently forms the individuality of a specific person. With the arrival of digitalization and specialized devices, fingerprint local minutiae can be captured and extracted precisely with more subtle details [21]. Plus, with computing deep learning algorithm becomes more and more established, fingerprint identity authentication and feature extraction also come into a new era.

The reason of why fingerprint identity authentication is popular are three folds [193]. Due to the generalization of smart phones, the acquisition of fingerprint data becomes more convenient and cost efficient. That is, one of the benefits of fingerprint identity authentication is for its compactness, accuracy and price effectiveness [77]. To put in more detail, there is a very good generalization and acceptance of fingerprint identity authentication system among all the general public is mainly because of three reasons.

1. Fingerprint compared with other physiology identity features has a good privacy. That is, the access of fingerprint data can be fully controlled by a person. Without the cooperation of the subject, fingerprint data is difficult to retrieve.
2. Efficient. In the regard of both fingerprint identity verification and authentication and cost, fingerprint data always offer better user experience due to the established fingerprint recognition algorithms and the generalization of fingerprint sensor in smart mobile.
3. Even though it is proved that fingerprint does have age and aging problem [154], but in general it remains constant over a person's lifetime and it is very difficult to alter. That is to say, fingerprint is very stable and persistent.

To put in more detail, depends on the resolution of fingerprint image, fingerprint feature can be extracted either globally(loop, delta, and whorl) or locally(minutiae) [333]. See **Figure 3.2** for a demonstration of fingerprint level 1, 2, and 3 features [20]. That is, level 1 features are fingerprint global features, and both level 2 and level 3 features are local minutiae features. To extract higher level feature, normally higher resolution is required. Therefore, a global fingerprint identity feature is defined by the patterns and flows of ridges and valleys with singularities such as loop, whorl

and arch. Whereas as a contrast, a local fingerprint feature shows more distinctive details with minutiae (Galton details), and even more small geometric details such as scars and pores with minimum acquisition resolution of 1000 dpi [495]. For a specific example, 20-40 sweat pores are proved to be sufficient enough to recognize a person [25], [495].

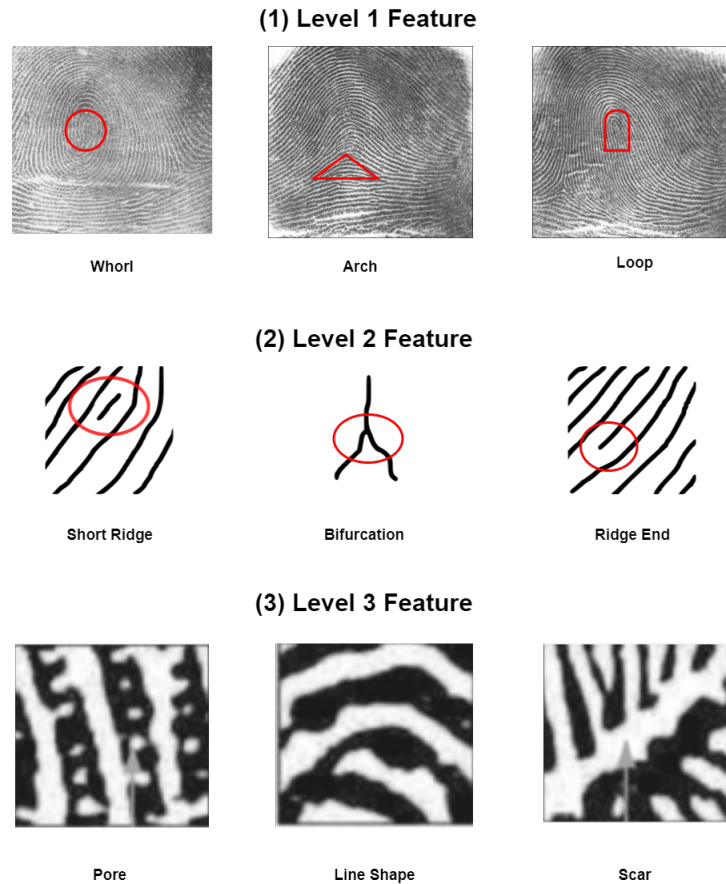


Figure 3.2: A demonstration of fingerprint level 1, 2, and 3 features [20].

In accordance with [285], [539], a general framework of extracting a person's fingerprint identity can be summarized as 1. Fingerprint image acquisition. 2. Image pre-processing. 3. Fingerprint feature extraction. 4. Fingerprint template generation.

- **Fingerprint image acquisition** - Fingerprint image acquisition now heavily depends on specialized devices, and differences of deploying different fingerprint raw data acquisition devices include image resolution (number of pixels per inch), area (size of fingerprint image collection window), and number of pixels(

number of pixels in one image) [77], [285]. To put in more detail, micro-capacitive sensor, which is cheaper and small in size, measures the capacity of the skin of the finger as ridges and valleys has very different capacity. Thermal sensors can detect the temperature differences between fingerprint ridges and valleys. However, thermal sensor in fingerprint data collection has a major technology challenge, which is thermal picture has to be taken in a very short time as temperature is assume to be either constantly changing or the difference vanishes very quickly. Ultrasonic sensor is to use high frequency sound waves to capture a reflection from the subject's finger surface [287]. As an advantage of ultrasound sensor, the picture captured by ultrasonic sensor always have better quality and is available for 3-dimensional data. Unfortunately, ultrasonic sensor is expensive and takes longer time to capture an image compared with micro-capacitive and thermal sensors. [21] thoroughly discusses the impact of digital fingerprint image in the regard of fingerprint recognition accuracy, which especially researches the possibility of using digital cameras to replace specialized fingerprint acquisition devices. It further concludes that digital camera can function perfectly until a compression ratio reaching 30 -40% of the raw image.

- **Image pre-processing** - Due to distortion, brightness, dirt, and scars, fingerprint raw data is very noisy and contains many inter- and intra- personal variance. Therefore, fingerprint data is very sensitive for external factors so denoise and pre-processing fingerprint image to improve clarity often is necessary [287]. Especially in the case of local minutiae feature extraction, fingerprint recognition accuracy significantly depends on the quality of the input image quality to capture enough valid minutiae points. Plus, if fingerprint image is captured by camera, fingerprint has to be segmented from its background from the image. To put in more detail, segmentation methods include colour-based [220], continuous restricted Boltzmann machines(RBM) [418], three-layer RBM [174], and contextual filters have been applied to fingerprint raw data to denoise. Image also can be enhanced so that clear and reliable features can be extracted. Established fingerprint image enhancement methods have three group. That is, pixel-wise, contextual-filtering, and multi-resolution. The goal of enhancement is to distinguish which are genuine minutiae from unclear images.
- **Fingerprint feature extraction** – To extract fingerprint global feature, orientation extraction and ridge frequency are two mandatory procedures. Specifically, the aim of orientation extraction is to evaluate if a fingerprint image is good enough to extract reliable features. Local ridge frequency denotes the number of ridges per unit length along a hypothetical segment centered point and orthogonal to local ridge orientation. For specific example

of feature extraction methods, [99] proposes a privacy-preserving scheme to store fingerprint data in a server or cloud infrastructure, and multi-IoT can be connected. [547] suggests a dynamic anisotropic pore feature model, which pore is extracted by filtering in a high-resolution fingerprint database to improve the fingerprint recognition accuracy of pore-based methods. Plus, deep learning methods like stacked autoencoder [513] for global arch, loop and whorl learning, and CNN [95], [256], [464] for local minutiae learning, have been applied to fingerprint identity feature learning.

- **Fingerprint template generation** – Due to the fact that biometric data are confidential and private, fingerprint template is not allowed to store in any database without further protection. Conventional fingerprint template protection methods include encryption, trusted-executive environment, tokens, and non-reversible transformation etc [548].

Fingerprint features can be matched based on three different methods. Specifically,

- **Correlation-based matching** Correlation between two corresponding aligned fingerprint image pixels, but the matching result is significantly affected by angles and positions. Therefore, single cross-correlation measure usually is not enough to generate a good evaluation of matching result.
- **Minutiae-based matching** Minutiae features can be consolidated into two-dimensional feature vectors, which is most widely used mechanism deployed in automatic recognition system. The result of minutiae matching is based on the spatial distance between two feature vectors.
- **Non-minutiae feature-based matching** - Global features, such as texture, shape, local ridge orientation, are used to identify a person's identity when fingerprint image quality is low. FingerCode [412] is one classic example of it.

Fingerprint system has venerable threats, such as fingerprint-based system can be jeopardized by spoofing attacks. Although the spoofing attacks in fingerprint identity authentication cause major hazard for remote authentication, like [413] has proven that fingerprint can be reconstructed using fingerprint minutiae template, it is very straightforward to deploy multimodal biometrics to strengthen the system security. That is, multimodal biometric feature authentication has been proved to be more secure at a cost extra computation and more raw biometrics data collection. Another good practise of anti-spoofing method is to do liveness detection. [20] proposes an improved feature descriptor to detect fingerprint liveness from fingerprint both level 1 and level 3 features.

For one specific example of generating fingerprint identity template, a set of minutiae $fingerprinttemplate = \{\vec{x}, \vec{y}, \vec{\theta}\}$. Specifically, tuple $\{\vec{x}, \vec{y}, \vec{\theta}\}$ normally

represents an un-order set of the extracted minutia, at location (x,y) with orientation (θ) of friction ridge discontinuities. Taking figure 3 left top one image in level 1 as input, the corresponding feature out set is illustrated as in below:

x	y	θ
178	154	144
71	207	198
98	158	164
.	.	.
126	200	113

Table 3.1: Extracted feature for fingerprint template

Facial image

Face identity generally refers to a person’s physiological appearance feature that is how a person look like. Similar with fingerprint, face identity feature can be distinguished by bare human visual. Human visual indeed has extraordinary advantages in image recognition especially for unconstrained environments. Independent of viewpoint, colour, texture, facial expression, scars, and disguise, human visual seems seemingly effortless to offer an accurate decision. As a result of that, human used to perform better than computers in the regard of face recognition [399].

Automatic face recognition methods have two mainstreams: feature-based local approaches and appearance-based global representation approaches [468]. Differ from these two mainstreams, deep learning computing algorithms can extract face feature in a hierarchical combined approach. However, the drawback of deep learning algorithm performs better when the training data are massive. That is, company like Google and Facebook who have access to that resources are able to proposing more accurate and more robust face verification system, such as Deepface from Facebook [476] who has privilege access to over one billion face images. Specifically, their training size is commonly between 4 to 5 million images for only tens of thousands of subjects [468]. An unbreakable milestone of face image recognition algorithm yet derived from [311], which is trained on 1.2 million images under unconstrained protocol with a verification accuracy rate of 99.41%.

Due to the fact that face image is publicly available, a person’s face identity is rarely used as a single factor identity authentication system to authorize high confidential access. Therefore, face recognition commonly applied to surveillance system, forensic science, access control, and social networks, which is closer to identity verification but not authentication [111]. To put in more detail, face image verification by human vision has compatible performance with robust computing algorithms [353],

which makes face recognition and verification cheaper. Plus, face image is easy and convenient to retrieve, which also has a very high acceptance rate. Therefore, face image is ubiquitously attached to an identity credential to add a stronger tie between the identity and identity holder [399].

There indeed has a massive difference between face image verification and identification [447]. Specifically, face identity verification refers to one-to-one match. That is, matching a pending image with one and only registered template, which the result of face verification commonly depends on a loss or cost function to be minimized. Whereas as a contrast, face identity identification is a one-to-many classification problem, which is to predict the correct label of the image and cross-entropy normally is used as its cost function.

Face recognition system has challenges. One fact about biometric identity authentication system is it never one hundred percent accurate at all [6]. That is, biometric identity authentication has significant bias due to the nature of biometric data and data processing mechanism. To put in more details, [154] talked about the bias in biometric identity authentication system. Specifically, automated authentication systems are bias based on algorithm, and social matters and equipment based technical bias do also generally exist. These bias lead to different outcome with the same biometric data input, which is unfair to the identity claimer. Like for instance, [411] claims that algorithm bias is one of the most serious challenges in biometric identity authentication system.

Plus, even though face recognition in general has an accuracy rate over ninety percent, but there is a consistency in poorest performance in distinguishing people who are dark-skinned or female [346]. As a result of these bias, biometric identity authentication system commonly suffers from either a differential performance or a differential outcome [232]. That is, a differential performance is differences in system performance between demographic groups, and differential outcomes are the system decision's accuracy vary a lot among different groups [142]. For one specific example, aging and age in fingerprint identification and verification is a big issue, and has been well researched in the past decades. Age issue in fingerprint specifically is to investigate if the accuracy is the same when one recognition algorithm is applied to different age groups. Whereas as a contrast, aging issue refers to if the accuracy varies when one recognition algorithm is applied to one person over different time of age. [193] investigates it profoundly and concludes that aging does cause a significant challenge to fingerprint recognition system. It also suggests different thresholds for fingerprint sample authentication to different age groups.

Similar with fingerprint, face image raw data contains a lot of inter- and intra-personal variance as well. That is, features that extracted from photography image are significantly affected by pose, illumination, external environment, cosmetics, and scars etc. In accordance with [370], the biggest obstacle in facial recognition is aging and

external scars or medical conditions. Therefore, denoise is a standardized procedure in face image processing. See **figure 3.3** for a demonstration for face image collection, detection, and standardization.

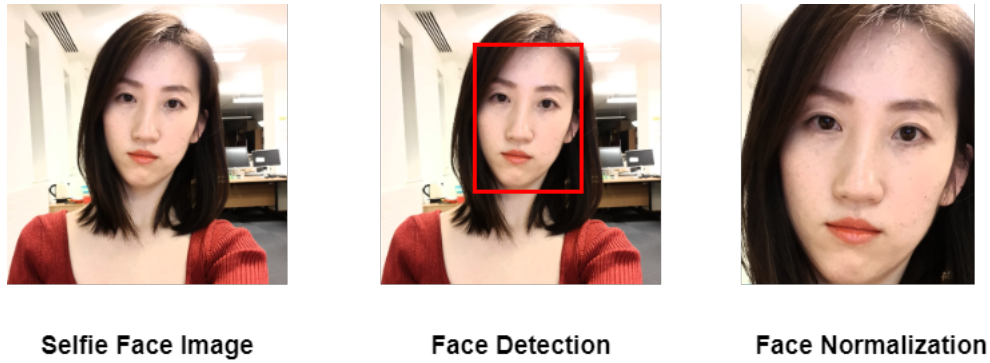


Figure 3.3: a demonstration for face image collection, detection, and standardization.

As biometric data are not allowed to be shared on public cloud, [1] suggests to use both zero-watermark and visual encryption to protect the privacy of biometric data like face image. The advantage of that system is that the decryption can be done by only correct secret key, which is easier compared with conventional cryptography methods. [342] suggests a biometric smart card prototype for secure fog computing data, but there are not any details introduced in the regard of database nor management. [396] uses Merkle hash tree-based encryption algorithm to protect biometric identity data, and proposes Minkowski distance-based authentication method to authenticate biometric identity. It reaches a 100% accuracy on NIST Special Database 4.

For a demonstration of extracting facial image features in a linear combination of basis faces, the facial image features are indeed a vector of weight coefficients. Taking **table 3.2** most left image as an example, the final facial identity can be expressed as in below:

3.3 Identifier in the Metaverse

Identifier is used to clearly identify one and only entity without any ambiguity in both the real tangible world and digital world, which is also required to be readable and machine resolvable [100]. That is, an identifier has the information required to distinguish an entity within its scope of identification. As a comparison to identity, identifier does not necessarily embody the identity of the referent, even though in most cases that may be the case [377]. Plus, identifier has to be unique within its scope, which means one identifier represent one referent only in one identifier system

Vector of Coefficients
0.25
0.11
0.05
-0.03
0.3
.
.
0.1

Table 3.2: Extracted final facial identity feature vector

[116]. To put it into another way, identifier can either be locally unique or globally unique.

Identifier can be completely random or based on any generation rule that is based on the content or the referent of the identifier. Identifier in Metaverse has no differences with identifiers in the digital world, which the identifier can be either in referent of a real world object or a completely made-up thing. However, the Metaverse identifier in the majority case are through Blockchain, which is decentralized identifier(DID). In later suction 3.3.3 decentralized identifier(DID), a thorough research about DID is conducted.

A standardized and generic identifier issuing mechanism does not only improve interoperability, but also improve the rigour of the identifier generic mechanism. Indeed, the generation of identifier can either in accordance with a syntactic context rule or encoding extracted data from referent [486]. Therefore, the uniqueness of the identifier significantly depends on the design of the syntactic context rule and the uniqueness of the extracted information from the referent. However, in most cases, a centralized registry authority is required to maintain the uniqueness of identifier when the system is adopting a generic syntactic context rule, in case identifier user has the subjective intuition to modify or re-issue an identifier [484].

Examples of identifier application in real tangible world are product barcode, passport number, national insurance number, and bank account etc. In digital world, identifier in general is defined by world web consortium (W3C) as in uniform resource identifier(URI) [116]. URI is used by web technology to identify both digital and physical resources. In contemporary view, when URI points to metadata, the identifier is also called uniform resource citation(URC); when URI provides a general mechanism or location to retrieve the resource, it is uniform resource locator(URL); when URI only offers a name string for the identified resource, it is uniform resource name (URN). Indeed, URC, URL, and URN all belong to a type of URI and follow URI generic syntax rule [486].

3.3.1 Identifier definition and generation

A person indeed has quite a lot of identities in lifetime. For example, a bank identity for house mortgages, patient identity for medication and healthcare, staff identity in employment, student identity in education, citizen identity to access to a country's public resources, online social media account to remotely communicate with people, and shopping account to shop online etc. These identities distinguish a person or a group of people from anyone else who are in the same kind, but in the case of identifying one single person at one instance, it is indeed a function of identity identifier [126].

To put it into another way, identifiers can be seen as a syntax that we use to identify a specific subject in a variety of contexts, like communication identifiers as in mobile phone number and email address, identity identifiers as in passport number, driving license, or insurance number etc [106], [377]. Specifically, an identifier can also be interpreted as the detailed content or a specific representation of an entity, which in most cases is name or string. Therefore, an identity identifier is the specific name or string in the identity that can specifically identify one single entity or a group of entities, which represents an instance of identity identification. That is, the name or string in an identity denotatively equals to an identity identifier [377].

For one specific definition of identifier, 'An identifier embodies the information required to distinguish what is being identified from all other things within its scope of identification' [486]. An example of identifier, we all have names to distinguish ourselves from anyone else, like Kate, Jack, or Peter etc. Name is an identity which distinguishes one person from another; however, it is one specific syntax 'Kate' that fulfils the task of identifying Kate as in one instance of identity identification [239], [377]. For another example, bar code is used to identify and distinguish one specific item among all identically produced products in production line. Therefore, put identifier in identity system, identity identifier is the specific name or string that differentiates every single identity that is issued in the same identity system [486]. To be more precise, identity identifier is the specific content of 'who we claim to be', which maps an identity claim to a real human. For short, identifier is used to distinguish an entity regardless of the identifying mechanism and content. There should no assumption that the identifier has to be that entity's identity, even though it may be the case very often.

In computing, identifier is strongly related with namespace where is the syntactic context rule for all identifiers that is how they are going to be constructed, generated, and defined [239]. To put in more detail, the namespace syntactic context can either be any arbitrary string without assigning any great meaning to the identifier at all, or there is a metadata rule to be followed so that extra information can be extracted. For one specific example, U.S. food and product code 38BEE27 with a metadata rule as in below table with an identifier representation meaning of 'canned tomato soup

(concentrated) [186]’. see blow **Figure 3.4** for an example of U.S. Food and Drug Product codes and product code builder.

Structure	Industry	Class	Subclass	Process Identification Code (PIC)	Product
Format	Number	Letter	Letter or Hyphen (-)	Letter or Hyphen (-)	Letter or Number
Sample	38	B	E	E	27
Meaning	Soup	Soup, Conc	Metal	Commercially Sterile	Tomato Soup, Concentrated

Figure 3.4: An example of U.S. Food & Drug Product codes and product code builder [186].

Most important, identifier has to have a clear defined scope to avoid the case of identifier collapse [374]. To put it into another way, duplicate identifier is possible as long as they are not used in the same namespace or identification system, but identifier has to be unique within one identity system. Therefore, identifier has to be unique within one namespace or system. The main function of an identifier is whenever the identifier is called, one and only entity is specified under that system [377]. That is, within the same namespace, every identifier has to be able to precisely and clearly identify one unique entity without any ambiguity whenever the corresponding identifier is called. One typical example of namespace system is file system, which a file’s name can be named in accordance with file naming rule but only duplicate file name under the same folder is forbidden [239]. Plus, identifiers are commonly in hierarchical structure so that some common string or names can be reused. Like telecom mobile numbers, it usually follows the syntactic context of country code at the very beginning, then region code, and finally followed by individual code; Or, a person’s full name is a combination of family name and given-name [421].

Identifier indeed is extremely crucial in identification system, as identifier fulfils the task of mapping a specific string or name to a real entity [310]. Specifically, the entity specified or identified by an identifier has a very wide range, like a concept, real human, an object, and a mathematical function etc. To make the mapping easier and faster, identifiers are usually encoded into one single serial string as in passport number, residential card number, or driving license number etc. In accordance with the property of the identifier, identifier can be separated into either a stable identifier or a dynamic identifier [486]. That is,

- **Persistent identifier.** By applying the same syntactic context rule or generation method to create one identifier for one entity, one and only identifier

will be generated no matter how many different times the rule has been applied to the same entity [163]. In that case, we call the generated identifier is persistent. To put it into another way, one identifier is allowed to map one entity only, but one entity can hold as many identifiers as possible.

- **Dynamic identifier.** Identifier functions as a mapping between an entity with the identifier string. A stable, persistent and standardized identifier improves interoperability, but at the same time it brings a lot of security concerns. Like for instance, in most cases digital identity uses pseudonym as user identity identifier, user privacy can be well protected. However, in the case of user has frequent and repeated actions, user's behaviour is still trackable. Therefore, there is a request for a dynamic identifier that is to improve security as in [221]. Dynamic identifier is a constantly changing string or name, but the referent is fixed. In most cases, the dynamic identifier is changing with some of the referenced content of the entity, but not the dynamic identifier generation rule.

As discussed above, identifier can identify a wide range of entities, and there are quite a lot of established syntax and mechanisms for generating an identifier [184], [374], [486]. In general, identifier can be constructed by either syntactic context rule which arbitrarily assigns a name or string to an entity, or extracting information from the referent entity and then encoded that information into an identifier [374]. To put in more detail, one general rule that all identifier generation rule should follow is avoiding identifier collision, which refers to in one identifier system the same identifier is issued to more than one entity. As a result of that, whenever that collision identifier is called, there are more than one entity is identified. Therefore, the crucial part of designing an identifier issuing mechanism is to keep all identifiers unique [486].

Conventional mechanism of maintaining the identifier uniqueness significantly relies on a central registry authority who takes the responsibility of recording and managing all issued identifiers to guarantee that there are no duplicate identifiers in one system [116]. Another classical mechanism of maintaining uniqueness in identifier system is to store all federated or decentralized identifier in one database and at the same time to request a strong syntactic context. Therefore, as long as all issuers issue identifier in accordance with that strong syntactic context rule, and then cohesively record all identifiers in the same database, the probability of having a duplicated identifier can be designed close enough to reach zero. Like for instance, a classical example of that is the universally unique identifier (UUID) [421]. That is, an UUID is a 128-bit label which is used to identify information in computer system. Based on the standard of ITU-T Rec. X.667 — ISO/IEC 9834-8:2005, UUID is consists of 32 hexadecima digits that is separated by hyphens as in an 8-4-4-4-12 format string [184]. UUID is standardized by Open Software Foundation(OSF), and there are 5 versions has been published altogether. In accordance with birthday problem [255] in

probabilities, it needs to generate more than 2.7 quintillion identifiers in order to have a 50 percent possibility to lead an identifier collision. Therefore, the impossibility of generating that amount of identifier makes identifier collision under UUID scheme impossible as well [184].

3.3.2 Uniform resource identifier (URI) and URI generic syntax

As computing technology gets more and more established, its applications also gradually become an indispensable part of our daily life. Internet of Things (IoT), world wide web, internet, and computing machine learning algorithms, an integration of these technologies makes computing applications ‘intelligent’, and the robustness of the integration significantly depends on persistent and accurate identification of all heterogenetic devices and resources over internet. Therefore, making identification in digital world is very important.

Dated back in 1990, the World Wide Web global information initiative uses identifiers in web-based technology as universal resource identifier in WWW. In year 2005, Internet Engineering Task Force (IETF) published the URI generic syntax, and define URI as ‘... A URI is a compact sequence of characters that identifies an abstract or physical resource [486].’ It obsoletes all previous specifications [485], and merge uniform resource locators [484] and relative uniform resource locators [184]. This new URI generic syntax [486] is a super-set of all current valid URIs and resolves URI references in all available forms on the internet. URI generic syntax is hierarchical and extensible. By declaring extra specific requirements in URI scheme, a further constrained generative grammar for one specific type of URI can be specified. To put in more detail, in URI, uniformity provides significant interoperability, which resources can be identified across variable accessing mechanism. The word ‘resource’ in a general sense refers to a wide range of subject but not only limit to ‘resource’, and the resource may not necessarily be accessible via the internet. Like for instance, a real human being and an organization.

URI can also be summarised as a unique hierarchical sequence of string that identifies resource used by web technologies [56]. That is, URI only guarantee a function of identification but not access the resource. The operation on identified resource is defined by protocol elements but not the syntax. Specifically, in accordance with the RUI generic syntax [486], there are five main hierarchical components in each URI. From right to left, components are listed in decreasing significant order. That is [116],

$$URI = scheme : [//authority]path[?query][\#fragment] \quad (3.1)$$

Where,

$$authority = [userinfo"@"]host[":port] \quad (3.2)$$

To be precise,

- **Scheme.** Established URI schemes include http, https, ftp, mailto, file, data, and irc etc. they are all case-insensitive. However, the typical way is the lowercase. All URI starts with a scheme name, and the scheme further restrict URI in accordance with the scheme specifications, which makes URI extensible. URI schemes are also required to be registered with the internet assigned numbers authority (IANA), even though public unregistered and private URI schemes are still allowed to use. RFC 2717 [378] contains the detailed procedures for registering a URI scheme, which clearly requires a corresponding RFC for the scheme syntax and semantics. All schemes are also requested to follow URI generic syntax rule, but one single scheme assigns identifier specifications within that scheme [116].
- **Authority (Optional).** An authority consists of an optional ‘userinfo’ sub-component and a host subcomponent. Specifically, ‘userinfo’ is a combination of user name and an optional information about how to get authenticated to access corresponding URI specified resource; however, for security concern, ‘userinfo’ component is largely deprecated. Plus, a host refers to IP literal like either a registered name or an IP address, and followed by port. The specified authority takes the responsibility to manage and govern the name space defined by the corresponding URI [116].
- **Path.** A path contains hierarchical information to identify one specific web resource defined within that URI scope. A path segment is separated by slash, which may be zero length as empty field [116], [310].
- **Query (Optional).** Preceded by a question mark, query component contains a key query string of non-hierarchical data. In most cases, the key query string is a pair of attribute-value separated by a question delimiter, which is also helpful to identify resources in the same URI scope.
- **Fragment (Optional).** Fragment directs or identifies a route to a secondary resource by reference to a primary resource, which the secondary resource in most cases belongs to a portion or a subset of the referenced primary resource. Fragment commonly results a retrieving action on the primary resource, and fragment format is decided by retrieved resource representation format. See below **Figure 3.5** for URI syntax diagram and Examples of URI [524].

To put in more details, URI is designed to have a global scope, and consists of basic Latin alphabet, digits, and a few special characters[486]. All characters that are used in a URI function as the specific identifying data and serves as an interface



Figure 3.5: URI syntax diagram and Examples of URI [524].

for identification across different system. Indeed, the ability of transcribing a URI between different medium is considered more important than a more meaning literal representation. Therefore, URI often is encoded as octets for simpler transport or presentation.

URI can be further divided into as a locator, a name string, or both [486]. However, the contemporary trending about URI classification is getting vague as there has not a clear and consistent separation about a digital locator as opposed to a name string. Specifically, when URI only provides the name string of the resource and only used for calling, they are Uniformed Resource Names (URN) and identified by ‘Namespace ID’. When URI does not only provide a name string for the referent resources but also specify a primary mechanism to retrieve and locate the resource, they are Uniformed Resource Locator (URL). Another widely used identifier is the uniform resource citation (URC), which is also a type of URI that is used to point to metadata. That is, an object is identified by a set of attribute and value pairs. Normally, URC belongs to knowledge representation scheme that is to describe resource in URC specified mechanism [407], and some of the pairs can be referent to URI directly. For more examples, the domain name system is a good example of namespace system, which

only give a name string to identify online resource. However, internet protocol (IP) address is a typical example of internet locating system, which locates the identified resources in accordance with the address.

Operation over a URI normally is defined by a protocol element, but general term like ‘resolution’ is defined to be the procedure of determining an access mechanism and locate appropriate parameters required to de-reference a URI. Most important, in accordance with [486], URI specification refers the Augmented Backus-Naur Form (ABNF) notation [374] with its core syntax rule defined within it as syntax notation, which US-ASCII coded character set is used to define its terminal values. Plus, percent-encoding will be deployed when octet’s corresponding character is outside the pre-defined scope. That is [374],

$$pct - encoded = \% "HEXDIG HEXDIG \quad (3.3)$$

3.3.3 Decentralized identifier (DID)

As discussed, all URIs follow the URI generic syntax [486] which aims to identify unique web-used resource. However, the specification of identification needs to be further extended by URI scheme. That is, the classification and function of an URI is assigned by the URI scheme specifications but rather the generic syntax. In accordance with [116], it is known that Microsoft keeps 20 to 40 private URI schemes for their own business number system, but all the rest public registered URI schemes are available and maintained at [281]. IANA [281] maintains 346 URI schemes in total for both provisional and permanent status scheme. Classical URI schemes like http, https, ipfs, mailto, and git are all included as well.

In accordance with W3C, the definition of a decentralized identifiers (DIDs) is “a new type of identifier that enables verifiable, decentralized digital identity. ...DIDs are URIs that associate a DID subject with a DID document allowing trustable interactions associated with that subject [460].” Even though it is claimed that the underlying technology is not particularly defined, it is inter-operable for most existing identifier system.

As a comparison to centralized identifier system, decentralized identifier (DID) is a brand-new form of identifier that is defined by W3C organization in 2000 [460]. Every DID has three components:

1. A controller who generates and owns the decentralized identifier;
2. A subject which is what the DID identifies;
3. A corresponding DID document which contains the mechanism enable DID controller to prove their ownership towards a specific DID.

To put in more detail, DID controllers define the subject of the DID, and one controller can hold as many DIDs as they define[460]. DID also has the capacity to identify a very wide range of subject, from a real person, a company, a tangible thing, to a deep learning data model or an abstract concept, as long as the controller decides what that specific DID identifies are. Compared with traditional identifier, DID is decentralized and completely self-sovereign by DID controller. See **Figure 3.6** for an example of a simple DID document [460]. The DID documents include (cryptography) material, verification methods, or services endpoints, which enable the anyone to authenticate DID controller has the full control over the corresponding DID referent [323], [460].

EXAMPLE 1: A simple DID document

```
{
  "@context": [
    "https://www.w3.org/ns/did/v1",
    "https://w3id.org/security/suites/ed25519-2020/v1"
  ]
  "id": "did:example:123456789abcdefghi",
  "authentication": [{
    // used to authenticate as did:...fghi
    "id": "did:example:123456789abcdefghi#keys-1",
    "type": "Ed25519VerificationKey2020",
    "controller": "did:example:123456789abcdefghi",
    "publicKeyMultibase": "zH3C2AVvLMv6gmMNam3uVAjZpfkcJCwDwnZn6z3wXmqPV"
  }]
}
```

Figure 3.6: A sample of DID document[460].

The invention of DID is to decouple digital identifier from centralized registries, identity providers, and certificate authorities [460]. Due to the fact that a digital identity consists of a unique identifier and its corresponding credential, DID makes verifiable, persistent, and decentralized digital identity possible so that entities like individuals or organizations can generate their own DID in a trustable system [323].

DID controller as the owner of DID who can make changes to the corresponding DID document [214], [460]. One DID can has multiple controllers, as long as they can typically asserted by the control over a pair of cryptographic keys. DID has to be resolvable to its corresponding DID documents [460]. That is, whenever system component DID resolver takes DID as input, it can produce a conforming DID document as an output. Each DID document contains a DID, verification methods, and services about how to interact with the DID subject. DID documents in most cases are serialized to a byte stream as in a chain, which is similar with Blockchain's

blocks [212].

Service within the DID document enables and defines the framework of trusted interactions associated with the DID subject, and other parties might be used to help enable the discovery of information related to a DID [460]. DID has distinguishing benefits, which are no central organization control over all individual's data, and is capable of significantly improved security. Plus, another distinguishing feature that makes DID especially suitable for distributed identity system is DID is persistent, which makes sure that a consensus can be maintained as all nodes enquiry will generate the same result [212].

The salient feature of DID is there is no 'identity provider' any more. However, DID controller, requesting party, and the DID subject can still interact efficiently based on the 'service' component that is defined in DID document. Specifically, 'service' in DID documents defines the mechanism or means of communicating or interacting between different DID subject via service endpoint [214]. DID is requested to register under a decentralized system (DID registries) has to operate independently with service provider and hence is free from any authority control [462].

Current decentralized system still retains some elements of centralized control, such as who controls the source code base and the specifications etc [212]. Plus, DID does allow small portion of parameters in 'query' component [316].

3.4 Identity Credentials

Identity credential is an indispensable part of an identity. In the real tangible world, the identity credential is also physically tangible, such as passport book, identity card, driving license, and birth certificate etc. Whereas as a contrast, the identity credential in digital world is normally a group of codes, which can be represented as an alphanumeric string, an image, or a clip of video or sound etc.

The format of the identity credentials decides the mechanism of how the corresponding identity can be authenticated [437]. That is, physical tangible identity credentials can be verified via face-to-face communication, that is to present both the credential and the corresponding identity subject at the same time. Physical tangible identity normally contains facial image, or legal soft identity such as gender or age etc. which can be used as a criteria to judge the veracity of the credential straightforwardly. Whereas as a contrast, digital credentials and its corresponding identity have to be transmitted over open internet to identity provider altogether to do the identity authentication. Normally, the digital identity is a unique resource identifier (URI) within an IdM, which is used for distinguishing one user from the other in the same database [76]. That is, a URI enables enquiry or index a user's information in database [520], and digital identity provider usually links user's identity with his credential. If user is able to submit the correct combination, user's access request

normally will be authorized, vice versa.

In general, identity credential classification can be made in accordance with where the identity attributes come from [148]. To be more precise, a digital identity credential can be [204] (see **Figure 3.7** for identity classification in accordance with where identity attribute comes from):

1. **What we know.** Password, PIN, and patterns etc.
2. **What we have.** Smart cards, cryptography private key, and ID badge etc.
3. **Who we are and what we do.** Fingerprint, facial image, voice, signature and other biometric identity attributes.

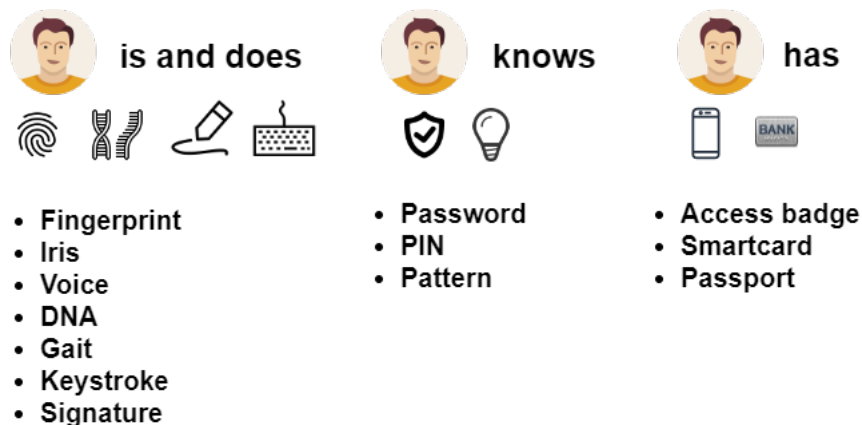


Figure 3.7: ID credential classification based on where identity attribute comes from [111].

Whereas as a contrast, credentials can also be divided into another two groups in according to the nature of the credential data itself [76]. That is,

1. **Private credentials.** Private credentials are credentials should be kept secret and private, such as biometric credentials, password or PIN, patterns or smart cards like bank cards, passports or digital tokens etc.
2. **Verifying credentials.** Verifying credentials are non-private, public available identity information about a subject, such as gender or signatures etc. They are used as a criteria to verify the veracity of an identity claim, or add verifiable credentials to supplement the authenticity of a private credential.

3.4.1 Digital Verifiable Credentials

By definition, identity credentials are identity attributes, which refers to the specific character that is able to distinguish one subject from the others who are in the same group. It has been discussed that in real tangible world, identity credentials are normally physically tangible as well. To put more in detail, the physically tangible identity credential normally contains information about issuing body (bank, government, or gym etc.), the subject and the credentials (photo, name, or identification number etc.), information about what the credential is (birth certificate, driving license, or passport etc.), evidence of how the credential is derived, and the credential constrains (expire date, valid from, or terms of use etc) [121]. By the same token, a digital verifiable credential is able to contain “all the same” information as well [121].

To put more in detail, verifiable credentials aim to express third party verified identity assertion in a standard format in open internet so that the verifiable credentials can benefit as much as third party issued identity credential in the real world. That is, [121] has defined a data model for the verifiable credentials, which claims it is “...a mechanism to express these sorts of credentials on the Web in a way that is cryptography secure, privacy respecting, and machine-verifiable.” To put it simple, a verifiable credential is a verifiable, temper-evident digital identity assertion or claim, which [121] provides a standardized way of expressing a identity credentials on web environment.

Most important, holder of a verifiable credential is able to generate a verifiable presentation to verify his ownership towards the verifiable credential. To be more precise, the basic structure of a verifiable credentials include three parts [121] (see **Figure 3.8** for a demonstration of verifiable credential’s basic structure [121].):

1. **Credential metadata.** As a digital credential that is designed for machine-to-machine communications, identifying the context of communication is crucial, and that refers to the credential metadata.
2. **Claims.** Claims are the specific content or assertions of a verifiable credential, such as a birth certificate, university degree certificate, or a driving license etc. Entity generates claims about the verifiable credential’s subject.
3. **Proofs.** Digital Signatures. Via a signature proof mechanism, proof materials, and evidence of the proof must be contained in this section. The verifiable credential can be proved by either an external proof such as JSON web token or an embedded proof such as linked data signature.

In essence, the verifiable credential data model v1.1 [121] indeed is only standardize the expression and workflow of a digital verifiable credential, which aims for improving the interoperability of digital identity credentials.

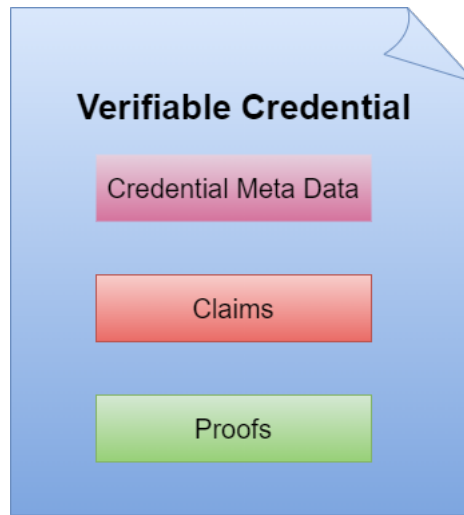


Figure 3.8: A demonstration of verifiable credential's basic structure [121].

3.5 Chapter Summary

In this chapter, identity and identifiers in the Metaverse are well explored. **Chapter 3.1 Digital Twin in Metaverse** and **Chapter 3.2 Identity in the Metaverse** focus on the concept of what identity in reality is, and what an identity is in the Metaverse(digital identity and digital twin identity). Specifically, identity, identity categories, and biometric identity like fingerprint and facial image are further discussed in detail in this section. Followed by **Chapter 3.3 Identifier in the Metaverse** that is about digital identifier and decentralize identifier definition and generation, **Chapter 3.4 Identity Credentials** are explored in terms of what it is. Specifically, digital verifiable credentials and W3C verifiable credential data model are introduced.

Chapter 4

Identity Management System

4.1 Identity management(IdM)

Identity management (IdM) in digital world is a pain. Compared with IdM in the physical world, identity credentials like driving license, passports and birth certificates are all physically tangible, which makes identity authentication can be easily verified by physically presenting the credential directly to a nature person. However, managing identity and authenticate its veracity in digital world is a lot harder than that. Most important, real world identity normally is legal identity that is backed by government, which not only the IdM administrator is trustworthy but also the veracity of the identity is very easy to verify.

Whereas as a contrast, the hardship of digital identity management are derived from three main folds in general:

1. **Pseudonymous.** That is, based on the requirement of level of assurance, digital identity in most cases is not required to be consistent with real life identity at all, which in most cases digital identity is pseudonymous [139]. Therefore, it is difficult to tell that the person who is using the digital identity now is the same person with who is using that digital identity previously.
2. **Privacy and security.** Digital identity management happens over open internet, which is machine-to-machine communications. Even though counter measurements are available to preserve digital identity privacy and protect IdM system security, there are still countless threats, vulnerabilities, and risks.
3. **No root of trust.** The majority of IdM are managed by IdM administrator who plays the role of a trusted intermediary between service provider and identity user to give assurance to both sides; however, system administrator who controls the entire system cannot be completely trusted either because there is a possibility for administrator functional creep [286].

For short, IdM system aims to tackle those hardship so that digital identity can be securely and confidentially managed and used over open internet [286].

Conventional identity management system is required to take care of both the whole life-cycle of a digital identity and the system maintenance, which specifically include the digital identity CRUD ¹ operations, identity authentication, access authorization, and identity user's records for reputation management etc [18], [96], [286], [331]. To put more in detail, IdM system as a computing system which has principles, participants(roles), models, and risk management conducts etc. In this section, all those components will be introduced in subsections. Most important, since IdM system is particularly designed for digital identity management, digital identity so topics like authentication mechanisms, protocols, preserving identity data privacy, and protect system security also will be discussed.

IdM participant's role

Digital identity management indeed is a very established topic. It has well defined standard, framework, and criteria. Conventional digital IdM methods involves four parties who play different roles with an IdM system [314]:

1. User. In conventional IdM model, user refers to the digital identity user who is subscribed service provider's service. A user only uses the digital identity but has no control over it nor own it at all.
2. Identity provider. Identity provider take the responsibility to issue the digital identity for user, and provide identity authenticate for service provider. Normally, the identity provider is the IdM system maintainer [294].
3. Service provider. Service provider provides service to user. Whenever user ask to access the service, service provider ask identity provider to authenticate user's identity and then service is authorized to user.
4. Relying party. A service provider who asks another service provider for user's authentication(or authorization) token to authorize the user's access request but not request identity credentials from users nor identity provider [204].

Established IdM models

IdM system architecture has three well-known and established models, which specifically the model defines a mechanism of how user's identity information is

¹CRUD is the alphabet abbreviation of creating, reading, updating, and deleting operations.

exchanged among different identity providers. To put it in another way, models vary in accordance with how many identity providers within an IdM system [286]. To put more in detail, the three models are [96], [204], [314]:

- **Centralized IdM.** Different service providers use one and the only identity provider in the IdM system. That is, one identity provider issue user's identity for a lot of different service providers. Obviously, centralized IdM is highly interoperable among all service providers, and user can use only one digital identity to access multiple services. But, similar with all centralized system, centralized IdM also suffers the problems of system breakdown, higher security and privacy risk for user, low latency, and central information leakage etc [314].
- **Federated IdM.** Multiple identity providers consist of a union which services the same user and service provider at the same time. The union establishes a trusted domain where user identities are shared within trusted identity providers. The advantage of using a federated IdM is the system security is dramatically improved as abrupt failure at one node will not affect the whole system at all [59], [314].
- **Independent IdM.** Independent service providers use independent identity providers. In most cases of independent IdM, it is the service provider who also plays identity provider role at the same time, like a gym manages its membership in its own database system. This model may cause two major problems. First, one user has a lot of digital identities for different services. It is easy for user get password fatigue so user forgets about any one of them which may also be hard to retrieve. Second, it is not possible for service provider to be interoperable as user has different identities for different service; thus, user's behaviour and data analysis where more business value can be explored is not possible at all. However, the distinguish benefit of this model is service provider do not need to grant trust over identity provider who may be corrupted [314].

4.1.1 IdM principles

It has been mentioned that identity has life cycle. For a formal definition, *identity* life cycle refers to from the birth of an identity to the death of it, which the death stage of an identity may involves more than a termination as its records may required to be kept [490]. Identity life cycle is closely related with identity provider, which in most cases identity provider can terminate, modify, and curtail an identity at any time. By similar token, even though the *identifier* life cycle should follow the life cycle of identity, identifier life cycle in general is longer in case of digital identity collision. Therefore, identifier normally will never be reallocated after initial distribution[490].

Whereas as a contrast, *identity credential* life cycle is very flexible, which is separated from identity provider and identifier. For example, user can change a credential at any time at his own will as long as the identity provider is informed [490].

To manage an identity in the IdM system, the system has to take operations on the identity, which most established and adopted operations include CRUD, identification, authentication, authorization, and auditing etc [490]. To put more detail about auditing operation, it records and analyses user online identity-related activity based on IdM system safety and security policy. Auditing will make an alarm to the system for potential misconducts which allow the system to make in-time adjustment before serious damage occur [76].

[18], [140], [490] has outlined the principles of IdM, which are privacy, security, and usability. To put more in detail:

- **Privacy.** From open internet communications to identity privacy data and sensitive information, from IdM system privacy policy to authentication protocol, privacy in IdM is always the first priority. Most IdM system are privacy sympathetic system, which is to limit the access and usage of private data [416].
- **Security.** The security of an IdM determines its trustworthiness to the general public, which security specifically defines how robust a system is in the regards of defending itself against malicious attack and offer consistent system performance.
- **Usability.** Usability considers more about user experience, which requires the IdM system to be simple, convenient, and effective. Even though a lot of IdM claims so, but in the heterogeneous environment, IdM architecture needs extra precaution as [18].

Plus, apart from above main principles, some general requirements such as affordability, trustworthiness, law enforcement, interoperability and functionality are also recommended to be considered in IdM system design [455], [522]. Most important, a good practise of IdM is also required to give correct response towards user's various access request. That is to say, one major part of one IdM job is to conduct identity authentication and access control. If a user requests access to a system, IdM should be able to distinguish from a forged identity to a real identity [557]. Indeed, a poorly designed IdM can cause serious system problem, which may lead to system break down and massive identity data breach.

4.1.2 Digital identity authentication

A canonical digital identity is a combination of digital identifier and its corresponding credentials, such as bank account and PIN, and email address and password [162]. Both the bank account and email address are identifiers and assigned by service provider(SP). Specifically, an identifier is a string that can uniquely identify an entity within its predefined scope. An identifier does not have to embed any information about the referent identity, even though in most case an identifier does so [116]. Identifiers only fulfil the task of identifying an entity, regardless of the methods, and with no guarantee of identify any mechanism or channels to access the entity. Therefore, the mechanism of issuing identifier becomes crucial in an identity management system [19].

Normally, the tie between the identity identifier and its corresponding user is set up by digital identity issuers or SPs, which the stronger the tie is, the more veracity of a digital identity is [162]. In accordance with their security level requirement, identity identifiers can be either a random string or embed with user legal identity or(/and) biometric identity for extra security [436]. Specifically, identifier embedded identity information can also be either contained within the identifier or linked separately with the identifier [544]. For one specific example, bank account number is the user identifier in that bank identity system, which normally our legal credential is submitted to the bank before a bank account number is assigned by the bank. That is, bank links user legal identity with user identifier. In that case, even though bank account number (identifier) is a completely pseudonymous string, bank is still fully aware every user's legal identity [408]. For one example of identifier contains identity information, soft identity conventionally is set up as the identifier of our biometric identity, like a soft identity bag of legal name, age, and place of birth etc is directly set up as the identifier for fingerprint identity in police criminal biometric database.

In chapter two "Identity and Identifier", it has been discussed that identity credential are classified into three groups, which are paper-based, digitalized, and digital credential [94]. To the relevant subject of a digital identity, the corresponding identity credential is represented by either a secret alphanumeric string or biometric data [94]. In accordance with [243], [294], the identity authentication method is mainly depends on what the corresponding identity credentials are, the authentication mechanism. To put it more precise, the principal digital identity authentication methods include:

- **Knowledge-based static password.** User passes a static secret to identity provider as identity credential to prove user's identity, which the secret is assumed a shared secret between service provider and user [408]. Knowledge-based credential is authenticated through the verification of knowing of the 'knowledge'; however, the major drawback of that is the secret travels online

which has low security [243].

- **Challenge-response.** Challenge based digital identity authentication mechanism does not require to send password to identity provider, which means user's password does not leave user's end device nor goes online. For example, pass phrase of an email address is to run a function of $y = f(x)$, which f is the challenge-based credential and should be designed to be as hard to invert as possible, supposing x is the secret with random Gaussian distribution, and y is the result of the challenge. SP binds y value with the user's email address; therefore, user needs to give correct y to get full access to resources by that email address [239].
- **OTP one time password.** OTP is generated by OTP generator with a combination of user's secret pass phrase and SP's challenge, which aims to protect the system against replay attack and internet eavesdropping[211]. To be more precise, SP sends identity authentication seed to identity provider, which the seed will be concatenated with user's pass phrase. The concatenated output then will be hashed by secret hash function by N ²times. The final output of N times hashed concatenated string is the final OTP [211]. The obvious benefit of using OTP is user's pass phrase never leaves user's end device nor goes online either.
- **Biometric authentication.** Biometric identity credential is biometric raw data itself. Therefore, to authenticate biometric identity, a fresher raw biometric data has to be collected by identity provider whenever the user raises request to access provider's service, and then match it with biometric identity template [251]. However, due to the nature of biometric identity data that is private and confidential, it is nearly impossible to share user's biometric identity template with different identity providers. Plus, user has to trust the identity provider very much so that user believe his biometric identity information will not be sold out for money, even though identity provider may do so [125].
- **Radio frequency identification(RFID).** Radio frequency identification (RFID) is to use radio frequency to read product identity through its bar code or the radio frequency identification (RFID) tag [142], [243]. The benefit of RFID is it can read an item or product's identity through radio frequency enabled near field communications, which normally the identity credential is the product identity as well. However, the major drawback of that is the identity credential is accessible by all radio frequency reader even without any notice at all [262]. The main advantage of RFID is it is very efficient in the regard of identity

² N is user randomly specified parameter [211].

authentication, which normally is only less than a second [142]. Most important, RFID is normally used for distinguishing extremely identical products in product line [441] or localisation [279], [373].

Biometric identity authentication

Normally, biometric data is not recommended to transmit online regardless of it is encrypted or not. Therefore, biometric identity authentication is normally conducted at local end device for access control only as in mobile or laptop access control. That is, authenticating biometric identity at the same place where the biometric identity is stored [111]. However, at the scenario where biometric identity authentication has to be conducted at where is different from the biometric identity template is storage, a smart card is normally used [204]. That is, storing the biometric identity template in a portable smart card, which enables the biometric template machine readable and portable. For example, current e-passport with a micro-chip carries user's biometric identity template for biometric identity authentication at airport e-gates [41].

Biometric identity authentication has four established procedures as discussed in chapter 2 [112], which include:

1. **Raw biometric data collection.** Biometric data needs to be collected first through sensor, camera, or video devices as candidate biometric vector (CBV). Biometrics data contains a lot of intra-and inter-personal variance due to data collection environment, devices, and positions etc.
2. **Identity feature extraction.** Biometric features will be extracted from CBV through feature extraction algorithms. Popular fingerprint feature extraction algorithms are stacked auto-encoder [513], convolutional neural network [464], and restricted Boltzmann machines [2].
3. **Template generation.** Extracted features in most cases will be encrypted and then stored in database as initial biometric vector (IBV) which serve as the biometric template. Classical biometric template protection methods include biometric encryption [157] and cancellable biometrics [372]. Due to the fact that biometric sample varies a lot at each time of collection, a major technology challenge of biometric encryption is how to re-generate the same key from different input biometrics. To tackle the nature variation in biometric data, Fuzzy algorithms, like Fuzzy Commitment[47] and Fuzzy Vault [466], are generally applied to biometric encryption.
4. **Matching.** Whenever a biometric identity authentication request is raised, a fresher CBV will be collected from identity claimer to match with IBV in bio-

metric template database. In most cases, biometric identity verification is one-to-one match, and the result of that depends on a cost function to be minimized. That is, $X = (x_1, x_2, \dots, x_n)^T$ is assumed as the extracted biometric template feature vector, and $X' = (x'_1, x'_2, \dots, x'_n)^T$ is a freshly extracted biometric feature vector. The matching metric rule is to calculate the distance by below formula [48].

$$dist(X, X') = (\sum_{i=1}^n (x_i - x'_i)^2)^{\frac{1}{2}} \quad (4.1)$$

The major challenges of biometric identity authentication in IdM include the unconstrained biometric recognition, interoperability among different identity providers[48]. To put in more detail, unconstrained biometric recognition has to deal with significantly large variance, which make a rigour request on the robustness of extracted biometric identity feature [420]. Plus, as biometric data are private and confidential with clear and strong law enforcement, biometric raw data is not permitted to transmit over different platform or channel, which make interoperability across platform even harder [428].

Most important, due to biometric identity raw data itself is both the identity credential and identity identifier, it becomes an issue of how to find a good identifier for the biometric identity. That is, it is a fact that identity identifier has to be discoverable, biometric identity identifier has to be set up as public information. However, in IdM system, identity provider normally use a pseudonym to protect user's privacy, which indicates the link between biometric raw data and the corresponding identifier has to be totally irrelevant. Even though in real tangible world, biometric identity identifier normally is the identity holder's soft legal identity such as name, age, and ethics group, and date of birth etc. The reason of that is soft legal identity identifier can strengthen the tie between identifier and the identity user nature person [251].

However, the benefits of employing biometric identity authentication are outstandingly favourable in some special scenarios. First, biometric identity authentication does not only authenticate an ownership of corresponding identity identifier, but also verifies the present of one and only real bodily human being [48]. Second, biometric identity authentication has distinctive legal enforcement function [251]. Third, biometric identity is proved to be persistent, stable, and globally unique [116].

Multi-factor authentication (MFA)

In section 3.3 identity credentials, it has been discussed that identity credentials can be divided into what we know, what we have, who we are and what we do, three groups. Indeed, each single factor that can be used to authenticate an identity is

called an authenticator, and a *multi-factor identity authentication* is to use more than one of distinctive authenticators from different credential groups to conduct the identity authentication [148]. To put more details in, an IdM authentication system is recommended to architect multi-factors authentication mechanism in one of either two ways as in below [204]:

1. Multi-factors are presented to verifier altogether.
2. Some factors are used to protect the secret that is going to be presented to the verifier.

In accordance with [204], the more authenticator used in authentication procedure, the securer the system will be. However, two factors authentication mechanism meet the highest security requirement of an identity authentication system [204]. Most important, occasionally, geography location information and device burn-in identifier are mistaken as one of the identity authenticator as well [148]. However, they are not considered as an authentication factors at all but can be used as a criteria to evaluate an identity claim's risk [204].

4.2 Established IdM Protocols

It is being reported that there are more than 5.16 billion of internet users all over the world, who in together accounts for more than 60% of the whole population on the planet [379]. It is not hard to imagine such a massive internet population will trigger a even larger digital identity supply as a person normally holds more than one digital identity. Therefore, a good IdM system is not only required to complete the task of identity management but also required to be efficient and accurate.

As mentioned, IdM is a very mature and established subject, which also has quite a few established framework and standard [314]. For example, HTTP basic authentication[188], OAuth 2.0 [218], OpenID connection [419], and Single sign on [39] etc. These IdM protocols defines established mechanism of how identity assertion and user identity information is exchanged and transmitted among different parties within the IdM. Start from HTTP authentication, in this section the major trending IdM protocols will be examined.

4.2.1 HTTP authentication

does not define how to authenticate an identity but instead defines how to communicate and exchange credentials, authentication tokens, or authorization tokens among parties.

Not all digital service providers request user identity authentications. Based on the requirement for the level of assurance and risk management, some digital service can be made available to user with user self-asserted identity information without the necessary to authenticate it at all [204]. Whereas as a contrast, for higher level of assurance, IdM may be required to deploy multi-factor authentication to authenticate user's identity and meet the requirement.

HTTP basic authentication

HTTP basic authentication is one of the scheme that client authenticate itself with password and user name under each realm [188]. To put in more detail, request for comments(RFC) 2617 HTTP authentication [188] defines a very basic framework for challenge-response based user identity authentication on web, which is *not* considered 'secure' as identity information is transmitted in base64 encoded clear-text format. That is, a web client initiates a request to a server that is expecting identity authentication information, and then the server responses client's request with unauthorized response message that a header contains the identity authentication challenge and a realm attribute is included [240]. when the web client sends the challenge-response back to server, server returns client's original request back to client but with an authorization header after server verifies client's identity.

Digest access authentication scheme

HTTP basic authentication above suffer major security risk as both user name and password are encoded but not encrypted. That is, it is in clear-text format. Therefore, HTTP basic authentication never considered a secure solution for identity authentication but only bring a framework for HTTP authentication scheme. HTTP digest access authentication is to tackle HTTP basic authentication main security concern, which is to encrypt user name and password etc[188]. To put it more clear, the digest access authentication scheme using MD5 encryption algorithm to encrypt the secrets in header, and then attach the cipher as a checksum in the HTTP basic authentication header. The checksum is called nonce which is opaque [188].

4.2.2 Single Sign On(SSO)

Single sign on (SSO) is a cross-domain protocol that allows user to use one digital identity to log into different independent systems or platforms [39], [241]. The real SSO can even fulfil auto-login to allow digital service providers to swift user credential automatically. For example, when user log into Gmail, YouTube is automatically logged in at the same time without re-submit any identity credentials. To be more

precise, SSO commonly is accomplished by IP networks and cookies as long as there is a shared DNS domain. That is, one single user identity authentication gives user access to multiple independent service providers [506].

To put more detail about SSO workflow, the general procedure description is [241], [394]:

1. User initially log in to one of the trusted applications or website via identity provider to do identity authentication.
2. A session token will be generated by identity provider after user initial login is successful. The token is an authentication token, which is a session and contains user authenticated identity information.
3. When user tries to access any other trusted application server or web service, the corresponding trusted application or web server will check with identity provider about if the user is already authenticated. If so, the identity provider will sign on the authentication token with a digital certificate so that the user can be authorized to access the application or web. However, if otherwise, user is required to submit identity credentials to log in.

The key component of SSO is sessions that maintain the single sign on and single logout function [39]. To put more in detail, sessions can be a local session that is maintained by local client application, SSO enabled authorization server, or an identity provider session such as Facebook or Google etc. In essence, there is only one entity authenticate user's identity and then that entity creates one session to share with other domains [39].

SSO is very user-friendly. That is, one digital identity can be used by multiple service providers, and user is only required to log in once only [241]. It does not only sort password fatigue problem but also saves a lot of identity management cost for service provider [222]. However, the criticism of SSO comes from identity security and misuse, as auto log in without the realization of users exists very often under SSO regime [506]. By similar token, another criticism of SSO is the identity provider can cause the single point of failure [222], [241]. That is, if Google IdM is broken down, all it's associated platforms and websites cannot be logged in through Google IdM any more.

4.2.2.1 Open Authorization(OAuth 2.0)

Open standard protocol 'Open Authorization 2.0(OAuth 2.0)' does not provide identity authentication but only access authorization [218], [222]. It exchanges user's access token among applications but never exposing any user's identity credentials [222]. To be more precise, OAuth enables user to authorize web clients to access user's

resources at the other websites [182]. The former web client is called relying party, and the later website is called identity provider [182]. For example, job application website ask user to share the GitHub account codes. The job application website is the relying party(RP), and the GitHub is the identity provider(IdP).

In general, if RP wants to get access to user's data or resources through IdP, RP takes the responsibility to redirect user from its own website to IdP website, which allows user to grant permit for this access. Once IdP notices user's request and user's identity is authenticated by IdP, IdP generates a proof (access tokens or authorization code), and then redirect user back to RP's website. Finally, RP can use the proof to access IdP's resources on user's behalf [182]. OAuth 2.0 has four established modes which define how the authorization token is generated and exchanged among them. That is [218],

- Authorization code mode.
- Implicit mode.
- Resource owner password credential mode.
- Client credential mode.

OAuth 2.0 as a branch of SSO, OAuth 2.0 access token itself does not store user password at all but shares the result of user identity authentication with RP through IdP, which protects user's data privacy and make IdM more efficient. However, the lead author of OAuth 2.0 [215] has been criticized for the core problem of the protocol as "unbridgeable conflict between the web and the enterprise world [222].", which is mainly because enterprise folks prefer SAML, Security Assertion Markup Language, claims rather than JSON web token [222].

To add on, since OAuth 2.0 has to make re-direction from one website to the other, the 307 redirect attack is very well-known [182]. That is, a malicious RP is able to steal user's credential through re-direct user to RP's own malicious website rather than the genuine IdP's website.

4.2.2.2 OpenID Connect(OIDC)

OpenID connection(OIDC) is an identity layer on top of OAuth 2.0, which enables end user identity verification on authorization server [361]. OIDC is applicable to all client ³ types, such as web, mobile, and JavaScript clients etc, and it is decentralized authentication protocol for promoting the task of using one identity to log into multiple unrelated websites [39]. For short, OIDC not only maintains all OAuth 2.0

³Clients here refer to applications on the corresponding server.

features but also adds on identity authentication to its protocol so that user identity can be authenticated by both identity provider and relying party [215].

Specifically, OIDC has three main modes available, which are implicit mode, code authorization mode, and the hybrid mode. To put more details in about the most popular Hybrid mode, which the specific procedure is [222], [361], [419]:

1. IdP requests credentials from user after user's client initiates identity authentication request to IdP, and then an ID token will be returned to user's client if user's identity is authenticated by IdP.
2. To get access or authorization, user's client contacts authorization server for permit by referencing user's ID token, and then an access token will be generated by authorization server and returned to user's client.
3. User's client sends the access token back to service provider to get access to the service, and service provider returns corresponding data or service back to user.

Now OIDC is adopted by some major internet IdP like Google, Facebook, and Microsoft etc, which is applicable to both mobile an web applications [222]. Most important, since OIDC is built on top of OAuth 2.0, features of OAuth 2.0 also applies to OIDC as well. That is, both protocols are suffer the pain of the identity provider can be the single point of failure [222].

To add on the difference between an ID token and an authorization token, that is [303]:

- **ID token.** ID token contains identity authentication information that can be transferred to both IdP and RP for further access authorization and/or identity authentication, which normally is a JSON web token.
- **Authorization(access) token.** a verifiable credential that proves user has authorized RP to access his resources on IdP, which normally is a JSON web token as well.

4.3 IdM Risk Management

Digital identity management as discussed before, takes the responsibility to enrol in, log into, and access to digital service, which is required not only to take care of the full life cycle of a digital identity but also required to maintain the IdM system security and robustness especially in the regard of defending system against threats and preserving user's identity data privacy etc.

To put more details in, there are quite a lot risks and security threats in IdM system. From digital identity proofing which is to established a connection between

one nature subject to a claim that the subject claims it is, to digital identity authentication over open internet and guarantee a user who accesses to the service is the same user as whom accessed the service previously, these procedures are full of risk concerns and threats [204]. As Peter claims ‘No one knows you are a dog if you are on internet [525].’ On internet, it has to be compromised that digital identity authentication in IdM system is a balanced outcome based on a ‘*reasonable risk-based assurance*’ [204], [386].’ Therefore, in this section, we will look into the components of risk management, risk assessment, and the level of assurance in IdM system.

By definition, *risk* is

“a measure of the extent to which an entity is threatened by a potential circumstance or event, and is typically a function of: (i) the adverse impacts that would arise if the circumstance or event occurs; and (ii) the likelihood of occurrence. Information security risks are those risks that arise from the loss of confidentiality, integrity, or availability of information or information systems and reflect the potential adverse impacts to organizational operations (i.e., mission, functions, image, or reputation), organizational assets, individuals, other organizations, and the Nation. [187]”

In accordance with [187], risk management has four main components, which include :

1. **Framing risk.** Framing risk is to construct a risk context, and it defines a specific context in which a risk-based decision is made. To be more precise, framing risk is to make a risk management strategy which the strategy is to address how the system should explicitly and transparently assess, respond to, and monitor risk, so that a risk-based decision can be made by the system.
2. **Assessing risk.** Within the risk frame and through the investigation into the information and communicating flows, risk assessment aims to identify:
 - System threats. An violation of security policy, which the identified event causes harm and breach system security rule [149].
 - Vulnerabilities. A flaw in system architecture and implementation etc that causes system security rule being violated [149].
 - Harm and the likelihood of harm. Given particular threat or vulnerability, the negative impact or harm(or the likelihood of the negative impact or harm) that may happen to the system.
3. **Responding to risk.** Based on a risk assessment outcome, this component defines how to respond to the identified risk, which aims for keeping the risk response consistent.

4. **Monitoring risk.** Monitoring risk *over time*, which is to:

- Evaluate risk response effectiveness.
- Identify change over time in the regard of risk-impacts to the system and the system environments.
- Guarantee the risk response is implemented as how it is defined.

4.3.1 Level of assurance(LOA)

Digital identity authentication normally is required when return visit to the same service provider. Like for instance, digital magazine subscription. The level of assurance(LOA) in digital identity authentication refers to the confidence level about how veracious a digital identity is after it is authenticated, and the person who is returning to access to the service now is the same person with who used the service previously [204]. However, [204] suggests to use system component-based elements to separate the level assurance into three different categories, which aims for improving flexibility in IdM solutions architects. That is, the National Institute of Standards and Technology(NIST) defines that for non-federated IdM system, only two LOA categories are required. They are [204]:

- **Identity assurance level(IAL):** Identity proofing ⁴ process.
- **Authenticator assurance level(AAL):** Authentication process and life-cycle management.

For federated IdM system, the third component is required in addition to the above two components. That is,

- **Federation assurance level(FAL):** Assertion strength.

To add more details in, IAL tackles how IdM new applicants become a formal subscriber through enrolment and identity proofing. AAL category indeed refers to the robustness of LOA, which a risk-based assurance is offered after a digital identity is authenticated. FAL then is federated identity design requirements based on identity assertion and information it conveys, which the level of assurance in this category can be divided in accordance with the potential harm and assertion strength [204]. Indeed, the separation of the discrete LOA category triggers a new trending that is lightweight “componentized” identity management service [204].

To add on, each single category of component-based description of the level of assurance above is further fine-graded into another three level of assurance based

⁴Identity proofing refers to the process of binding and verifying a nature subject’s digital identity with his real world identity [204].

on *the strength* requirements for each category. That is, the strength is assumed a relevant measurement of risk counter-measurement capability. The more strength is required, the system is more risk-bearable. For example, in identity proofing process category, the corresponding three level of assurance based on *strength* are [204]:

1. level 1(IAL1): identity proofing is not required.
2. level 2(IAL2): identity proofing is required.
3. level 3(IAL3): identity proofing is required, and identity assertion information is required as well from the corresponding user.

That is, the higher level it is, the stronger level of assurance it will offer after a digital identity is authenticated. By similar token, AAL has level 1 to 3, and so does FAL. For further referencing of how to define each level of assurance within its subgroup, it is recommended to go [204] page 18 for detail.

For a short summary, LOA suggests that digital identity authentication is a ‘reasonable risk-based assurance’ process, which the level of assurance has three system component based category. Each category has three different levels of assurance, which the higher level it is, the higher assurance it obtained. Most important, to decide the LOA requirement for an IdM system, instead of using one LOA level for the whole system, it is recommended that starting from confirming the LOA for each category first based on the *IdM model and risk*, and then add on privacy-enhancing technique to the system with flexibility [204], [435].

4.3.2 Risk assessment

In the session above, it has been discussed that risk in IdM can be generalized down to the fault’s and error’s negative impacts to the system. Since an authentication decision is made based on reasonable risk-based assurance, IdM system uses the level of assurance to reflect the system’s capability in the regard of mitigating risk [204].

To be more precise, risk assessment is defined as one component of risk management, which aims to identify specific risk elements within the system. To be more precise, IdM risk assessment outcome determines which risk factors that the IdM must mitigate in order to maintain the system’s safety and security. Risk concerns are also required to be tackled by relevant computing techniques and protocol architecture tactics [80]. Therefore, risk assessment is an indispensable part of an IdM design, implementation, and evaluation process [187], [204].

In current literature, there are some established framework and procedures to conduct a risk assessment. [187] defines the general framework about how to conduct a risk assessment, which include four procedures:

1. Preparing for risk assessment.
2. Conducting a risk assessment.
3. Communicate risk assessment results.
4. Maintain risk assessment over time.

In the regard of assessing a IdM system particularly, in according to [187], [204], risk in IdM is determined by LOA levels. That is, it has been discussed that risk has significant impacts on IdM system security, which the design of an IdM system should aim for minimizing those negative impacts. To be more precise, those risk impacts are categorised into six sub-groups, which include [187], [386]:

1. Sensitive information leakage.
2. Agency program or public interests damage.
3. Law violations.
4. Damage, distress, or inconvenience to standing or reputation.
5. Economy loss or agency liability.
6. Personal safety.

In accordance with [386], impact value is labelled as *low*, *moderate*, and *high*, which the required level of assurance is determined in accordance with the value of the impact [204]. Therefore, if the level of assurance is determined, the three categories of LOA can also be determined in the regard of which specific level of LOA it belongs to.

4.4 IdM Security and Data Privacy

Not only particularly for IdM system, computing system in general is required to be secure and privacy preserving. So far in IdM management system, it has been discussed about the identity management's participants, models, design principles, authentication mechanisms, established identity authentication protocols, and risk management. Here in this section, IdM system protection specific conducts and privacy enhancing techniques particularly for IdM system will be explored.

4.4.1 Preserving identity privacy

User data protection for preserving privacy is an important part of IdM, which sensitive information leakage, violation of social law, damage to standing or reputation from impersonation, and economy loss caused due to private identity information leakage etc. have to be tackled. Here in this section, preserving identity data privacy conducts are discussed. It shall be seen that identity data protection is one of the most important part of IdM risk assessment, especially the IdM is involved with user biometric identity information.

In 1968, Westin defines the concept of privacy. That is, "...privacy is the desire of people to choose freely under what circumstances and to what extent they will expose themselves, their attitude and their behavior to others [523]." In current literature, privacy has been classified into two forms. That is, intrinsic (or body) privacy and information privacy [520]. To be more precise, the intrinsic privacy has "a right to be let alone [520]" such as biometric identity itself. An intrinsic privacy is intruded when stealing biometric identity data itself from data collection devices such as sensors and cameras etc. Whereas as a contrast, information privacy refers to Westin(1968) [523] privacy definition, which the privacy is not derived from the generation of the information itself but the attachments to the information makes it become private, such as password and PIN, the PIN number itself is meaningless but when it becomes a credential to a digital identity, it is private data and should be preserved.

To add on, the privacy concept [523] indeed emphasizes the idea of sharing information with limits. Therefore, intuitively, a direct and straightforward conduct to preserve privacy in open internet is to conduct effective access control. That is, access control to preserve digital identity privacy can be powerful especially when thinking about the nature of IdM system that can be easily used as a tool for surveillance [162].

Privacy is in danger mainly in three scenarios, which are:

1. Malicious attacks. Malicious attack may happen at every procedure involving in private data handling, such as data transmission which may cause man in the middle attack or eavesdrop, and user re-direct attack for identity phishing.
2. Misconduct of private data administrator. Private data can be lost due to the data administrator's or owner's misconduct, which is data breach at where the data is stored or from who generates it.
3. Records linkage and footprint tracking. Even though user private identity information is not lost, user's online activity and footprint can still be possibly tracked which is deemed privacy violated.

[217] has constructed three main components to be tackled for privacy-enhancing techniques in IdM, which include:

- A secure infrastructure that can support pseudonym within the risk-based assurance level.
- A secure service that can support anonymity in order to prevent liability of private information.
- The capability of handling system scale issue.

Apart from the privacy involved directly with IdM system itself, there are private security concerns in IdM related fields, such as Cloud-based IdM privacy [375] and life-long privacy [381] etc. Indeed, the general public awareness about what private information can be stored on open internet are very low [18], [493]. Therefore, it is very crucial to preserve user's privacy on user's behalf to set up a healthy and sustainable digital identity environment.

4.4.2 Protect IdM security

In the U.S., reported data breach is increased by 40% in 2006 [162]. IdM involved system security like data theft, data breach, hardware jeopardized, phishing, and illicit communications surveillance draw even more attention since digital identity population expanding massively.

IdM system security is tightly correlated with system robustness, which refers to the system capability of defending itself against malicious attack and keep consistency of system performance [490]. To put more in detail, the system performance consistency refers to for the same input, the system will always consistently give exactly the same output. What is more, scale capability can also be a system performance constrain, which refers to the system is able to offer consistent performance even there is a sharp number of user increase.

Apart from scale capability, system security can be fine-graded into two subgroups, which are:

1. **System threats.** System threats by definition refers to the potential violation of the system security policy [274]. In accordance with [149], threats modelling to assess a system's potential threats should be prioritized before risk assessment.
2. **System vulnerability.** An vulnerability is "...a flaw or weakness in a system's design, implementation, or operation and management could be exploited to violate the system's security policy [445]." the main vulnerability in IdM system include [18]:

what is more, IdM well known malicious attacks like identity-spoofing [253] and phishing [253] should be tackled as well to protect system security.

4.4.3 Network security

Transmitting identity information in open internet is risky. Securing the identity transmit channel and keeping the transmitted data integrity is part of the IdM responsibility. To put more details in, there are some established network protocols that can be used in IdM system to secure that identity data transmission. Such as:

- **Transport layer security (TLS)**. A cryptographic protocol to offer a secure communication between two communicating applications [144]. It is most used in security layer of a network to prevent eavesdropping and tampering, which effectively secure the open internet traffic[144]. To put more in detail, TLS constructs a state connection by a handshaking procedure, which uses both asymmetric and symmetric encryption. It aims to provide data integrity and privacy protection solution for multi-party communications over computer applications. However, TLS is assumed that a reliable transport channel is available, so that TLS can be inserted between the application layer and the transport layer [442].
- **Trusted execution environment(TEE)**. For definition, TEE is “...a secure, integrity-protected processing environment, consisting of memory and storage capabilities [36]”. Based on the concept of a separation kernel, TEE offers a trusted immutable processing environment [417]. To be more precise, TEE regulates data separation, sanitation temporal separation, information flow, and fault isolation security policy, which protects system execution environment both at run-time and database [159]. TEE normally is a separated and isolated components that are out of the “normal” processing environment [36], [159].

4.5 Chapter Summary

Since the proposed system is closely linked with identity and identity system management. In this chapter, the identity management system is investigated. In **Chapter 4.1 Identity Management**, identity participant and roles, models and principles, and established digital authentication mechanisms such as challenge-response, OTP, biometric identity authentication, and multi-factor authentication are all introduced. **Chapter 4.2 Established IdM protocols** HTTP authentication and Single-sign-on are discussed. They are all established IdM protocols that are used to establish or maintain communications between different endpoints. In **Chapter 4.3 IdM Risk Management**, definition of risk in IdM and the IdM risk management components are explored in the subsection. Followed by the introducing the level of assurance to clarify that all digital identity authentication indeed is a risk-based assurance, **Chapter 4.4 IdM Security and Data Privacy** is discussed.

Specifically, preserving identity privacy methods and protecting IdM security methods are reviewed.

Chapter 5

Blockchain

5.1 Blockchain System Overview

Blockchain system draws exclusive attention since its initial publication in 2008 [349]. It does not only challenge traditional computer systems but also brings new form of economy. The price of Blockchain based crypto-currency Bitcoin was sold at only \$ 0.0009 per token in 2008 when it is initially issued. However, it worthes \$ 2,500.00 per token at the time of now in 2023 [536]. Bitcoin value grows more than 2.5 million times over 15 years, which emphasizes the capability of Blockchain system in creating more business value. In accordance with [537], Blockchain application revenue in the US is estimated to reach \$ 19.9 billion by 2025. In 2015, major investment banks JP Morgan, Goldman Sachs, Barclays and IT giant IBM joint together to develop a Blockchain assisted banking system [235]. They believe Blockchain disinter-mediation feature can overthrow current centralized banking system, which sheds the light on the capability of Blockchain applications on Fintech [487], [512], [543]. Apart from cryptocurrency and Fintech, Blockchain has been applied to many other fields as well. Another established Blockchain based application is anti-counterfeit logistic management [115], [190], [368], [482]. DHL and Accenture work on a product serialization project which aims at improving the track-and-trace ability in pharmaceutical industry and can efficiently prevent bogus medicine from circulating in the market [499]. Similarly, Blockchain non-fungible token (NFT) for digital art, Uber for decentralized food delivery, Airbnb for decentralized home stay hotel, and physical device ownership for smart property etc. Blockchain system is penetrating into every walks of our life and gradually become indispensable.

Public Distributed Ledger

Blockchain public distributed ledgers are records of all validated and confirmed

transactions, and it is miners who take the responsibility to maintain the ledger book(block chain) for the system. To put in more detail, Blockchain system has users, and users takes different responsibilities in accordance with their roles in the Blockchain system. Particular users who help the Blockchain network maintain its consensus rule, are called either miners or full nodes [64]. Full nodes mine new block, push it onto Blockchain, and broadcast it to the rest users. All miners are independently distributed connected with each other, which we refer it as the peer to peer (P2P) distributed network [347]. Distributed network makes direct and dis-intermediary communication possible. Plus, all transactions in Blockchain are automatically executed by smart contract, which the execution of a contract is guaranteed even there is no intermediaries at all. Most important, all user are anonymous and represented by an “address” [65].

By another relevant definition, Blockchain is “...distributed ledger technologies enable decentralized, transparent networks that require no central authority to validate data [26].” In conventional computer system, there has to be a ‘centralized trusted authority’ to tackle equivocation problem so that the system is protected against errors and attacks. However, Blockchain revolutionary removes the ‘centralized trusted authority’, and uses a timestamped distributed server automatically records a chain of immutable ledger [305]. Each ledger is corresponding to one particular Blockchain transaction that is validated and confirmed by miner, and all transactions are digitally signed, which are irreversible, immutable, transparent to the public, and timestamped [13], [74]. Even though irreversible is not favourable when disputes happen among users, it dramatically reduces transaction cost as none needs to pay for intermediary fee [235], [347].

To add on, Blockchain can also be described as “...a new form of shared database [328].”, and “The block chain provides Bitcoin’s *public ledger*...is used to protect against double spending and modification of previous transaction records [64].” etc, which all above in discussion confirms the fact that public distributed ledger is a good description of Blockchain.

Blocks in Blockchain

Blocks is the building units of block chain. As seen in **Figure 5.1**, which demonstrates the block header in detailed. Blockchain as its name that is consist of blocks. To the specific content of each block, it contains block header and Merkle Tree [64]. Block hash has double semantics. When block header refers to the identity of that specific block, block header is also called block hash that is a group of alphanumeric string and is generated by hashing all information in the block header [477]. It is globally unique and different with all its kind. At the same time, block header can also refer to a substructure in each block besides of Merkle tree, which has components as below:

1. **Version.** Version of the consensus rule that miner used to validate the current block. In Jan. 2009, the genesis block in BCT was version 1. BCT block version 2 was introduced in September 2012, the major change is coinbase transaction s required to include *block height parameter*. BCT version 3 imposes strict *DER* encoding to all signatures. Latest version is 4 in BCT, and it supports new opcode `OP_CHECKLOCKTIMEVERIFY` since Nov. 2015. Version number suggests miners which consensus rule to follow to validate blocks [65].
2. **Previous hash.** Block hash of the previous block [347]. It guarantees that no blocks can be modified without also modify all previous blocks.
3. **Nonce.** An 32-bit unsigned integer number that miner randomly adjust it to make sure the mined block hash is less than the nBits suggested target threshold [70].
4. **nBits.** nBits is a 32-bits base-256 version of scientific notation, which is a *compact* expression of block target threshold. Block target threshold is a target threshold (256-bit unsigned integer), which a block hash can never exceed this nBits threshold to count itself a valid block. Therefore, nBits uses 32-bit data to compact express 256-bit unsigned integer information, which works as "...a base-256 version of scientific notion [70]."
5. **Time.** It is a Unix epoch time¹ that records at what exact time the current block is being mined. A block cannot be valid block if the time in that block is more than two hours in the future [70].
6. **Merkle root.** Mekle root is the final hash output of all transaction identifier (TXID) in that particular block [69]. However, Merkle tree in BCT Blockchain has compulsory order to place each TXID. That is, coinbase transaction identifier has to place in the very beginning of each block [68].

One single block has to store one transaction at least, and all transactions information is hashed first and then paired with another transaction hash, the final hash string is stored in block header as Merkle root². The very first transaction in each block has to be coinbase transaction in BCT Blockchain [65]. Coinbase transaction generates new electronic coins for the system, which comprises of both system rewards

¹Unix epoch time: Time measurements by seconds. Starts at 00:00:00 UTC on Thursday, 1 January 1970, at when the Unix epoch time is 0.

²Here is a vague description of how transaction units turn into a Merkle tree in block chain, that is transaction raw data is hashed first, and then paired with another transaction hash. This process will keep repeating until only one hash string is left and that is the Merkle root. In section 5.2.1.1, a detailed introduction of Merkle tree will be given.

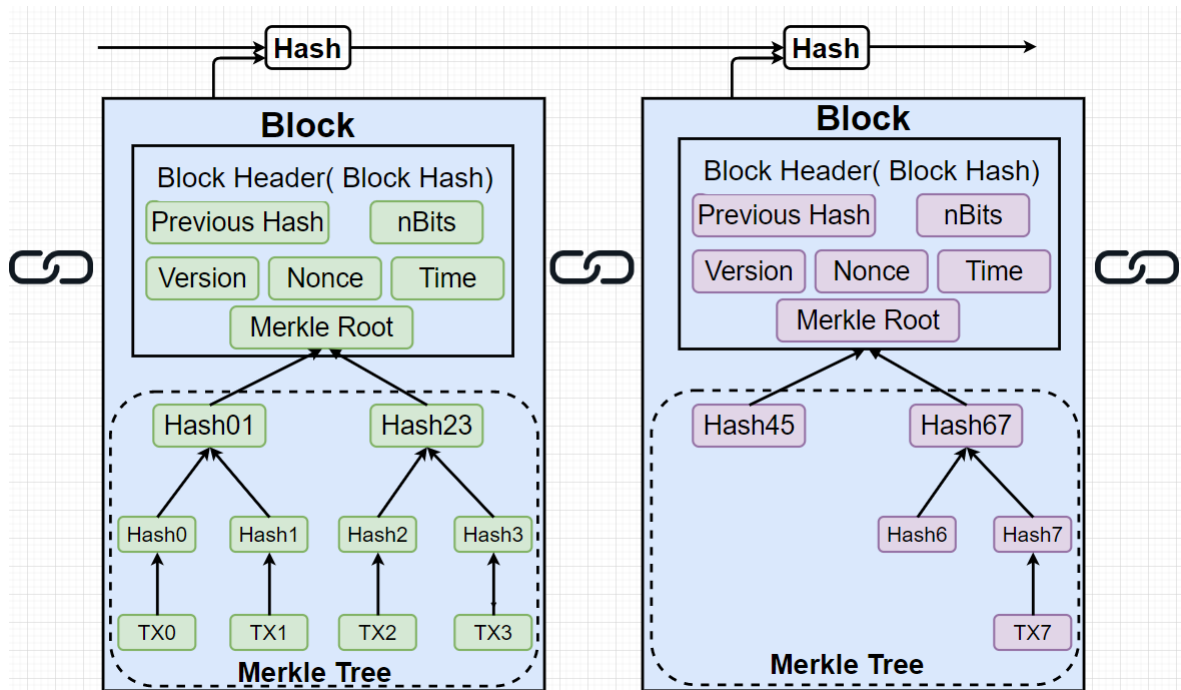


Figure 5.1: Block chain structure in detail.

to full nodes for block validation and transaction fees that are charged from transaction makers [13].

Under the current BCT consensus rule, a block can only be mined if *“its serialized size is less than or equal to 1MB [70].”* The 80-bytes block header is definitely contribute to the 1MB serialized size, and the other two parts of those are also contributing to the 1MB serialize size are compactSize unit and raw transaction [70]. To put in more details, compactSize unit is the total number of bytes (or units) that the following piece of data has ³, and raw transactions in raw transaction format are serialized byte format for peers communications. Raw transaction data has to be in the same order of their TXIDs in Merkle tree first row [208], [549]. All new block with a height that is less than 6,930,000 will reward miner 50 bitcoins, and that 50 bitcoin has to wait until until it is halved to spend, when the halved time is every 210,000 block that is roughly every four years [70].

Block has a height, which is how far that block is away from the genesis block in the regard of number of blocks[70]. Like for instance, genesis block is defined as block

³CompactSize: *“...a type of variable-length integer to indicate the number of bytes in a following piece of data [70].”* However, in Bitcoin raw transaction, the compactSize refers to *“...Number of inputs/outputs in this transaction [70].”*

0. If current block is block 10, it is 10 blocks away from block 0. therefore, current block's height is 10.

As mentioned in block header structure, “time” indeed is one of the very important variable in block header. Specifically, time refers to Unix epoch time that is measured by seconds and represented as an integer. Therefore, the time of when each block is mined is hashed into block hash and constructs a serialized continuous chain, it is claimed that “every block is time stamped” on block chain. Time stamp in accordance with Nakamoto [347], “The time stamp proves that the data must have existed at the time, obviously, in order to get into the hash.” From the time recorded in each block header, it is very straightforward to know how long it takes for the miner to validate a block,⁴ and the time cost to valid each block indeed is another brutal variable in Blockchain system and we will introduce more about it in later sections 5.2.2 subsection mining time, which is a vital parameter that is relevant to system security and scale capability [61].

Chains in Blockchain

Blockchain has well established chain rule applied to several components of its system. Specifically, Blockchain is chained together by block hash chain and transaction input and output correlation chain:

1. **Block chain.** Current block hash contains the hash of previous block. Therefore, all blocks are chained together. Most important, transaction id is also hashed into block hash, therefore, if anyone intends to change any particular data in one transaction, they do not only have to change the data of that particular transaction but also all the transactions data from previous block as well [224].
2. **Transaction chain.** Each input transaction spends the coin of previous output, and each output can only spend once [13]. As coin moves from one transaction to another, “...the input of one transaction is the output of a previous transaction [65]”

In accordance with accessibility, block chain can be categorized into either *Permissioned* Blockchain or *Permission-less* Blockchain [64], [328]. Classic permission-less Blockchain like Bitcoin, all public users can be a miner to maintain the consensus rule for the network without any further system access constraint. However, as a comparison, permissioned Blockchain like Hyperledger, only permitted users can maintain the consensus rule in the Blockchain network [328]. Full nodes are very important in Blockchain network as dishonest nodes have the possibility to destroy

⁴The time cost to mine current block(in seconds) = current block time - previous block time

the network completely. Even though the possibility is proved to be very low in public Blockchain, it is significantly high in a permissioned Blockchain network [430]. By the similar token, Blockchain also categorized into public, private and consortium Blockchain, which the classification is made in accordance with who is control the Blockchain network [143]. Public Blockchain is controlled by the public, private Blockchain is controlled by private organization which is especially popular among business entities. Consortium Blockchain is controlled by several entities and all entities together update the status of the system [532].

As mentioned before, blocks are mined by independent full nodes(miners); therefore, it is very likely that multiple full nodes mine the current block at very similar time. In this occasion, if that happens, it causes the block chain to *fork* [70], [549]. That is, a fork by definition is "...an actual divergence in block chains [65]. To put in more detail, independent full nodes have to broadcast their new block to the rest of users in the system after each new block is generated. If there are multiple nodes broadcast their newly mined block at the very similar time, it will cause the rest of light-weight users a problem as they will be not sure about which one to use. In BCT block chain protocol, "node usually use the first block they see [70]." therefore, there will be a fork (or multiple forks) occur when the light-weight users fork into different chain from different full nodes.

However, forking in that occasion indeed is very easy to sort out, as the opportunity of independent full nodes mine two blocks at the very similar time is nearly impossible. Therefore, as soon as another new block being mined, a longer chain will immediately available to be broadcast in the system. All previous forks will come back to the longest block chain [13], even though it is being said that longer time forking is still possible [65].

By BTC protocol, a fork can be either *hard* or *soft* [65], [307]. Besides the fork we discussed above that fork happens when two full nodes mine new block at very close time, that fork in most cases is temporary and very easy and quick to fix.

To put in more detail, block chain changes system consensus rule to bring new features or prevent network abuse [65]. However, some nodes in the system do not update the consensus rule in time, which causes block chain divergence between the nodes who knows of the new version consensus and some nodes who are completely ignorant about this updates and still adopt the old version of consensus rule. In BTC, a hard fork refers to a permanent divergence from one same origin block chain and therefore separate into different chains [247], see **Fig. 5.2** for a demonstration of this hard fork.

That is, when system consensus rule is updated, it will cost nodes some time to realize this updates. Therefore, some nodes validates a block in accordance with the new consensus rule, but some nodes still do not. Therefore, it causes two kinds of "valid" block exist in the same system (consensus break), which specifically are:

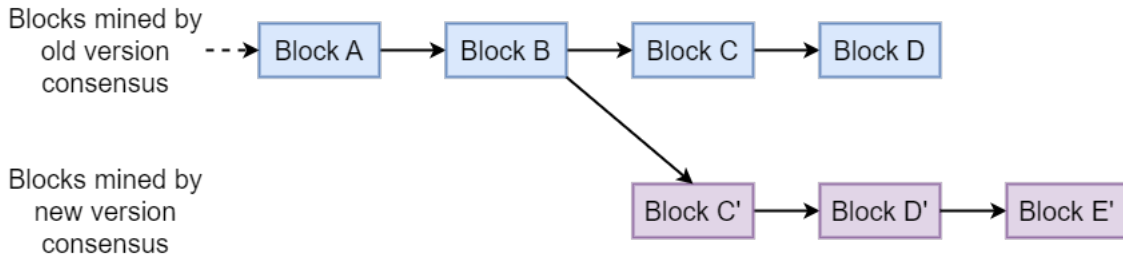


Figure 5.2: Hard fork: Diverging a chain between new version consensus and old version consensus.

1. A block is validated by updated nodes by new consensus rule. Therefore, non-updated node will reject this block as it violates the old consensus rule.
2. A block is validated by non-updated nodes by old consensus rule. Therefore, updated nodes will reject this block as it violates the new consensus rule.

In above case 1, previous valid block are still valid under old consensus rule, but will be rejected by updated nodes. That is, previous valid block becomes invalid after the consensus rule updates or vice-versa [521]. Therefore, the original chain will hard fork into two chains permanently. That is, one chain is maintained by new consensus rule, the other chain is maintained by old consensus rule. Non-updated nodes have to update their consensus to reach a consensus with updated nodes.

Whereas as a contrast, in above case 2, a new block can still be validated by non-updated nodes with old version consensus rule but it is only a matter of being stale. That is, a soft fork happens when a backward compatible change occur to the consensus rule is imposed [307]. As in this case, the new block will be accepted by non-updated nodes but will be rejected by updated nodes ⁵ [247]. Most important, If the updated nodes have more computational power altogether than the non-updated nodes altogether, it is possible to prevent a permanent fork [307]. See **Figure 5.3** for a demonstration of soft fork.

Non-updated nodes indeed is pretty harmful to the system, which may cause significant financial loss and difficult to reach a universally-recognized best block chain [521]. Therefore, in Blockchain system, detecting forking is important. In BCT protocol, a remote procedure call will be trigger to warn operator about the existence of such non-updated nodes if certain criteria is detected, e.g. received block header is six blocks more proof of work than the best chain it considers valid [65]. By the same token, full nodes can detect forks by keeping an eye on the transaction

⁵A valid block under old consensus rule will always be accepted by non-updated node, so does new version block [247].

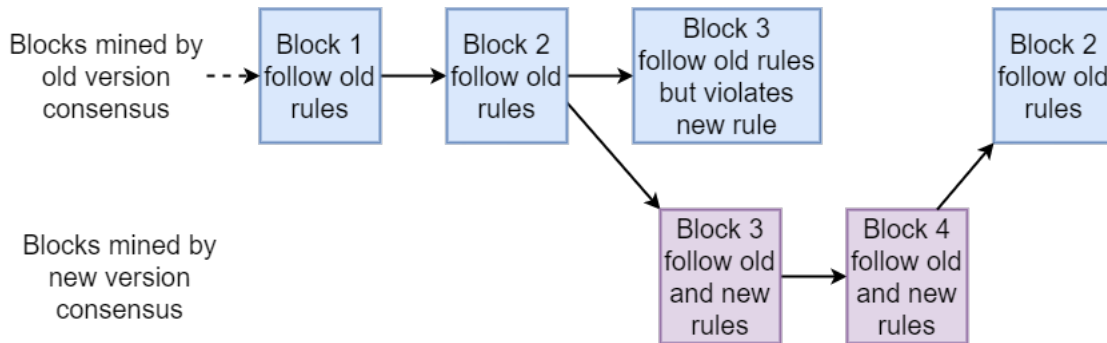


Figure 5.3: Soft fork: Stale block is still maintained by old consensus non-updated nodes.

version number, and simplified payment verification (SPV) clients also recommended to connect with several full nodes to be aware of forks [65], [307].

Pros and Cons

Blockchain decentralized network has some outstanding advantages compared with traditional database system:

1. Blockchain can well protect user private information [511]. Traditional database supported system like banks usually protect client privacy by blocking information access. They share transaction details only with involved clients but not anyone else. As a result of that, banks normally take full control of client's information, which may cause information leakage and privacy issue [234]. Whereas as a contrast, encryption is deployed in Blockchain network to protect user privacy. Taking advantages of Blockchain system, users can even take full control of their own private identity information in a self-sovereign decentralized manner as in [295], [449]. Blockchain transaction is initiated by an address, which is totally irrelevant and non-disclosure of user's identity information; therefore, the privacy is well preserved in Blockchain system.
2. Blockchain can significantly reduce transaction execution cost [235]. Traditional database system has to have intermediaries to dissolve all potential default risk, and usually intermediaries charge high transaction fee. However, Blockchain as disintermediation P2P network which can automatically execute contract by smart contracts [477]. There is no intermediary to charge fees in Blockchain system so the cost is reduced.
3. Data stored in Blockchain are practically immutable [64]. Taking Bitcoin network as an example, if someone wants to tamper one transaction data that

already exists in the Blockchain, they have to re-do the consensus for the block that contains this transaction and all the previous blocks in the Blockchain [347]. Plus, they have to compete with other nodes who are trying to add new blocks in the Blockchain, which is practically impossible [552]. This feature indeed is very favourable especially when further calibration is required or disputes happen.

Blockchain network does have drawbacks as well. To put in more detail, the two major drawbacks are:

1. **Scalability.** Particularly, Blockchain system scalability has three main dimensions [350]. To put in more detail:
 - (a) Size scalability. The capability of adding more users and/or resources to be supported in a distributed system without any sacrifice in the regard of both system performance and user experience. Constrained by computational capacity, storage capacity, and the network etc, size scalability bottleneck occurs when it faces a sharp increase number of requests, which often is a limitation of centralized services in a distributed system.
 - (b) Geographical scalability. Resources, users, nodes, and applications could be far away from each other, but the synchronization communication among them all should not have any delays at all so that all messages are up-to-date. It requires the network between them offers at worst a few hundred microseconds communication, which is difficult to fulfil especially when involved two parties are geographically located very far away.
 - (c) Administrative scalability. Running one distributed system in multiple independent administrative domains often cause policy conflicting in the regard of payment, management and safety etc. A suggestive solution of that is to construct a global computational grid system, which allows a distributed program to directly access resource from another.

Put it more clearly, Ethereum Blockchain network is only able to process three to seven transactions per second, Bitcoin cash is 200 transaction per second [63]. Compared with Visa network process roughly 1,500 transactions per second [124].

2. **Transaction privacy.** In Blockchain, user's privacy is well preserved by anonymous address; however, the transaction somehow is tractable and linkable as all transaction details are public information [55]. That is, to prevent double spending, Blockchain normally opens all transactions information to the public. Like for instance, the value of the transaction, balance under each public key,

and public addresses are all publicly available [64]. Even though Blockchain encrypts user's information, they are pseudonyms but not anonymous [551]. The potential risk of that is user IP address can be linked with his Blockchain pseudonyms [55].

5.1.1 Decentralized distributed system

Indeed, digital data theoretically can transfer from one end to another directly without any intermediary; however, the need of managing the identifier (IP address) from one end to the other creates the necessity of a centralized authority to coordinate and assigning those identifiers [73], [122]. That is why the traditional system are a server-centric architecture in the regard of data storage and control.

For a comparison between centralized, federated, and decentralized system, the most obvious difference is the number of authorities. A centralized system has one and only authority who is at the top of the hierarchical components and takes full and absolute control over the whole system. Centralized system controller supervises all lower level components by instructing their behaviour, and therefore a centralized system behaviour is a result of the central controller's command [73].

Federated system has few or some authorities, and a federated system decision in most cases is decided by a group decision of all authorities [551]. Every participant is also an authority entity in a decentralized system, and therefore, all participants have to agree with and aware of all the things happen in the network [57]. Decentralized system behaviour is an emergent property of all acts upon local components. Each local component is equally responsible for contributing to the global status of the decentralized system, and there is no one component holding more control than another. Decentralized system has a pre-defined consensus rule which all components have to agree with and obey with to maintain system. Based on the number of authorities required to modify a global status of a system, decentralized system is the most difficult one to corrupt and has the most reliability derived from system design [122].

The differences between decentralized system and distributed system is very confusing and has some overlapping at the very beginning. Starting with Unix time-sharing and multi-user operating systems when is back 1969 in Bell Labs [406], computer back in that time was expensive and very rare. To make most use of computing resources, multi-user systems allow more than one user to log into one computer to make computing operations at the same time. Similarly, time-sharing system is to 'borrow' computational power from someone else's computer to support computing operations beyond one single computer's maximum capacity, which is much closer to the idea of distributed system [280]. Assuming different users will not call the same function in the same computer at the same time, multi-user system is feasible

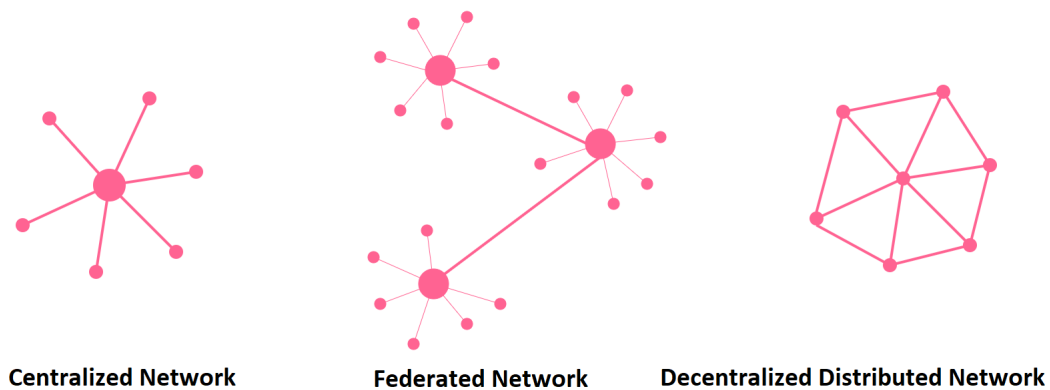


Figure 5.4: A demonstration of centralized, federated, and decentralized distributed network structure.

with very limited number of users. However, these decentralized users do not have their own CPUs at all but just dumb terminals bugged into one shared computer. Therefore, they essentially are still a centralized processor operating system [107].

Later in 1990s when computer becomes widespread with stronger processing power, and internet is available to everyone at worldwide scale at a faster networking speed, computer machines at different geolocation can be directly connected for direct communication and information transfer, which triggers a clear division between distributed and decentralized system. Decentralized system components like an edit server, a compiler service, a raw machine, or an operation system shell etc. have to obey predefined consensus rule to maintain global decentralized system behaviour which is an emergent property [107]. A thorough shift of a decentralized system from low level processing management of running tools and files to communicating with allocating resources and procedures on remote machines happens [122]. Whereas as a contrast, distributed system distributes one massive computing task to multiple independent subordinate components so each component takes care of only a small portion of the whole task, which significantly improve workload efficiency [463] [57].

To put in more detail, a distributed system has two major structural features [479] [463]. That is,

- **Independent nodes.** A collection of networked independent nodes (computing elements). Each single node in a distributed system behaves completely independently from each other. They are fully connected with each other for the realization of system common goal [479].
- **Coherent system.** All users in the distributed system believe they are ‘dealing

with a single coherent system' to achieve a common goal [280], and a coherent system refers to all users in the system do not realize the fact that the system distributes the common goal across all networked nodes. [479]

Most important, the collection of all nodes normally is organized as an overlay network, and the common goal of the system is realized by direct communications between all nodes [463]. A well-known example of the overlay network is the peer-to-peer (P2P) network, which all nodes are connected with each other so that a direct communication path can be guaranteed.

As a result of those two structural features, synchronization and nodes authentication are also necessary in a distributed system [107]. That is, synchronization can have all independent nodes referent to a 'global clock', and nodes authentication (or registration/admission) can ensure all nodes know who they can directly communicate within the same system [280]. Plus, the main aim of deploying distributed system and making building a distributed system worth the effort is to make online resources easily accessible through networked and independent nodes, which the distribution of the system common goal is required to be scalable and invisible (transparent) [479].

5.1.2 Decentralized Autonomous Organizations (DAOs)

The concept of decentralized autonomous organizations origins from decentralized autonomous corporation (DAC) when just a few years after Bitcoin at 2008, but it was later in 2013 when DAO terminology is widely adopted and discussed in dominated websites [87], and the fulfilment of DAO is the breakthrough of Blockchain technology and the availability of its infrastructure [52].

The initial idea of DAC brings new form of corporate governance, which is "incorruptible" tokenised shares to replace conventional dividends to shareholders "without any human interventions" [219]. DAC is heavily correlated with corporate governance and therefore it seems non-relevant with Blockchain-based application in other fields. To fit in a more general purpose, DAC is gradually replaced by some other emerging terms such as decentralized applications (DApps) [258] and DAO [89]. It is widely recognized that DAO does not have a uniformed version of definition. In a more generalized term to describe DAO, it is organizations using smart contract on top of Blockchain network to fulfil organizational purpose [219]. In a more precise term, DAO is defined as "...virtual entity that has a certain set of members or shareholders which have the right to spend the entity's funds and modify its code [164]." To remark, the first widely recognized DAO is a venture capital fund "TheDAO" in 2016 but crash down completely in 2017 [156].

To put in more detail, DAO is to take advantage of Blockchain technology, constructing an easy to join, entirely digitally native, and global in reach organization [527]. Instead of conventional manager and managing board who are the central

authority of an organization, DAO introduces a membership scheme, and all participants (members) are “managers” to the organization [52]. Members obtain control over the organization through smart contract, and organizational decision is normally reached by digital voting scheme over Blockchain among all members. Membership is proofed by the possession of tokens, which also suggests their influence over the DAO decision [89]. To break the system down, there are some outstanding characteristics of DAO can be drawn:

1. **Decentralized distributed organizations.** In DAO, there is no such single entity that dominates the whole organization. Its aim is to achieve organizational consensus through peer-to-peer (member-to-member) voting and communications [254]. Most important, DAO has Blockchain-like disintermediation peer-to-peer transactions owing to its distributed feature.
2. **Highly autonomous systems.** Since DAO is Blockchain-based organizations, all transactions are fully automated by smart contracts. DAO e‘inherits all possible features of Blockchain smart contracts, which includes self-execution and autonomous [52]. Indeed smart contract is one of the most important part in DAO. It does not only defines transactions workflow, but also defines and codes for organizational governance as well. However, automation requires highly standardization and determinism in inputs, which challenges smart contract capability in the regard of performing non-routine organizational tasks [556].
3. **Digitized organizational governance.** “...Organizational governance defines a legal and regulatory framework for the management and supervision of companies aimed at the sustainable interests of the company and its stakeholders [283].” DAO is governed by consensus rule, and a consensus rule as in Blockchain defines the method of reaching an agreement on the updated status of system [156]. At present-time, the established methods are voting and selective group consensus [527]. Under organizational governance rule in DAO, all members can submit organizational proposals for all peer members to review and vote [52].

Conventional institutional economy has three modes of governance: markets, hierarchies, and networks [392]. That is, market and hierarchy mode governance involve balancing transactions cost and frequency through internal/external activities that include incentive, contracting, coordination, the length of control etc [556]. By similar token, network mode of governance shares the same governance concept with market and hierarchies governance; however, network mode governance extends the organizational transaction cost and frequency with a concept of “a shared goal [392].”

Since the emergence of Blockchain DAO, [51] and [138] argue that conventional institutional economy modes of governance are not adequate to fully describe online

mode of governance, which seems wisdom of the crowds or generativity [470]. Therefore, the Blockchain DAO governance mode is typically classified into on- and off- chain, which on-chain governance is governance procedures or protocols are directly programmed into DAO source code, and off-chain governance is all the otherwise scenarios [133].

The goal of organizational governance is to make sure the organizational desired goal is fulfilled by imposing governance mechanism between governors and governed [556]. In DAO, limitations of governance have shown in three aspects:

1. Appeal and decision difficulty. As DAO has no central authority, decision-making and appealing in most cases is through voting [426]. Voting process can be very inefficient and time consuming. Considering Arrow social decision theorem, if there are more than three distinct options available to voters, “...no ranked voting electoral system can convert the ranked preferences of individuals into a community-wide ranking [34].”
2. Centralization. [553] has found that Blockchain indeed prone to be centralized among mining power and code development. Therefore, the decision and appealing process is deemed not as decentralized as it is assumed to be. Counterbalancing miners, users, and development team is a problem to tackle for a healthy and sustainable DAO.
3. Non-standardized inputs. Smart contracts only takes parameters and inputs in pre-defined format and methods. All non-standardized inputs will not be accepted nor executed by DAO smart contract; therefore, it causes DAO limitations in processing non-standardized non-routine inputs and transactions. It significantly calls for the intervention of a mediator in DAO[556].

DAO as a fully digitized organization, there are some established reference model can be made as in [535], [147], and [201]. Through reviewing those referenced models, below DAO construction structure is the mostly referenced model structure:

- **Fundamental infrastructure layer.** It contains Blockchain network, internet protocol, data collection etc. elements to initiate the DAO at very basic level [514]. Plus, if there is any internet of things applications within DAO, it is highly recommended to construct IoT as BoT(Blockchain of things) from the fundamental layer [150].
- **DAO governance layer.** This layer defines DAO governance structure, policy, strategies, voting mechanism, consensus rules, on/off-chain collaborations etc., which should be a group of smart contracts corresponding to its fulfilment of organizational function. Like for instance, contractualization(smart contracts) for

defining a standard procedure for organizational decision making, digitization of business processes, and artificial intelligence involvements etc [514].

- **Incentive layer.** Incentive mechanism alters user willingness to participate in particular activity [8]. Token and incentive mechanism is especially crucial in DAO. Tokens are digital assets in DAO. They are issued to or brought by members, which can represent member's rights over the DAO and function as a credential [164] Plus, investors and participants normally decide to join a DAO in accordance with the incentive mechanism [89]. Therefore, a good design and mechanism of the token and token incentive design is closed linked with if the DAO will success or not.
- **Manifestation layer.** The manifestations of a DAO depends on what business it offers to the public. In most cases, it could be a new funding pool via a new crypto-currency, or a marketplace for trading crypto-currencies [514].

It is a given fact that DAO particular fits in financial organizations; however, there do exist quite a lot DAO in other fields. Like for instance, Metaverse or Gaming (UFORIKA.io), social communication community (Discord), media (Water & Music.io), marketplace (OpenSea.io) etc. DAO also sheds its lights on decentralized collaborative organization (DCO) [130] and distributed cooperative organization(DisCo) [492] etc. which are all variations of DAO but emphasize on collaborations and distributed feature even more.

For a short summary, DAO indeed is a product of Blockchain technology application in self-governance, which Blockchain governance is also applicable to server-less infrastructure, transparent process, and self-enforcement smart contracts etc.[147], [426]. DAO particularly fulfils self- and auto-governance through programming in Blockchain source code [160].

5.1.3 Microservice

Microservice is lightweight, small, and autonomous computing service program that is deployed independently [489]. It chops chunk information system into smaller program with well defined purpose with service-oriented architectures [351], [475]. As a result of that, microservice can scale independently, has more fault-tolerant, and is Cloud native, which can vertically decompose computing applications into business-oriented services [475]. One of microservice most popular business-oriented service is in logistics field [97], [317] and electronic voting [334].

Microservice architecture emerges in 2012 with the aim of decomposing a monolithic software architecture into independently deployable services to ease overall system maintenance and system development [489]. Microservice architecture

environment has proven to be able to be replicated and implemented through a group of smart contracts [88], [474]. [489] uses case study to prove that it is possible to fully implement a system as a micro-service with Blockchain smart contract. That is, the smart contract is the self-contained code to be able supply “service” to the public by itself, and Blockchain virtual machine environment is the gateway to provide such a service.

To distinguishing smart contract architecture on Blockchain and micro-service architecture, a smart contract architecture on Blockchain normally consists of two layers [88], [334], [518]:

1. **Application Binary Interface (ABI).** ABI enables an virtual environment that external applications can interact with Blockchain network so that smart contracts can be called by external applications and then supply services from the smart contract.
2. **On-chain smart contracts.** Each individual services is packed in one smart contract as self-contained program, and then deploy the smart contract on the Blockchain waiting to be called. All smart contracts have different unique address to distinguish between each other.

The communications between layers are managed by remote procedure calls(RPC) from web3.js. Compared with established micro-service architecture, which normally has three layers [223], [489]:

1. **User interface layer.** It offers user a channel to submit service call and input corresponding service parameters. typical user interface are applications of websites or mobiles.
2. **API-Gateway.** API-Gateway can contain a collection of different interfaces, such as web and mobile applications etc. and then API-Gateway can call micro-services to return data to caller. Even though direct communications between graphic user interface to micro-service is technically possible, it is assumed anti-pattern [489].
3. **Micro-service.** Micro-service uses publisher-subscriber pattern to lightweight communicate with other micro-service. Different micro-service has to subscribe the corresponding channel to be able to communicate [489].

Micro-service has established reference model, which normally is a set of patterns and principle. In accordance with [299], [475], micro-service can be classified in accordance with below principles:

- Business capability.

- Updating process.
- Infrastructure automation.
- Endpoints intelligence.
- Heterogeneity and decentralized control.
- Data control decentralization.
- Design for failure.

Conventionally, software patterns are generic solutions to a common problem in a similar software design context [454]. In general, the patterns can be categorized into three groups. That is, architectural patterns, design patterns, and using patterns [9]. To make sure patterns can be re-used, a pattern template is always adopted to represent a pattern [454]. The most used pattern templates include GoF pattern format and the Alexandrian form [480]. In both pattern templates, a pattern is expressed name, the context, and the recurring problem [454].

In accordance with above micro-service principles, patterns are the real mechanisms to join small piece of micro-service together to construct the whole service procedures, which the recommended patterns include [475]:

- **Aggregator.** Invoking the other micro-service to gain access to or process data.
- **Proxy.** Similar with aggregator but without aggregation.
- **Chained.** Chain all small piece of micro-service together so that a summaries response can be given to one single request.
- **Branch.** Allowing simultaneous response processing from possible exclusive parallel chains of micro-service to Aggregator.

Benefits of deploying micro-service rather than traditional large size software include [152], [299]:

- **Scale independently.** Instead of put whole business in a flat service process to make it into a big chunk, micro-service chop it vertically and make the whole business process into individual independent pieces [489]. Therefore, each independent service can be scaled independently in accordance with the demand particularly in that one piece of service rather than scale the whole software.
- **More fault-tolerant.** The break-down of individual micro-service is no harmful to the whole system, which make the whole system or the whole business procedure more fault tolerance.

- **Easier to maintain.** As the size the micro-service is significantly reduced, it is getting easier to maintain in the regard of testing and updating [153].

5.2 Key Components in Blockchain

No matter how different it is among different Blockchain protocols, Blockchain have some common and indispensable components in every Blockchain system, which include transaction, wallet, P2P network, and mining etc. Each components may have sub-components as well, like transaction has digital signature, smart contract, and Merkle tree etc [27]. All main components and sub-components comprise of Blockchain system that is non-equivocation, coherent, decentralize, and distributed.

Here in this section, we will first introduce major key components in Blockchain system, and explain how these components make the system achieve its design targets. As a decentralized distribute system, non-equivocation problem is initially well sorted by revolutionary “proof-of-work” one CPU one vote consensus [347], and we will look into it and make a sense of how it is being sorted. Most important, as a digital currency, a double-spending issue is unavoidable. It will be explained how Bitcoin Blockchain effective prevent double-spending and how the digital economy is formed.

5.2.1 Transactions

Transaction is the most essential component in Blockchain system. It is believed that all the rest of Blockchain components are making sure *transactions* can be securely and smoothly made and recorded. Fundamentally, “Transaction let users spend satoshis [69].” Each transaction has some sub-components, and all sub-components altogether enable all sorts of transactions on Blockchain. The record of a confirmed transaction is called (public) distributed ledger, and all ledgers recorded on Blockchain network are timestamped, non-reversible, and immutable [64], [69], [347]. Due to Blockchain system is decentralized and distributed, all transactions are made directly between peer to peer timestamp server, which is non-intermediary [27].

In Bitcoin Blockchain, the basic unit of one electronic coin is define as “...a chain of digital signatures [347].” Conventional misconception that coins move from one digital wallet to the other, but from the definition it can be derived that indeed coins moves from one transaction to another transaction [65]. Transaction indeed is chained together through hash functions and transaction identity, which is similar with block chains. Most important, transaction can be created by all users of whom the Blockchain network is open to. To describe a fully complete transaction procedure in Bitcoin Blockchain, it goes as:

1. Transaction initiation: public- and private-key pair. User must have at least

one public- and private-key pair prior to initiate a transaction in Blockchain system, which the public- and private-key pair is normally stored in *digital wallet* [257]. Public- and private-key pair is independent of Blockchain network, and is generated by key generation algorithm that only take randomness as inputs [549]. Currently well established and broadly well-used digital signature algorithms in Blockchain include Elliptic Curve Digital Signature Algorithm (ECDSA) [33] and Schnorr Digital Signature [433] etc.

2. Compiling transaction: smart contract. In accordance with different transaction goal, transactions requirements and details can be customized and compiled into different payment script via *smart contract*. Transactions can be sent to receiver in accordance with multiple variables, such as pay to public key hash(P2PKH), pay to script hash(P2SH), and pay to script hash multi-signatures (P2SH multisig) etc [68], [69], [347]. While compiling an input transaction, a digital signature script will be generated in order to submit parameters to meet the condition in the conditional public key script. While compiling an output transaction, in most cases, it compiles a conditional public key script that limits the person who can spent the coin of this output transaction pays to is only the owner of the private key of the corresponding public key in the public key script[65], [542].
3. Transaction principle: “each input spends a previous transaction’s output [69]”. In each transaction, there will be at least one input and at least one output. Each input spends the output previously received, and each output stays in digital wallet to be spent by later input as Unspent Transaction Output(UTXO). Therefore, the input amount has to be larger than the total output amount in each transaction, or otherwise the transaction will be auto-rejected, and all output should only take UTXO as input [27], [65]. However, the difference between output amount and input amount in each transaction will fell into miner’s pockets and become miner’s reward for validating that transaction [69]. An output transaction normally includes index number (suggest location in the transaction), an amount of the output it pays to, and a public key script. An input transaction identifies which particular output to be spent by output index number and transaction identifier(TXID), and must submit a signature script to prove his ownership over the coin they are going to spent in this transaction [65]. Most important, the signature script in input transaction is not signed by signature itself; therefore, it is an open spot for modifying attacks [68]. It is highly recommended that only include necessary signature validation data in signature scripts.
4. Transaction completion: digital signature [69], [347]. Every transaction is signed

by a private key to finish off the transaction at transaction maker side. Digital signature protects transaction data integrity, and verify if the condition in conditional output payment script is met or not [27]. Signature has script as well, which is generated as soon as the transaction is signed by digital signature algorithm and formatted particular in accordance with its Blockchain designed protocol. Transaction then needs to be broadcast to Blockchain main-net to be validated by full nodes, and therefore, the status of the transaction is not confirmed yet so far [224].

5. Validation: consensus rule. Once a transaction is signed off and pushed to Blockchain network, full nodes will take the responsibility to validate the transaction in accordance with consensus rule. As soon as the transaction is confirmed, it will be broadcast again and then recorded on the Blockchain as a public ledger to be reviewed by all public audience [69]. To count a confirmed transaction, it requires at least one full nodes to validate the transaction and add to its own Block. To count, a comprehensive confirmed transaction requires at least six full nodes to validate the transaction[64], [549].

Obviously, there has exceptional transactions that do not follow above process at all, like coinbase transaction in Bitcoin Blockchain [74]. That is, coinbase transaction can only be created by full nodes, and it is particularly used for create Bitcoins to reward full nodes' hard work. Coinbase transaction is always placed at the very beginning of each Bitcoin block, and it exempts from very much a lot of Bitcoin Blockchain transaction rules [69]. Like for instance, the unspent transaction output of any coinbase transaction is limited for at least one hundred blocks afterwards, which aims to prevent double-spending the coins created in current block which may later be stale or un-spendable after forking [65]. Most important, block can have one transaction only, which in that case that only one transaction would be coinbase transaction, even though miners do include additional transactions in almost every blocks to collect additional transaction fees. All transaction has fees and charges to pay for miner's computation work in accordance with the total byte size of a signed transaction. Therefore, it is the miners choose the fee that are happy to charge to put a transaction on the Blockchain network [68]. In accordance with [530], the average number of transactions included in each block is 1953.

Transaction has status, which is either *confirmed* or *unconfirmed*. To protect transaction integrity, transaction is atomic, which means the transaction is considered either not started yet or else completed [503]. As mentioned early in transaction full process above, it requires a minimum of one full node to valid one particular transaction to make it changing its status from unconfirmed to confirmed. Therefore, a transaction bears more risk for revising (or double spending) at any time before it is confirmed [549], and indeed the time cost in confirming a transaction is another

important parameter in Blockchain system that it is going to be discussed further in section 5.2.2 subsection Mining Time.

Apart from the time cost in confirming a transaction, it is compulsory that all transactions have to meet some conditions and formatted in the right formats to be able to be confirmed and validated. That is, the *transaction rule* and *raw transaction format* [542]. To put in more detail about *transaction rule*, which is

1. **Signature script meets conditional public key script's condition.** It has been vaguely mentioned in transaction full process above that when an input transaction is initiated, initiator has to give the right signature script to prove they do own the coins they are going to spend. Therefore, they sign a signature on the transaction and compiles one signature script as a proof of his ownership over those coins so that everyone can verify that. To put some more detail, a signature script is also called scriptSigs, which contains signer's full digital signature ⁶ and signer's unhashed full public key, see **Figure 5.5** for a demonstration of how to compile a signature script in a pay-to-public key-hash standard transaction. That is, all unspent Transaction Output(UTXO) has a conditional public key script, which contains the public key hash of the coin receiver. Therefore, if anyone intends to spends that UTXO, they have to give the right signature script to proof his ownership over those coins. By doing that, all audience of the public ledge can verify his ownership by doing some very simply checksum matching and hashing. In section 5.2.1.3, a detailed explanation will be given about how digital signature is generated and how scripts verify signer's ownership by non-disclosure of the private key at all.
2. **Output amount exceeds inputs amount.** Each input spends the output of previous transaction. Therefore, to guarantee there is enough coins to pay current input transaction, it is compulsory that output amount has to be larger than the input amount.
3. **Ad hoc.**
 - (a) Lock time: the lock time must be in the past time when is no later than the current height). It records the time when the transaction first appear or exist in the system; therefore, the transaction confirmation time (block time) has to be a future time of it [68]. However, a transaction without a lock time is allowed to make up some mistakes in any transaction with a lock time. Since block time is allowed to be up to 2 hours ahead of real current time, so a transaction with a lock time can be confirmed two hours ahead of the lock time expires [69].

⁶In accordance with different digital signature algorithms and variable Blockchain protocols, the representation format and the mechanism of a digital signature vary significantly.

- (b) Sequence number: Each input has a four byte sequence number, which has a maximum value of $0xffffffff$ by Bitcoin Core default settings. Sequence number makes update a transaction among multiple signers possible [68], [69]. Putting the Sequence maximum value as the sequence number will disable time lock. Therefore, to enable lock time function, sequence number has to set below the maximum value even the number is zero [68].

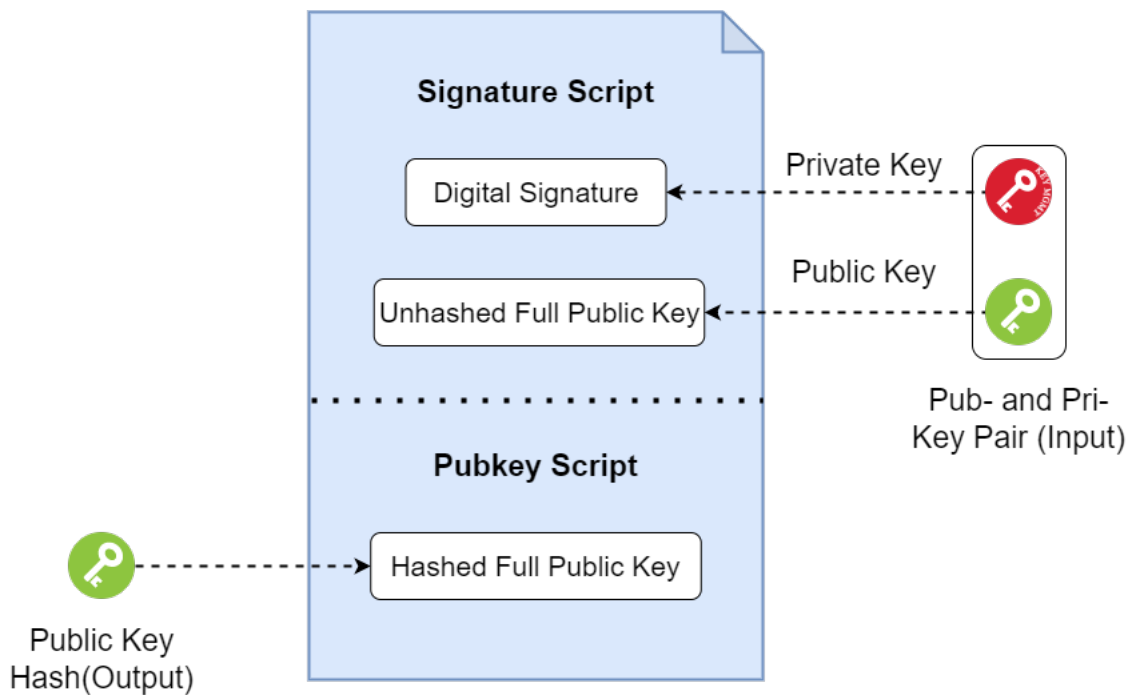


Figure 5.5: Transaction scripting: a demonstration of scripting signature when spending a pay-to-public_key-hash output [70].

Apart from signature script and public key script, transaction has to be formatted in the right format to be spendable and confirmed. That is, transaction has different formats on different Blockchain. Particularly in Bitcoin Blockchain, the transaction script language is *c*, which designed to be stateless but not Turing complete [65]. The benefits of that are two folds [224]:

1. All coins in the network are permanently spendable.
2. A lack of loops (Turing-incompleteness) makes script language more straightforward so significantly simplifies the security model [68].

To put in more detail, all transactions are hashed into blocks in binary raw format, particularly see **Table. 5.1** for a demonstration of the Bitcoin transaction element

lists of top-level raw format structure, and see figure **Figure. 5.6** for an example of Bitcoin raw transaction format [64].

Bytes	Name	Data Type	Description
4	version	int32_t	Transaction version(signed integer).
varies	tx_in count	compactSize unit	Number of unputs in this transaction.
varies	tx_in	txIn	Transaction inputs.
varies	tx_out count	compactSize unit	Number of outputs in this transaction.
varies	tx_out	txOut	Transaction outputs.
4	lock time	uint32_t	Unix epoch time or block number.

Table 5.1: A raw transaction follows the top-level format [70].

Transaction has to follow above instructions so that it can be confirmed or otherwise will be auto-rejected(throw errors) by the system protocols. therefore, the rejected transaction will neither be confirmed nor pushed to the Blockchain ledger book [64].

Transaction has variable types as well in accordance with where the transaction sent to and the content of the transaction, which in Bitcoin Blockchain the standard transaction include pay-to-public_key-hash (P2PKH), Pay-to-script-hash (P2SH), Multisig (multiple signature), Public key (Pubkey), and Null data [69]. To put in more details,

Pay-To-Script-Hash (P2SH)

a P2Sh is to sent the coin to a script hash. Normally, coin spender indeed does not care about the public key script as much as the receiver. As long as spender sends the coins out, who on earth gives the right signature script and spends the coins in the public key script, spender will not mind at all as long as the right signature script is given. However, receiver normally would like to secure his ownership over the coin more than just giving a correct signature script. In this occasion, rather than making a conventional pay-to-public_key-hash, a pay-to-script-hash transaction can does the job for receiver [12], [65]. The script in a pay-to-script-hash transaction is the third

```

01000000 ..... Version

01 ..... Number of inputs
|
| 7b1eabe0209b1fe794124575ef807057
| c77ada2138ae4fa8d6c4de0398a14f3f ..... Outpoint TXID
| 00000000 ..... Outpoint index number
|
| 49 ..... Bytes in sig. script: 73
| | 48 ..... Push 72 bytes as data
| | | 30450221008949f0cb400094ad2b5eb3
| | | 99d59d01c14d73d8fe6e96df1a7150de
| | | b388ab8935022079656090d7f6bac4c9
| | | a94e0aad311a4268e082a725f8aeae05
| | | 73fb12ff866a5f01 ..... [Secp256k1][secp256k1] signature
|
| ffffffff ..... Sequence number: UINT32_MAX

01 ..... Number of outputs
| f0ca052a01000000 ..... Satoshi (49.99990000 BTC)
|
| 19 ..... Bytes in pubkey script: 25
| | 76 ..... OP_DUP
| | a9 ..... OP_HASH160
| | 14 ..... Push 20 bytes as data
| | | cbc20a7664f2f69e5355aa427045bc15
| | | e7c6c772 ..... PubKey hash
| | 88 ..... OP_EQUALVERIFY
| | ac ..... OP_CHECKSIG

00000000 ..... locktime: 0 (a block height)

```

Figure 5.6: Raw transaction: an example of Bitcoin simple raw transaction [70].

script apart from the signature script and public key script, which normally is used to include any script that receiver would like to include. That is, as seen in **Figure 5.5** which is a pay-to-public_key-hash standard transaction scripting, rather than sending a full public key and a digital signature in the signature script, a full script and a digital signature will be included in the signature script [65], [66], [69]. The full script will be matched with a redeem script that is compiled by the UTXO maker. Most important, pay-to-script-hash can be hashed into standard public key hash format with some minor difference. Therefore, a pay-to-script-hash normally is as secure as a pay-to-public_key-hash standard transaction [69].

Null Data

Null data transaction can be mined and recorded on the Blockchain but will be labelled “unspendable” by adding arbitrary data in the transaction, which aims to allow user to package some customized data in the transaction [68]. Therefore, the un-spensible null data transaction will not be stored in UTXO database at all. Most important, null data transaction limits the size of the null data in the transaction. In Bitcoin protocol, the public key script size is limited to 10,000 bytes and no more than 520 bytes per push [68], [69], see below for an example of null data transaction operation code:

$$\text{PubkeyScript} : \text{OP_RETURN} < 0 \text{ to } 40 \text{ bytes of data } > \quad (5.1)$$

which is no signature script at all as null data transaction cannot be spent at all [68].

There are two main operations that apply to all transactions. That is, transaction identity (TXID) and signatures. To put in more detail, signatures is scripted by transaction initiator(sender), which all signature hash types sign transaction’s locktime(timestamp). Locktime is the earliest time that a miner can add it to Blockchain, which also suggests that transaction can only be valid in a future time compared with the lock time. TXID is the hash string of all content in one transaction, which is also the Merkle tree leave in Blockchain structure [452].

Problems and Concerns

Transaction suffers malleability, which is leaving a door open for the denial-of-service attack. That is due to the fact that the Bitcoin signature hash does not protect the signature script at all, which creates an opportunity for attackers to make non-functional modifications to a transaction without rendering it invalid [69]. Even though Blockchain operation codes now check the IsStandard of a transaction, transaction malleability is still a topic to research. Plus, even though Blockchain transaction are made anonymous, it is still tractable of one particular coin transfer path. As all user’s public key or address will be exposed to each other, if the same key is reused very often, it is very easy for any outsider to track that person’s transaction history, including the amount of crypto-currency is in control under that address or public key. Therefore, avoiding repeatedly key reuse is a countermeasure of the system security, which the expose of public key may provide attacker opportunity to reconstruct corresponding private key from it. To add more about transaction-related attacks, they include:

- 51% attack(majority attack). Under proof-of-work consensus, the entity who owns the majority hashing power in the system is assumed can take full control over the whole system [28]. Even though it is proven that this attack is get

harder and harder while the block chain is growing, dishonest nodes have to compete with honest nodes to modify the system [347].

- **Finney attack.** Similar with selfish miner, when a block is newly mined, miner keep it as a secret rather than broadcasting it to the rest miners, a very similar transaction with the transaction in the newly mined block is created, and it will cause the revocation of the second transaction straightaway [22], [185].

Most important, in a digital currency system, double spending is another crucial concern to tackle. That is, double spending refers to the crypto-currency spent more than once [347]. Distinguishing from real physical currency that is design to recycle and reusable. The ownership of a physical real currency is proved by the possession of currency physical entity. Whereas as a contrast, the ownership of a Blockchain crypto-currency is proved by private key with its corresponding public key [534]. That is, a regular double spending may happen in this way: one normal transaction, and the other transaction with the same amount of coin sending back to himself. They try to convince the system that the transaction sending coins back to himself occurs before that normal transaction; therefore, his coins can be spent more than once (double-spending).

Fees

Transaction has fees, which is transaction maker to reward miner's help in confirming and validating their transactions. Fees are not a constant value, which a miner can decide at his own will about how much fee to charge for each transaction [181]. However, in general, fees are calculated by fees per byte based on the demand of space in mined blocks, which fees will increase if the demand increases as well [69]. In a similar fashion, UTXO with coins that have not been moved for a long while are labelled "high-priority transactions" and there is 50KB space reserved for it in each block. High priority transaction is exempt from the normal fee requirements [68]. Plus, Bitcoin Blockchain requires a minimum fee of 1000 satoshi and a transaction with lower fee should be prepared for longer waiting time to be confirmed. Fees are paid through "change UTXO" outputs[68].

Fees are charged not only for rewarding miners but also for system security and health, which indeed is contributing to the system sustainability[465]. That is, as discussed above, fees will increase if the space demand goes up as well. That is, as every confirmed transaction will be recorded on the block chain; therefore, if the storage space of the block chain system becomes a scarcity, the price for pushing one transaction to the system becomes even more expensive to discourage people to make transaction, vice versa [273]. However, fees in Ethereum is charged for limiting computations to prevent the system from malicious attack [293].

5.2.1.1 Merkle Tree

In computing, a tree is a hierarchical data type that is used to represent the data structure, which is constructed by recursive structure like nodes [359]. To put in more detail, nodes are connected by edges, and each node contains one unit of data of the tree. A tree always has a root, which locates at the topmost of the tree and it is the only node who has no parents nodes at all. Any node without a child node is the leaf node, and the nodes who share the same parent node are siblings [359], [444]. Trees sometime are linked with tries, which is a kind of prefix tree that is used for researching or locating specific keys within the tree data structure [90].

Named after Ralph Merkle [337], Merkle tree is to encrypt blockchain transaction data, scripts, and signatures to keep the data integrity and security. Merkle tree is constructed by a repeating procedure, which is to pair the bottom row every two transactions identifier together, concatenate them, and then hash the concatenated data. This process will keep repeating from the bottom row straight up one layer by one layer until at the topmost that there is only one hash result left, and that is the Merkle root. See **Figure 5.7** for an example of the Merkle tree structure. Merkle tree in Blockchain system is formatted in internal byte order when they're concatenated, the resulting Merkle root is also in internal byte order when it's placed in the block header.

Merkle tree root is the final output of hash functions which include all transactions identifiers in the block [69]. All included transactions are placed in an compulsory order:

1. The very first transaction in every block: In BCT Blockchain, the coinbase transaction has to be the very first transaction in every block.
2. Input and output transactions: Input transaction can spent output within the same block; however, the TXID of that output transaction has to place prior to the input transaction. It makes sure that "...any program parsing block chain transaction linearly will encounter each output before it is used as an input [65]."

In the scenario where the bottom row has odd number of transactions, directly pair every two transactions will leave one last transaction with no other transaction to pair. If that happens, in most Blockchain protocol, the final transaction should pair with itself, concatenate itself at its end, and then make a hash [68], [269].

The benefits of using a Merkle tree data type is efficiency when proof some data unit is used to generate a particular Merkle root. That is, "...the proof size(Merkle tree) scales logarithmically with the number of data blocks as opposed to simply concatenating and hashing all the data at once which would require storing large

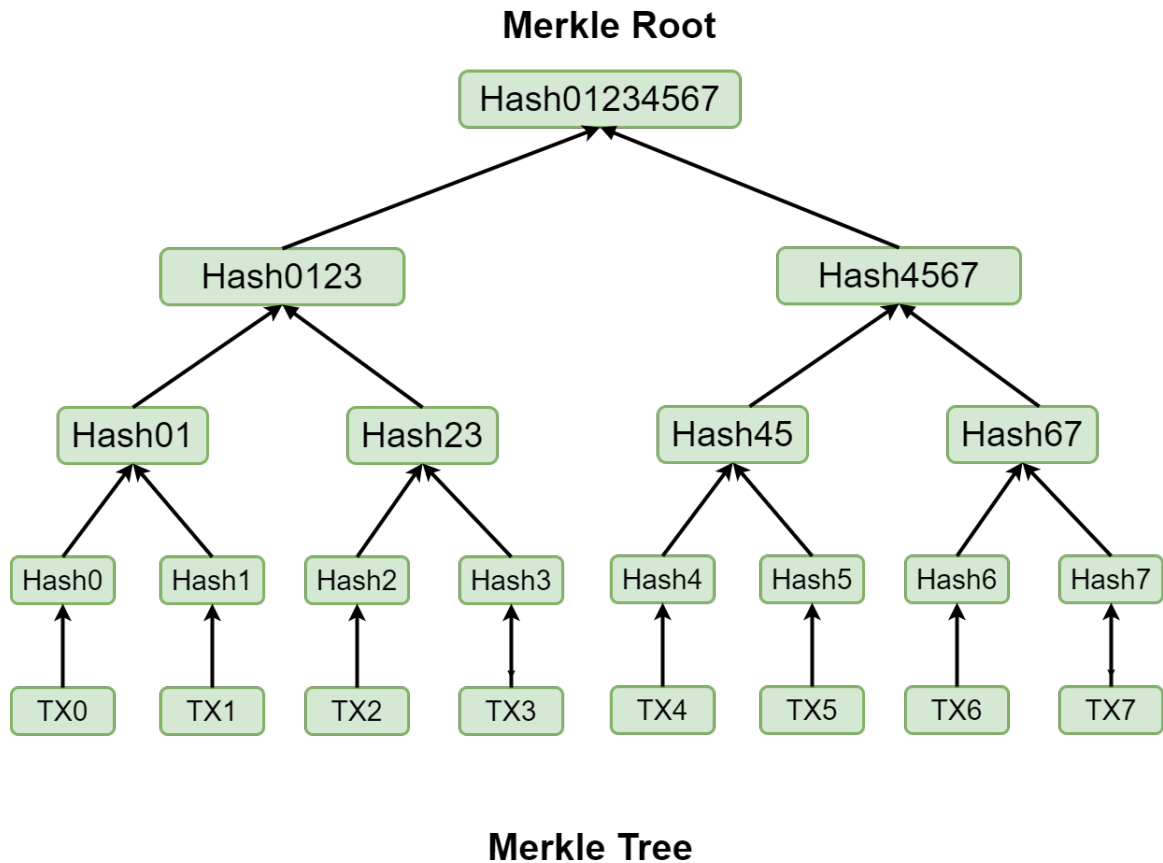


Figure 5.7: Merkle tree: An example of Merkle tree structure and the bottom-up construction process[70].

amounts of data [393]” Like for instance, as in Figure 4.7, if one would like to verify TX6 is one data unit of the Merkle root hash, only TX7, Hash45, and Has0123 are needed to complete the verification rather than requiring the all bottom row transactions that is TX0 to TX7. Particularly in Blockchain, Merkle tree is used to efficiently verify if one transaction is confirmed yet by checking that particular transaction identifier is derived from a Merkle root of a mined Block.

Hash Functions

Encryption in Blockchain is mainly used for maintain the security of the plain-text data being transferred. Similar with encryption, hashing functions are one-way encryption that once the data is hashed, the data can never be recovery from the

cipher unless the hash function is broken [216]. Therefore, there is no decryption at all in hashing function related encryption.

There are some advantages of using hash functions. They are:

1. Hash functions are resistant from quantum computing, as they are based on not only encryption but also black magic of moving and shifting numbers around.
2. ‘Avalanche effect’: change 1 bit of the input data, there is about half the output bits will be changed. Therefore, hash function is an efficient method to verify data integrity.

Hash function also have some properties. Like for instance,

- **Pre-image resistance.** Given y , you cannot find any x such that $Hash(x) == y$. That is to say, given a hash function output y , none can find y ’s pre-image x , which makes $Hash(x) == y$. The opportunity of finding that pre-image is roughly equals to $1e78$ (2 to the power of 256) [401], which takes longer enough and large enough computational power to make it practically impossible to find out.
- **2nd pre-image resistance.** Give x, y , such that $hash(x) == y$, there is no such x' that $x' \neq x$ and $Hash x' == y$.
- **Collision resistance.** There is no such thing ever exist: any x, z such that $x \neq z, Hash(x) == Hash(z)$.

In Blockchain, the most used hash function is the SHA256, which is also well adopted in digital certificates and data integrity [393]. N.I.S.T.[269] developed SHA256, which can take arbitrary length of data as input and then hash it into 256-bits output.

5.2.1.2 Smart Contracts

Given the definition of smart contract from major Blockchain systems, “Contracts are transactions which use the decentralized Bitcoin system to enforce financial agreements.” —Bitcoin [66]. “A smart contract is simply a program that runs on the Ethereum Blockchain.”—Ethereum [508]. “Smart contract is essentially business logic running on a Blockchain.” —Hyperledger [503].

In the context of present time, smart contract are able to define very high customized transactions, which is applicable to not only financial service but also healthcare, energy management [481], logistics, and even political voting [334] etc. Bridging smart contract connection between Blockchain server and oracle server, distributed Blockchain system is capable of receiving external information without

any equivocation at all, which extends smart contract capability in the regards of constructing even more complex functionality to transactions [54].

In a more general sense, smart contract can be summarised as computer program that is designed to automatically execute agreed actions on Blockchain[114]. The aim of using smart contract is to enable non-intermediary, trusted, and peer-to-peer transaction over internet[518]. Most important, comparing with conventional online banking and digital service, the benefits of deploying Blockchain smart contract are three folds:

1. **Trust.** Trust of executing a specific contract is given to the computer program of the smart contract rather than any third party, and the security of a smart contract system is depends on computational power attached to the system but not any central authority who control the system any more [318].
2. **Non-intermediary.** All smart contract initiated transactions are non-intermediary and peer-to-peer, which saves a lot of transaction fee compared with conventional business contract mode [515]. Non-intermediary is in favourable as transaction participants identity can be well preserved, even though the transaction is trackable by the public ledger audience [88].
3. **Automatic execution.** A smart contract is initiated by posting parameters to it, and due to its atomic nature, the transaction is either not started yet or otherwise completed after initiation. Smart contract will automatically execute as soon as new parameter input is given, and there is nothing can stop a smart contract from self-execution unless it is the smart contract itself call itself to cease [273]. Therefore, a smart contract is able to maintain a fair business procedure with all participants in the same smart contracts [181].

As mentioned above, smart contracts are programs on the Blockchain. Therefore, smart contract functionality and capability vary drastically on different Blockchain. To put in more detail, smart contract on Bitcoin Blockchain network is designed as a simple transaction to enforce financial agreement, which is mean to minimize dependency on any third party [347], [362]. Therefore, apart from defining Bitcoin standard transaction, the smart contract on Bitcoin Blockchain are mainly for the use of preserving user public key such as CoinJoin smart contract, micro-payment smart contract, and escrow and arbitration smart contract etc [64]. Whereas as a contrast, smart contract on Ethereum are crafted for functionality diversity and high extensibility to transaction. Therefore, smart contract on Ethereum has much more types compared with smart contract on Bitcoin Blockchain [273], [318]. As for Hyperledger Blockchain smart contract, two main types are defined [503]:

1. **Installed smart contract.** Prior to the Blockchain network launch, installed smart contract can define business logic through validators in the network.

2. **On-chain smart contract.** Very similar with transaction, a on-chain smart contract defines business logic via smart contract itself and then is committed to the Blockchain system by initiating a transaction, which the smart contract finally becomes one recorded and confirmed ledger on the Blockchain.

Processing Smart Contract

One Blockchain system can has as many smart contracts as required as long as the system has space to store it. Each smart contract is differentiated with each other by contract address like all the rest of users on the system [91]. Transaction requests initially are sent to the right smart contract in accordance with contract address, and then being executed automatically [66]. There are some Blockchain platforms that can be used to construct smart contracts, like Ethereum, Bitcoin, and NXT etc. Most important, users do not write new code every time they want to request a computation on the Ethereum Virtual Machine(EVM), programmer store these codes and use them whenever those codes are called [432], [503].

To put more details about how a smart contract is processed, in accordance with [273], [503], a smart contract is initiated by an input with four variables:

1. Contract identifier.
2. Contract request.
3. Optional transaction dependencies.
4. Current state.

Inputs go through contract interpreter and then give four variables as outputs [273], [503]:

1. Accept/Reject.
2. Attestation of correctness.
3. State Delta⁷ including any side effects.
4. Optional ordering hints.

which the contract interpreter specifically includes the smart contract code and the current state of the ledger. Contract interpreter waits to receives transaction calls and take immediate actions upon the call [273], [503], see **Figure 5.8** for a demonstration of the smart contract processing procedure.

⁷There will be no Delta data available if the transaction is rejected by the contract interpreter [503].

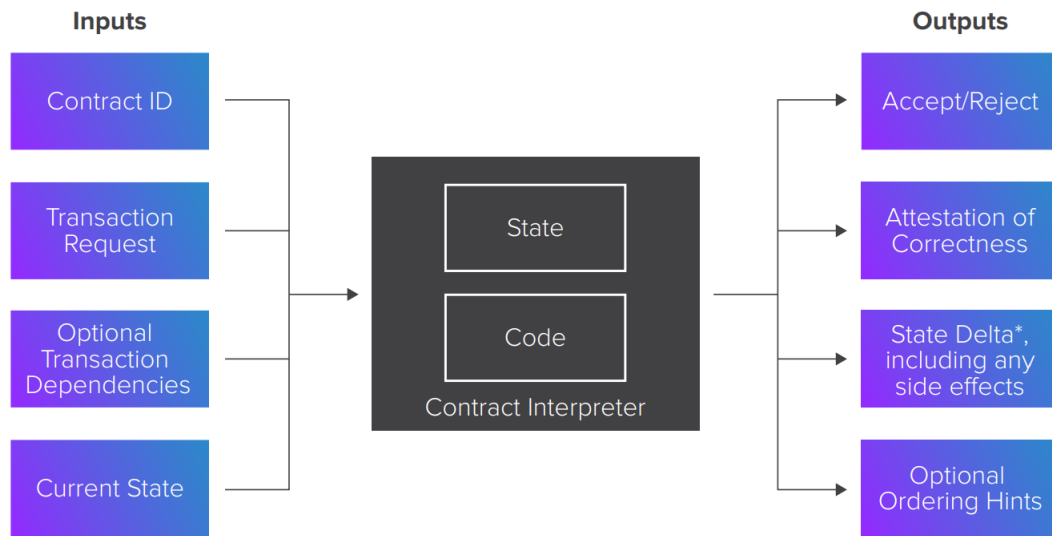


Figure 5.8: Smart contract processing: A demonstration of smart contract processing procedure [503].

It is very common that a smart contract throws errors out after inputs are submitted. It has been generally outlined that smart contract errors are either syntax error or logic error [503]. That is, syntax error such as invalid inputs, invalid signature, and repeated transaction requests occurs to any transaction that will lead to transaction being dropped. However, in a more complex scenario where a logic error occurs, *the system policy* should decide whether it will be executed or not [273], [503].

Smart contract is highly correlated with consensus rules, as consensus rule defines the correct format of a smart contract output and the transaction confirmation rules [181]. That is, the contract interpreter packages all *output* to the Blockchain consensus server for making a commitment to push the ledger to the Blockchain network [68], [503].

There do exist some significant limitations in smart contract, such as smart contract alone cannot get information about ‘real world’ events because by design, relying on external information could make it extremely difficult to reach any consensus in distributed system, which is very important for security and decentralization [508]. Oracle, which is a key-pair server, has been recommended to tackle non-external information limitation [78]. It does have Most important, another major limitation in Ethereum smart contract is the size has a maximum of 24KB or otherwise it runs out of gas [432], [508].

Third Party Dependencies

It is well acknowledged that smart contracts make transaction atomic, which not only keeps the integrity of a transaction but ensure that the transaction completion as long as it is started. Therefore, a dependent action in smart contract has to be committed together, which a dependent action is outlined as in “if . . . , then . . . ”[503]. As there are quite a variety of Blockchain systems available in the currently literature, the frameworks of how a dependent action should be captured vary a lot as well. That is, dependent action can be captured in one single smart contract, or otherwise capturing dependency across of multiple smart contract is still possible. However, the reference of the dependency to multiple smart contracts has to be made either *implicit* or *explicit* through atomic commitments[503].

To put more details about *implicit reference* and *explicit reference* [483], [503]:

- **Implicit reference.** A smart contract without any references will make it impossible to figure out what is the order of execution among multiple dependent smart contracts, for example Bitcoin smart contract normally has implicit references. Therefore Bitcoin sorts ordering issue in implicit reference is to constantly ignite transaction calls to the transactions until the prerequisite transaction are found and completed.
- **Explicit reference.** Explicitly specifies transaction order in combination of transaction identifiers.

Dependency has dependency graph, which can be either *cyclic* or *acyclic* [483], [503]:

- **Cyclic.** Cyclic dependency refers to the order of the multiple referenced dependency transaction starts at and finishes off at the *same* smart contract. In this scenario, it is required that all cyclic referenced transactions have to be confirmed and pushed into the same block on the network.
- **Acyclic.** Acyclic dependency refers to the order of the multiple referenced dependency transactions start at and finishes off at *different* smart contract. Therefore, as long as the dependent transaction occurs prior to the referenced transaction, it is permitted to put the transactions into different blocks.

It has been mentioned that Blockchain smart contract on its own cannot retrieve external information, as it makes a consensus impossible and therefore jeopardises the system completely. However, it has been suggested and well established that oracle servers can be a solution to push external information to the smart contract without any equivocation problem at all [54].

As a third-party service, oracle is able to fulfil the task of pushing different types of external data to smart contract to execute contractual agreements [497]. Oracle itself is not the origin of the external data, but it is a architectural layer that makes enquiry, verify, and delivery the external from the data origin to the smart contracts [54], [78]. Most important, some oracles are even able to send smart contract output to external servers [54].

In accordance with the source of the external data, oracles can be categorised into [54], [497]:

- **Software oracle.** Software oracle normally collects data from external websites, databases, and servers etc. It enables real-time data enquiry, which is the most used oracles, and the external data being collected normally are foreign exchange rates, real time flight information or crypto-currency prices etc.
- **Hardware oracle.** Hardware oracle collects data from internet of things such as sensor, actuators, bar code scanners, transportation, and robots etc.
- **Human oracle.** Human oracle collects human based external information such as expertise opinion, custody, and authority feed backs etc. Human oracle is in use when the human(or human department) is crucial, and human oracle normally has well designed identity authentication system to prevent fraud and faking information.

Apart from above oracles, computation oracles can perform “off-chain” computations to save smart contract transaction costs, and inbound/outbound oracles that can not only delivery external data to smart contract but also delivery smart contract output to external world [54].

Oracles can be either centralized or decentralized [54]. That is, centralized oracle is depends on only one entity to delivery external data to smart contract, which clearly causes Blockchain system being criticised as centralized system and suffer trust issue again like every centralized system[497]; Whereas as a contrast, a decentralized oracle is to collect external data from multiple entities. Based on pre-defined social consensus, data returned from multiple entity will be accepted if and only if multiple entities return exactly the same data to the smart contract. It well tackles the criticism from centralized oracles. However, decentralized oracles normally takes longer time to return a data request from smart contract, which is inefficient [54]. To sort all the problems discussed above, the concept of “provable things” is brought in [497].

To put in more detail, the main idea behind “provable things” is to provide a solution to proof the data that oracle submitted to smart contract is not modified and authenticated. Without assign any trust to oracle, the integrity and authentication of the data can be verified by attaching verifiable digital signatures to the external

data [123].

Smart Contract Related Issues and Tackling Methods

In accordance with the survey [327], there are 16 problems are identified in Blockchain system that is related with smart contract, which is categorized into three groups:

1. **Blockchain mechanism.** Blockchain mechanism related with smart contracts refers to the consensus rule, which defines the method and mechanism of how a transaction should be formatted, business logic, validation, and confirmation process. To put in more detail, proof-of-work consensus is constantly criticised for energy waste, proof-of-stake suffers the problem of nothing-at-stake and therefore cause massive security and lack of transaction finality issue [483].
2. **Contract source code.** Smart contract as a program coded on Blockchain, which suffer coding errors, malicious attack, and poor design issue as every other codes in computing system [503]. Particularly in Blockchain smart contract, the source code issue can come from call-to-the-unknown, lack of privacy, and exception disorder etc [327]. It has been discussed that transactions on Blockchain suffer privacy problem, and that problem can be tackled by customized smart contracts such as zero-knowledge-proof secured user address or the CoinJoin smart contract in Bitcoin Blockchain [69].
3. **Blockchain virtual machine.** When a smart contract is invoked by another smart contract or self-invoked, there is a risk of the stack-overflow exception [438]. It has been found in another study [321] that there are nearly one-third of smart contract on the market suffer stack overflow problem.

Even though problems and issues exist in smart contracts, luckily, there are some methods can be taken to tackle those issues. Like for instance, the consensus problem can be well tackled by leveraging the consensus to a combination of both proof-of-work and proof-of-stake [155]; To the problem of call-to-the-unknown, avoiding internal calls and typing correctness are suggested [327]; As for the stack overflow issue, it can be tackled by checking if the stack size is reaching its limits beforehand [137], or check up the return value of that particular call execution [438].

5.2.1.3 Digital Signature

To sign a signature on a transaction, only a private key is required. Even though the digital key generation algorithms are drastically different, the private key generation

method is very consistent which is only take randomness as inputs. Most important, to verify a digital signature, one only needs the public key of the signature [246], [313].

In Blockchain system, digital signature is used to sign off a transaction and complete the transaction procedure at user-side [347]. The functionality of a digital signature in Blockchain system are two folds [265]:

1. Check the integrity of the encrypted transaction data.
2. Verify the identity of the message sender with the public key contained in the signature.

To generate a digital signature, usually only three mathematical equations are required, which are:

1. Key generation: *generateKeys()*.
2. Signature: *sign(secretKey, message)*.
3. Signature verification: *verify(publicKey, message, signature)* return *Boolean*.

Particularly, each Blockchain user has at least one public key and one private key [242]. Both keys are alphanumeric and stored in the cryptocurrency digital wallet where is independent from the Bitcoin network. Public key is used to generate user's public address and initiate Blockchain transaction. Private key is used to digitally sign and finish off a transaction and proves payer's Bitcoin ownership. Public key and public address are publicly available, but private key is kept confidentially by user. When a payer initiates a transaction to transfer Bitcoin to a payee, payer needs to identify payee's public address first and then initiate the transaction with his own public key. To finish off a transaction, payer has to use his private key to digital sign '... a hash of the previous transaction and the public key of the next owner and adding these to the end of the coin [347].' which in accordance with Bitcoin developer's guide, Bitcoin Blockchain deploys Elliptic Curve Signature Algorithm(ECDSA) with the secp256k1 curve as the fundamental algorithm for digital signatures [64].

To put in more detail, unlike conventional cryptography encryption, which discrete logarithm and integer factorization are normally imposed, ECDSA's distinctive sub-exponential-time algorithm is no one as such yet. Therefore, it is claimed that '...the strength-per-bit is substantially greater in an algorithm that uses elliptic curves [257].' Specifically, secp256k1 curve [313] is a specification of recommended 256-bit elliptic curve domain parameters in finite field F_p , which include six tuple $T = (p, a, b, G, n, h)$. That is, element p defines the finite field, a and b defines the curve as [81]:

$$E : y^2 = x^3 + ax + b \tag{5.2}$$

For secp256k1 curve particular, $a = 0$ and $b = 7$, and G is the base generate point, in a compressed form is [81], [257]:

$$\begin{aligned}
 G = & 02\ 79BE667E\ F9DCBBAC\ 55A06295 \\
 & CE870B07\ 029BFCDB\ 2DCE28D9 \\
 & 59F2815B\ 16F81798
 \end{aligned} \tag{5.3}$$

And the order of G is $n = FFFFFFFF\ FFFFFFFF\ FFFFFFFF\ FFFFFFFF\ FFFFFFFF\ E\ BAAEDCE6\ AF48A03B\ BFD25E8C\ D0364141$ and G 's cofactor is $h = 01$. After the elliptic curve's parameters are fixed, a secp256k1 curve is defined [81]. The general simplified formula of a digital signature is:

$$\begin{aligned}
 \text{Signature} = & \text{SignAlgorithm}\{ \\
 & \text{Hash}(\text{previous transaction}, \text{payee public key}), \text{payer private key}\}
 \end{aligned} \tag{5.4}$$

To prove payer's transaction is valid, nodes validate transactions by signature verification algorithm which can prove the signature attached at end of the transaction can only be created by payer himself. The private key is pure random but the public key is calculated by elliptic curve. elliptic curve allows smaller keys for the same security compared with non-elliptic curve cryptography algorithms [242]. Cryptography public-key is based on the concept of mathematics intractability problem, which is unsolvable. Most important, elliptic curve digital signature algorithm (ECDSA) security is based on the hardness of factoring and discrete logarithms; therefore, they are suffer the broken risk from quantum computing [11], [33], [85].

5.2.2 Consensus rule

In Blockchain system, all nodes work independently as in a coherent system, and all independent nodes work for one common goal [280]. Specifically, Blockchain miners store confirmed and validated blocks into their own blocks only after they have been validating that block by themselves. When different nodes have exactly the same block in their own block chain, they reach a consensus [64]. To put in more detail, the rule that all nodes use to maintain the network consensus is called consensus rule [347]. Consensus rule indeed does not only specifically define how blocks and transactions are validated but also defines a specific workflow about how to reach an agreement on what an updated state of the system is [352]. Plus, since transaction is initially hashed into Merkle tree, and then hashed again into block hash, transaction format becomes part of the consensus rule [68]. Most important, consensus rule sets the rule for how the system network is constructed; therefore, the system security is closely

linked with it [104], which maintains the network security against non-equivocation problems, threats and attacks etc [355].

The core function of a consensus rule has been defined as in [502], which outlines an explicit and guaranteed ordering of transaction and block validation. To put in more detail, a consensus rule must explicitly defines [502]:

- Defines the correctness of all transactions.
- Define agreements on system global state which include the order and correctness of transaction execution outputs.
- Interfaces and dependencies on smart contract layer to define how an ordered transaction set in one block is verified and confirmed.

Consensus has properties, which are safety and liveness [502]. That is,

- Safety. For the same input, all different nodes will give exactly the same output as long as the same consensus rule is used. That is, the consensus rule is required to behave identical to all nodes in the system, and all nodes will end up with the same system state after a number of transaction being executed.
- Liveness. Assuming communication in the Blockchain network is always successful, all normal node without any faulty is able to receive all transactions that are broadcast by miners.

Consensus has interaction with other parts of the Blockchain system as well, such as smart contract. Consensus main functionality has be outlined above, which the main activity of a consensus can be concluded as “...ordering of transaction and validating transactions” in Hyperledger Blockchain [502]. Therefore, the initial process of a consensus is always related with receiving the transaction request from client end, and then based on the ordering of the transaction, the consensus will trigger a corresponding transaction [502]. The ordering of transactions specifically collects transaction requests based on consensus rule and system configurations [502]. Since the smart contract is the component in Blockchain that specifies the contract business logic; therefore, to validate a transaction, consensus is heavily dependent on smart contract layer to ensure the transaction conforms to the system policy and the smart contract terms and conditions [502].

It is very common that consensus rules are updated to introduce new features or to prevent network abuse, which a change in the consensus rule may cause the chain fork into different chains [65]. As Blockchain network is maintained by anonymous users, to protect the benefits of honest user and find out who is being honest, consensus rule is define [502]. In the currently literature, there is a very wide range of Blockchain consensus rule. In accordance with survey [73], [104], some consensus rules are dominated the current Blockchain architecture, which include:

- **Proof of work (PoW).** The main idea of a proof-based consensus is to obtain sufficient proof to acquire the right to post a new block in the system [104]. As if there is no cost at all to post a new block into the system, the system may end up with a real chaos that the ledger book can be messed up with false ledgers. To distinguishing honest nodes and dishonest nodes, the proof is vital. That is, the proof is the evidence of distinguishing honest and dishonest nodes. A good proof-based consensus is able to make it impossible for dishonest nodes to mine any new block without the proof, and proof-based consensus is normally deployed in public Blockchain [552]. To put more details in, at the year 2009, decentralized currency Bitcoin was first time implemented with proof-of-work consensus by Satoshi Nakamoto [91]. Proof-of-work consensus revolutionary sorts two things out [347]:
 1. Agreement on system canonical state updates. All individual nodes behave completely independent as in any coherent system, all nodes collectively agree upon the same canonical state updates.
 2. Free entry. Free entry solves 'who has more weights in a system' political problem, by completely removing Conventional system entry barrier like user registration.

Specifically, the work in PoW refers to nodes change nonce value for every try that they have made until a satisfied hash value is found [66]. The work itself is computational trial and error computations, and the proof of the work is reflected in nonce, which is stored in block header [65]. The first node who finds that satisfied hash value will nominated as the creator of that block, and will get rewarded for the work they have been done for maintaining the system consensus [70]. The node then needs to broadcast newly validated block to all other nodes. Upon receiving new state updates, all nodes keep a copy of this new block and add it at the end of their own chain [64]. Nodes give expression to accept a new block by using it as previous hash for the next new block. When multiple simultaneous broadcasts happen, the node who has the longest chain will be voted to be valid. Most important, PoW is one-CPU-one-vote [347].

proof of work is significantly related with '*difficulty*' variable. As discussed in section 4.2.1 transaction, difficulty is represented by nBits in each block header, which is decided by the average time cost for mining one specific block, and the right time should be cost on mining one block is a compromised decision between power energy waste and the proof-of-work that required to secure the system [271].

The drawbacks of proof-of-work consensus is the double-spending attack, which is the sender trying to spend the coins more than once. One straightforward

method of this attack is to obtain more computational power than the honest nodes altogether. Another method is to initiate one reverse transaction at the same time with the output transaction, and try to convince the system the reverse transaction comes before the output payment [82]. By a similar token, proof-of-work cannot work securely if the network does not have enough computational power attach to the system yet or the network is simply with no value [473].

- **Proof of Stake (PoS).** The stake can be user's account balance or the age of the crypto-currency, etc. After block being valid, node who has the most stake will be voted as the block creator and gets reward from the network [194]. That is, to distinguish an honest miner with a dishonest miner, PoS requires the miner holds some crypto-currency of the system; therefore, the person who holds more assets in the system are assumed prone to behave honestly for the sake of the system. Plus, PoS saves massive electric energy compared with PoW [104]; however, PoS mining has nearly zero cost and may cause the most stake node dominates the network. Or by the same token, it may also cause the problem of a node has no any stake at all [134]. In current literature, Ouroboros protocol [277] is one of the most well known proof of stack consensus that is already deployed in Polkadot Blockchain. To put in more detail, through a coin flipping algorithm, a miner is randomly selected [194]. As for the main drawback of this consensus, it is there is no guarantee nor limits about how much stake that a dishonest node could have [104].
- **Prove of Authority (PoA).** Instead of miners racing to find a solution to a difficult problem, authorized signers create new blocks based on some proposed rules [131]. Similar with PoS, PoA directly assigns who can validate transactions and blocks directly, but PoA requires validators identity authentication and a reputation evaluation scheme attach to it [330]. Compare to PoS, PoA is cheaper to implement as there is no request for validator to hold any coins in their accounts at all [58].
- **Prove of Authentication(PoAth).** PoAth [329] is proposed for Blockchain with limited computational resources within its system. That is, in computational resources limited Blockchain systems, it is fairly easy to acquire more computational power than all honest nodes. [388] suggests to use ElGamal asymmetric encryption method to create another private-key pair for block signatures ⁸, but only differ in the block validation process with traditional Blockchain systems. Specifically, normal nodes generate blocks as

⁸e.g. $y = r^x(mods)$, both r and s are publicly known, and x is the private key, y is the corresponding public key.

in conventional methods, but they are required to sign a digital signature after block generation [387]. Trusted nodes within the system are asked to authenticate these signatures for the purpose of block validation, authentication, and broadcasting the corresponding block to the network. PoAth is simulated with five Raspberry Pis consisted Blockchain network with an average of 3.8 seconds for generate and validate one block, which compares with bitcoin network with a time of 10 minutes or so [388].

- **Byzantine Fault Tolerance (BZFT).** Byzantine fault in computing refers to fault presenting different messages to different message receivers, which is a loss of any distributed system that system consensus is compulsory [501]. Therefore, Byzantine fault tolerance is the mechanism of being able to defend against such failure of a system and coherently reach an agreement among honest nodes which such an agreement is required to maintain system stability and security [98]. Specifically, suppose n nodes exist in a system, t of them are dishonest, and assume nodes are peer-to-peer connected. Further assuming one node x tries to broadcast a message of a , the rest of nodes can communicate with each other and verify x posted value of a , and the system will be finalized at an eventual value a' . A good practise of Byzantine fault tolerance scheme can be [558]:

1. if x is honest, then all the rest nodes will agree on the value of x eventually. That is, $a = a'$.
2. if x is dishonest, all honest nodes will discard x proposal and will agree on the value of a' eventually. That is, $a \neq a'$.

For a short summary, if there are n dishonest nodes existing in a system who are sending inconsistent messages to other nodes, a minimum of $3n + 1$ nodes are required to exist in the system to combat dishonest nodes and maintain system correctness, or $3n$ with a digital signature [376].

Another major concern in BZFT is that it only verifies the consistency of a message that is communicated and exchanged by all nodes, but does not encompass the correctness of the message itself. If a node deliberately sends consistent wrong messages to other nodes, this kind fault will not be caught by BZFT [558].

The quality of network (networking) infrastructure affects the performance of the distributed network, and the performance and soundness of a consensus rule can be measured by below variables [73]:

- Transaction rate.
- Scalability.

- Adversary tolerance.
- Energy consumption.
- Blockchain type.

Mining Time

In BTC Blockchain, with proof-of-work consensus rule, mining time is roughly pre-defined as a constraint parameter which is closely related with transaction time [549]. That is, every block in BTC Blockchain is mined roughly about ten minutes⁹ [271]. The ten minutes minting time is a system predefined parameter by Mr. Satoshi, which is a compromised decision between power energy cost, system security, and system performance.

To clarify, the time cost for mining a new block is limited by three things:

1. **Consensus rule.** Consensus rule defines how a transaction should be validated, and also how a block should be mined. In proof-of-work consensus rule, a block is mined by hashing data below the nBits target, and a transaction is validated if any miner put that transaction into a new block and successfully mined the block; therefore, the time cost for mining a new block is mostly decided by the hashing power the miner has [120]. In generally, the more computational power a miner has, the quicker a new block can be mined [347]. Whereas as a contrast, for proof of stake consensus rule, the time cost in mining a new block is determined by a combination of brute luck and computational power [277].
2. **Network performance.** Network performance has a big impact on mining time as well, such as network bandwidth and capacity which significantly affect the data transfer speed and availability. The network bandwidth of the whole system determines how quick new block can be broadcast to the rest of nodes and therefore, confirmation latency affects user experiences transaction has to queue to be accepted by miner to valid it [73].
3. **Block size.** How long it costs to mine a block is decided by the block size. That is, the block size have an impact on throughput to allow enough time¹⁰ for enough node be fully aware of a new block is mined before another new block is mined is critical to reduce fork possibility [554]. The larger block size, normally will result in longer mining time in the same Blockchain system [67].

⁹To make sure that each block mining time is ten minutes constantly, every 2 weeks when roughly 2016 new blocks will be mined, BTC protocol will adjust nBits target difficult to make adjustments [65]

¹⁰The time here specifically refers to the gap time between two successive blocks.

Reducing the mining time will result in [28], [67]:

- Transaction validation time decrease. Reduce the risk of zero confirmation transaction.
- Payout variance decrease. As the difficulty is reduced, it does not require the same computational power; therefore, mining pools where have larger computational power is no difference compared with small mining pools.
- Less energy waste. By reducing the mining time of each block, the first transaction validation time is shortened as well. As the result of that, less computations is required which leads to energy consumption goes down as well.

In accordance with [119], at the ten minutes interval, the probability of a new block can be mined is about 63%. If the time interval is 30 minutes or 60 minutes, that opportunity goes up to 95% and 99.7% respectively. That is, shorter block mining time will significantly increase the risk of forking, which will definitely be harmful to the system security.

5.2.3 Wallets

Wallets are either a program or a file, which mainly enable a user to receive and send crypto-currencies[62], [260], and wallets optionally store related transaction history of a user. In accordance with [62], a digital wallet in Blockchain system has to take care of three necessary but separable parts, which include:

1. Key generation and distribution. In accordance with different key generation algorithms, Blockchain private- and public-key pair is always stored and generated in digital wallets. Plus, storing the root seed is also always recommended.
2. Signing program. Wallet hold the private key for users; therefore, wallet has to take care of the signature process to finish off a transaction that user initiated.
3. A networked program. Wallets also needs to take care of ‘broadcasting’. If any new block is mined, wallets take the responsibility to make sure all nodes know about the new block. Therefore, wallets have to communicate with the peer-to-peer network.

To put in more detail, Comparing with traditional digital wallet, crypto-wallets can be generally divided into either cold wallets or hot wallets [259]. That is, a hot wallet is directly connected with internet; however, a cold wallet is not. In accordance with service a wallet supply, crypto-wallets can be classified into full service wallet, sign-only wallet, distributing wallet etc [62]. That is,

- **Full service wallet.** Full service wallet offers service of generating private key, distribute public key whenever is necessary, monitor all spending transactions to the public key, and broadcasts the validated transactions to the rest of nodes in the network. Therefore, it is fair to claim the distinguishing benefit of using a full-service wallet is that it offers all-in-one service, which is easy to use and convenient. However, the obvious drawback of a full-service wallet is that it stores user's private key on a device that is connected with internet, which exposes it to the spot of malicious attack.
- **Sign only wallet.** To add more security elements to private key, signing-only wallets are designed to preserve the private key more profoundly. That is, sign only wallet is separated with key generation wallet and network wallets but only make digital signatures to sign off a transaction for user. To put in more detail,
- **Distributing wallet.** Distributing wallet is designed only distributing public keys but nothing more, which aims to enable secured transactions in insecure environment like web-servers.

As for the format of keys in Blockchain system, private key format in Bitcoin Blockchain wallets is constraint to 256-bit number. To improve the security of private keys, hierarchical deterministic key and hardened keys mechanisms are recommended for extra security [62]. That is, a conventional private- and public-key pair is generated by following the below formula:

$$\text{point}(\text{pri}_{key}) == \text{pub}_{key} \quad (5.5)$$

Where ' $\text{point}()$ ' is the ECDSA key generation function, and pri_{key} is the randomly generated large integer. Distinctively, hierarchical deterministic key follows the below formula

$$\text{point}((\text{parent}_{\text{prikey}} + i)/p) == \text{parent}_{\text{pubkey}} + \text{point}(i) \quad (5.6)$$

Which the most common option for variable 'p' is a global constant number that is set by all Bitcoin software. That is, child's public key can be generated by combining parent's public key and any $\text{point}(i)$, which makes receiving payments to the same wallet by different public addresses possible [62]. However, the major concern in hierarchical deterministic key is if one parent public key is found, the rest of child's keys are all deductible by brute-force trails as they are all come from the same parent key and constant p is a known value [62].

5.3 Bitcoin, Ethereum, and Hyperledger

Blockchain system is evolving. Since the very beginning of when Blockchain is deployed into real practise in 2009, the system is getting more and more robust in the regard of anti-malicious attack, capability and flexibility. Now, Blockchain is not only widely used in financial area, but also healthcare [47], logistics [209], media(Water & Music.io), and gaming(UFORIKA.io) etc. In accordance with different protocol and architecture design, Blockchain functionality and capability are also drastically different. In this section, Blockchain system evolution: 1.0, 2.0 and 3.0 will be introduced, compared and contrasted.

Turing Completeness

In accordance with [228], Turing completeness is used to describe data manipulation rule of a system. That is, if a system is Turing-complete, it means the system data manipulation rule is able to be ‘decidable’ when it communicates with other system’s data manipulation rule. To put in more detail, a data manipulation rule is ‘decidable’ when the computing problem is solvable by using the data manipulation rule of the system [228].

The language that is used to compile smart contract is vital, as it has a decisive impact on the security of the smart contract, and therefore, has an impact on the security of the system as well[252]. Even though [252] has found that most smart contract in current market do not require Turing completeness at all, Turing completeness language compiled smart contracts enable more complex business procedures. Indeed, the drawbacks of Turing incompleteness is not only limited business procedures but also the impossibility of controlling complex automated tasks [528].

5.3.1 Bitcoin

Bitcoin as the very first Blockchain system application, it is particularly designed for secured transaction in distributed system. To secure the system, Bitcoin smart contract “...can often be crafted to *minimize dependency* on outside agents, such as the court system, which significantly decreases the risk of dealing with unknown entities in *financial transactions* [64].” Therefore, the style of Bitcoin smart contract is very limited.

In accordance with [68], Bitcoin smart contract style include:

1. Escrow and arbitration.
2. Micropayment channel.

3. CoinJoin Style.

Bitcoin has proven to be Turing-incomplete as its scripting language does not support loops, which is unable to execute any algorithms that require loops but only linear and tree-like instructions [528]. By avoiding infinite loops, it effectively protects the system against denial-of-service attack. Plus, Turing-incompleteness was a significant constraint that limits Bitcoin Blockchain applications scope to outside finance service; however, [528] uses parallel oracles to Bitcoin Blockchain network, which successfully enable loop constructs in the combined system.

From 2009 until the present time, Bitcoin is very persistent with proof of work consensus rule. Even though it is proved to cause massive energy waste, it keeps the system very robust and secure. In 2017, Bitcoin suffered massive usage decline that its market share dropped from 95% to 40% as a result of its system usability issue. That is, between 2016 and 2017, Bitcoin system cannot reach a consensus about increasing the network capability; therefore, the transaction fee becomes very expensive and the Bitcoin becomes unreliable [71]. In August 2017, Bitcoin had a hard fork to Bitcoin Cash ¹¹, and a soft fork for segregated witness backward-compatible consensus rule updates [315]. Both fork are aiming to increase Bitcoin network capability. Now, Block size is increased to 32MB as a maximum rather than 1MB [315], which leads to the throughput goes up to 150-200 per second from 3-7 per second [63].

5.3.2 Ethereum

The initial publication of Bitcoin Blockchain was 2009 [347], very soon after it another two major Blockchain systems are published as well, which are Ethereum Blockchain in 2014 [164] and Hyperledger Blockchain in 2015 [26]. Compared with Bitcoin Blockchain, Ethereum successfully enables both Turing-complete infinite loops, proof of stake consensus, and state machine enquiry, and Hyperledger is focus on permissioned Blockchain to construct a distributed and shared database with trusted and permitted entities for enterprise applications.

Ethereum went live in 2014 when is roughly five years after Bitcoin Blockchain was published. Three major modifications are made by Ethereum compared with Bitcoin Blockchain. That is [507],

1. **Turing-completeness.** The data manipulation rule in Ethereum is Turing-complete, which means looping is feasible. This major change makes Ethereum smart contract is able to support and complete more complex tasks [508], [518]. In order to prevent accidental or hostile infinite loops or other computational wastage in code, each transaction is required to set a limit to how many

¹¹This hard fork starts at Bitcoin block 478558.

computational steps of code execution it can use [318]. A computational step costs 1 gas¹², but some operations cost higher amounts of gas because they are more computationally expensive, or increase the amount of data that must be stored as part of the state. There is also a fee of 5 gas for every byte in the transaction data [293]. To put it clear, “...the intention of the fee system is to require an attacker to pay proportionately for every resource that they consume, including computation, bandwidth and storage; hence, any transaction that leads to the network consuming is a greater amount of any of these resources must have a gas fee roughly proportional to the increment [483].”

2. **State machine.** Ethereum Blockchain is more than a public ledger, which specifically Ethereum, Blockchain can be “...a distributed state machine [167].” That is, instead of accounts and accounts balance, Ethereum has a machine state which carries a way larger data structure. To put more details in, the machine state refers to the capability of making the system state data enquiry at block level, and execute codes in Ethereum virtual machine environment directly rather than smart contracts only [167].
3. **Proof of stake consensus rule.** In September 2022, Ethereum switched from proof of work consensus rule to proof of stake. That is, instead of requesting to use electricity and hashing power as a proof for honest nodes, Ethereum ask nodes 32 ETH as collateral to behave honestly, which the collateral can be destroyed if dishonest behaviour is detected [169]. A new block validator is proposed directly by block proposer; therefore, it significantly speeds up and improves the transaction throughput [165]. Thanks to this modification, even though the new block speed does not change very much ¹³, the energy was significantly saved by about 99.95% [170].

It has mentioned above that Ethereum data manipulation rule is Turing-complete, which seems to expose Ethereum at the spot for denial of service and infinite loops attacks. However, Ethereum tackles this issue with imposing “gas”. Therefore, for initiating a infinite loops attack, it requires to pay an infinite number of gas fee, which makes the attack not worth doing at all [168].

5.3.3 Hyperledger

Founded in 2015, Hyperledger is targeting commercial and enterprise applications which in most cases are either private permissioned or consortium Blockchain [26].

¹²Gas is the name of fee that is charged for computational steps in Ethereum transaction, which in a way efficiently limits the number of computational steps of a transaction can take[168].

¹³Ethereum new block was mined about every 13.3 seconds before the merge, and it is about 12 second per new block after the merge [170].

Therefore, the proof of work consensus rule is not suitable for Hyperledger as both Bitcoin Blockchain and Ethereum Blockchain are public Blockchains. That is, prove of authority and Byzantine fault tolerance is more applicable to Hyperledger especially when private Blockchain does not have enough computational power to initiate secure transactions at the very beginning [502]. Hyperledger is one and only established private permissioned blockchain system in the current literature with more commercial favourable innate features [477]. For short, Hyperledger aims for providing a better and direct connection between commercial organizations and individuals [26].

Comparing to Bitcoin and Ethereum Blockchain, Hyperledger effectively constrains who can access the Hyperledger network, and the purpose of constructing a private or consortium Blockchain is to efficiently share data with trusted entities only. That is, different from a public Blockchain that everyone can interact with and access to it, Hyperledger permissioned Blockchain only allow permitted user or permitted entity to access to and interact with it [477]. The main benefits of that are two folds. That is:

1. A shared database system can be constructed among all trusted entities or organizations. Especially in the cross-industry commercial applications, it is more than efficiency but also opportunity-catching, which really makes the most use of the Blockchain system distributed feature. Plus, permission Blockchain can establish a channel for private and confidential data sharing. Like for instance, Hyperledger permissioned Blockchain application in healthcare can construct a private and confidential data sharing channel between doctors and their verified patients [22], [390].
2. It establishes an enhanced trust model among all involved commercial participants through Hyperledger smart contracts. That is, in traditional business model, human factor is always the main causation of business failure. Hyperledger smart contract as in all Blockchain smart contract which automatically execute the contract whenever the contract conditions are realized without the request of any human intervention at all [508].

The significant improvement in Hyperledger Blockchain is some canonical consensus rules in permissioned Blockchain. Compared with a public Blockchain, a private permissioned Blockchain normally has very limited computational resource, which makes the consensus rule even more important in the regard of maintaining the system security [502]. Particularly, Hyperledger is designed for partial trust which only trusted entity can take control over the system; therefore, proof of work consensus with anonymous miners and public access is not suitable as it is too time consuming and energy demanding. Instead, voting and lottery based consensus rule does the job for Hyperledger Blockchain. For example, Hyperledger Indy is designed for self-sovereign digital identity management system, which adopts voting based redundant

byzantine fault tolerance (RBFT) consensus rule [38], and Hyperledger Sawtooth Blockchain is particularly for logistic service is constructed based on lottery based proof of elapsed time(PoET) consensus rule [502].

5.4 Blockchain Major Issue and Threats

5.4.1 Major issue

Blockchain system as introduced has some favourable features, which is immutable, trust-less, and decentralization [128]. However, as one coin has two sides, there are some issues to be tackled in Blockchain system. The major issues in Blockchain system are two folds. That is,

1. **Scalability.** [298], [458] It has been discussed in section 4.1 that Blockchain system scalability has three dimensions, which include

- (a) Size.
- (b) Geographical.
- (c) Administrative.

and the reasons of why Blockchain system suffers from scalability are the consensus rule of the system, network performance ¹⁴, and block size [173]. To put in more detail, block size decides how many transactions can be recorded in each block, and therefore, the larger the block is, the easier the system scale up and bigger transaction throughput will be. However, large block size will take longer time to broadcast to it the rest of nodes after mining. By a similar token, shorter the block interval will cause the chain to fork much easier which put the coin at risk for double-spending [28], [67], [173]. Therefore, a good choice of both block size and block interval become a compromised decision between system security and scalability.

2. **Transaction tractability.** Blockchain user address is pseudonymous but not anonymous; therefore, IP address and user Blockchain address can be linked especially the user uses the same Blockchain address a lot of times. It causes user transaction footprint tractable, which is not desirable [198]. Plus, due to the protocol of the Blockchain system, the majority of the transactions details are public available information, which could significantly violate user privacy [55], [511].

¹⁴The Blockchain consensus rule and network performance altogether can be summarised as block interval; Therefore, factors that have decisive impact on the Blockchain scalability is the block size and block interval [173].

To tackle with Blockchain scalability issue, sharding [113], multi-chain [465], consensus rule modification [166], and lightning network [272] etc are introduced. First edition Bitcoin Blockchain only process 3 to 7 transaction per second but now the highest throughput Blockchain Ripple now can process 1500 transactions per second, which is compatible with Paypal and Visa [63], [405]. By similar token, to solve selfish miner issue, ZeroBlock [456] proposes a regime that is new block has to be accepted within a time interval to get the reward, and fresher new block has priority to be voted. To secure transaction privacy, mixing and zero-knowledge anonymous [183] techniques can be deployed.

Apart from both major issues being discussed here, there are some concerns as well. That is,

- **Centralization.** Blockchain is a classical decentralized system; however, when decentralized users construct one single major group, it causes a serious centralization issue. Like for instance, if miner A is a mining pool with 30% hash power and B has 10% hash power, A will have a risk of producing a stale block 70% of the time (since the other 30% of the time A produced the last block and so will get mining data immediately) whereas B will have a risk of producing a stale block 90% of the time. Thus, if the block interval is short enough for the stale rate to be high, A will be substantially more efficient simply by virtue of its size [91].
- **Consensus Diverges.** The main challenge in decentralized system is the distributed system consensus rule. As users are decentralized and distributed connected in the system; therefore, all users have to be agreed with which system status is trust-worthy. Indeed, a sound decentralized system requires all decentralized users have exactly the same system status. Unfortunately, due to the fact that all users are decentralized, there is a large opportunity for the consensus diverges at the point of there is an excessive computational power group formed, which can cause the whole system being modified completely [104], [352].
- **Equivocation.** Equivocation is an essential issue in all distributed system protocol design, which refers to dishonest participants behave conflicting with the system protocol statements and cause system errors or attacks [305]. Byzantine-like faults and double-spending [98] in Blockchain, and multiple certificates in public key infrastructure [305] are two typical examples of equivocation issue consequence in distributed system.
- **Selfish miner.** Miner is another name for full nodes who maintain Blockchain consensus rule because normally they will receive reward from the network like

gold miner get gold for the mining work that they have been done [225]. Selfish miner refers to nodes who successfully create a block without broadcasting it to other miners and users. The benefit of doing that is selfish miner will get a longer chain than other miners and therefore will get more chance to be voted as block creator and get the reward from the network [537]. The main point of selfish miner is not just they can constantly get reward but indeed the worst case is they are dominating the network completely [13].

5.4.2 Security threats

It has been discussed that the main security concern in Blockchain system is the private key security and malware [534]. These two major threats hinder the generalization of decentralized system, and makes it harder to be deployed in a large scale. To put in more detail about these threats, they are summarised as:

- **Sybil attack** Attacker subverts a reputation system of network service by creating a large number of pseudonymous identities and uses them to gain a disproportionately large influence [151]. Reputation system is user's trust will be built on through rating each other in the online communities. For example, eBay, amazon, stock exchange online advice systems, even though the 'proof of work' consensus can prevent Sybil attackers from re-do the whole Blockchain in their own favour. Current time of writing is 2 power of 187, the network must make an average of 2 power of 69 tries before a valid block is found [64], [91]. Every 2016 block, this target nonce will be adjusted again so that in the network the new block is produced every ten minutes [534].
- **Greedy Heaviest Observed Subtree(GHOST)** Modified GHOST Implementation. That is, Blockchain with fast confirmation times currently suffer from reduced security due to a high stale rate – because blocks take a certain time to propagate through the network, if miner A mines a block and then miner B happens to mine another block before miner A's block propagates to B; miner B's block will end up wasted and will not contribute to network security [91], [534].
- **Malicious attack** Execute unauthorized actions on the victim's system: as cryptographic protection on Blockchain is strong, malicious attacker can only target the part without cryptography in Blockchain. That is, the order of the transaction:
 1. Send some Bitcoin coin to a merchant in exchange for some product or service.
 2. Delivery of the product or the service.

3. Make up another transaction sending the same Bitcoin coin to himself.
4. Convincing the network that the transaction to himself was the one that came first, so that transaction should be accepted and validated by the system.

Apart from above well-known attacks being discussed, 51% attack is also very brutal [431]. That is, if malicious nodes occupy no less than 51% of the whole system's hashing power, malicious nodes can fork from the main chain and confirm malicious transactions at their own will [28]. However, 51% attack will cause massive attack cost, especially in public Blockchain which the hashing power is immense.

5.5 Chapter Summary

In this chapter. Blockchain is well researched. Starting from **Chapter 5.1 Blockchain System Overview**, the concept of public distributed ledger, blocks in Blockchain, chains in Blockchain, and pros and cons are introduced beforehand so that a general concept and property of Blockchain is established. Further discussion in the subsection of decentralized distributed system, DAO, and Microservice are reviewed to offer an idea about what Blockchain system characteristics and functionalities. In **Chapter 5.2 Key Components in Blockchain**, transactions, consensus rule, wallets are explored. Specifically, smart contract as a subsection in transaction is also well reviewed. Chapter 5.3 Bitcoin, Ethereum and Hyperledger is a comparison between Blockchain system 1.0, 2.0 and 3.0. Particularly, the main concept of Turing completeness is explained, and each system's outstanding features are discussed. Finally in **Chapter 5.4 Blockchain Major Issue and Threats** are discussed.

Chapter 6

Cloud, Edge and Fog Computing

6.1 Internet of Thing(IoT)

The invention of internet blurs the concept of distance, and completely changes the way of communication and business model. The widespread of such internet applications are inseparable with the development of IoT. To be more precise, internet of things connect computing devices or objects to the internet, which breaks geographical restrictions and makes remote control possible [300].

Remote control indeed is a very demanding and challenging task, which is required to collect real-time in-situ data so that control action can be taken accordingly. In IoT ecosystem, control behaviour can be fulfilled by either human or computer. In the case of human takes the control task, IoT is only an information recorder or transmitter. It replaces real human to work at the site, like CCTV [105]. Internet transfers collected information from IoT device to a person so that he can take control actions as responses. In the other scenario where IoT devices can automatically complete the control task, IoT has to be proper programmed to ensure direct device-to-device communication possible [448]. In this case, IoT devices has some programmed artificial intelligence(AI) so that automatic decisions can made [496].

Normally, AI IoT are managed by an intelligent user interface(UI), which the UI itself can be either a specialized application or a web-page. It gathers all real time information from all connected devices and is able to automatically send control order to corresponding devices in the way it being programmed. A good example of that is smart homes, multiple devices can directly communicate with each other to adjust home appliance settings through user interface so that home is more comfortable and enjoyable to live [367], [422], [427].

For a formal definition of internet of things, which is defined by the internet society that “...*network connectivity and computing capability extends to objects, sensors and everyday items not normally considered computers, allowing these devices to generate,*

exchange and consume data with minimal human intervention[409].” Therefore, the digital twin IoT is the IoT mapped to the digital world already.

To put more details in, it seems that IoT is already an indispensable part of our daily life. For example, mobile phone, monitor camera(CCTV), vehicle, display monitor, home appliance, and robotics etc. they are all good examples of IoT. Particularly, mobile phone is supported by cellular network, home appliances are supported local area network (LAN), and vehicle communication are usually supported by wireless network. The connectivity of all these heterogeneous devices and networks reveals the real power of IoT, and that is the seamlessly integration of heterogeneous networks and interoperability of all devices [5], [367].

For another example, in smart city IoT ecosystem, it can count vehicle number on road and control traffic lights accordingly to ease traffic jam. IoT in these cases [79], [207], [391], [546] are required to cooperate with sensors and actuators. Sensor detects the number of vehicles on road and when there is traffic congestion, which transfer physical signal into electrical output. Actuator will be triggered whenever it receives sensor’s electrical output. Actuator indeed is connected with traffic lights so that traffic flow can be adjusted in accordance with specific actuator is being triggered, which actuator turns electrical signal input into physical output.

Another more intelligent application of smart city is real time CCTV and image processing. In [179] and [340] sensors and actuators are not required any more. Monitor cameras with large computational power can fulfil the same task with even better performance. Such as [142], Canada electronic toll collection system uses Optical Character Recognition (OCR) based smart cameras to process real time images to identify vehicle license plates without any on-board units.

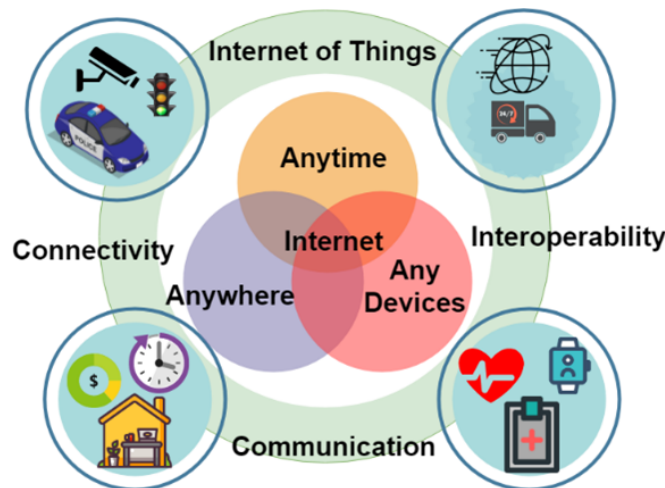


Figure 6.1: A demonstration of IoT features.

6.1.1 IoT features

As it is being introduced, IoT generally refers to all computing devices that connect to internet [367]. Thanks to the consistency of internet service, IoT devices are also acknowledged to have *extraordinary consistent performance* as well. As real human need to take break or accidentally being careless or neglect, IoT devices do not. Even though IoT do have out of power and device broken, it can be effectively avoided by regular check-ups. In general, IoT devices can guarantee 24/7-hour 365-day uninterrupted and consistent work, which is very favourable in remote control.

What is more, since internet now is available at nearly every corner of the world, it makes IoT service available at wherever internet is available. Indeed, IoT applications are already widespread in all walks of our life. Like healthcare, smart home, smart city, industrial manufacturing, and logistics etc. Specifically, milestone IoT application in healthcare can monitor patient condition at 24/7-time frame and alarm in-need emergency treatment before server symptoms happened [390]. Logistic IoT can make real time dynamic trace and track possible [391] and traffic IoT on highway is used to monitor over speed limit vehicles to take traffic control [453], [550].

To be more precise, talking about all these IoT applications and devices, it should be emphasized that the main characteristics of IoT technology include (see **Figure 6.1** for IoT features demonstration.):

- **Heterogeneous Communications.** The seamlessly integration of heterogeneous networks enable communications among heterogeneous IoT devices. That is, the fulfilment of that makes direct communication between heterogeneous IoT devices possible. No matter how fragile the IoT devices are being placed, data can always be collected and transferred to all requests nodes[5].
- **Connectivity.** IoT connects all possible devices which have an access to the internet. Computational power and storage capability can be attached to all IoT devices. As all devices are connected to internet, computational power and storage can be attached to all devices even though the device itself is originally a non-computing device. Internet turns computational power and storage services into computing resources that are being distributed via Cloud. The connection between devices and internet makes computational power and storage available to all connected devices [189].
- **Interoperability.** IoT operation environment indeed is very fragile. Even though the connection to heterogeneous network is not a problem at all, the design of IoT backend data structure and workflow may have drastic difference. Therefore, imposing standardized procedure in IoT would be able to significantly increase the interoperability.

6.1.2 Pros and Cons

AI indeed is a crucial part in digital twin IoT applications, as AI can solve IoT major shortcomings and at the same time create significant business values. Jiang has argued in [145] and [335] that the capability of IoT devices are indeed not decided by the devices themselves but rather is the computational power embed to the devices. What is more, this claim is even more true if the drawbacks of IoT are considered. That is,

- **Privacy issue.** IoT device has the opportunity to collect private data from end user directly, such as ECG heart beating data from a hospital patient, CCTV camera video for footprint tracking, and mobile phone biometric identity authentication to secure an application access etc. IoT handle private data regularly and how to protect the private data become a major issue. Indeed, privacy issue significantly hinder the adoption of IoT. Even though it is claimed that the privacy issue in IoT surveillance applications is being amplified, the general policy should still encourage to process user private data with cautions [409].
- **Security Concerns.** As IoT devices are connected via Internet, they suffer the same security concerns with internet, such as blackmail, vicious attack, and personal security information leakage etc. IoT data are collected from all devices and transmitted through internet [533]. Therefore, the security of each individual device can be an entry point for attacks. Plus, internet data are easily tampered so IoT records can be altered, which raises a significant security concern that is data collected from IoT cannot be fully trusted.

Indeed, IoT digital twin is hard to meet massive data storage and management requirements as IoT itself is intrinsically design to be ***small, portable, and low power consumption*** [189]. Therefore, when digital twin IoT needs to complete computational intense and latency critical task, it normally requires extra support from third party, like Cloud or data management company.

Many AI based solutions have been suggested to verify or calibrate IoT data veracity. As in [189], real time dynamic data first being collect from IoT devices, and then being matched with pre-stored or pre-registered information in the database by robust AI algorithm. In this way, data veracity and authentication can be verified. Admittedly, AI cannot solve all drawbacks in IoT but it does make a huge difference. Below sections will introduce how to use Edge and Blockchain to create a safer IoT ecosystem [264].

6.2 Cloud

In 2006, Amazon Web Services (AWS) launched the first Cloud storage service, which remarks the birth of public Cloud [141], [284]. Cloud computing draws exclusive attention from both industry and academy, as Cloud provider claims it can dramatically reduce networking cost and improves computing resource efficiency to a remarkable high level [443]. Admittedly, Cloud user does not need to pay infrastructure cost anymore; however, there is still an on-demand service fee to be charged. It is like mobile ‘pay-as-you-go’ bundle, and you only pay for what you have been used. [304] and [205] have investigated if Cloud is cost saving or not, and they both conclude that Cloud does save cost especially when the run-time is short and computational requirement is large. Cloud thoroughly change the concept of computing resources, which essentially brings a new idea of using computing as utility [31].

Cloud computing offers a new regime for getting virtualized computing resources through internet, which is configurable, on-demand, and with minimum management efforts [32]. The National Institute of Standards and Technology (NIST) has defined Cloud with five essential features. Specifically, they are [336]:

1. On-demand self-service.
2. Broad network access.
3. Resource pooling.
4. Rapid elasticity.
5. Measured service.

Technically, Cloud computing is derived from the idea of computing time-sharing. That is, “. . . sharing of a computing resource among many users at the same time by means of multi-programming and multi-tasking [10].” One single user may make inefficient use of a computer when long pause or non-usage happens; however, multi-users share the same computing resources at the same time would dramatically improve the efficiency as non-usage of one user can be replaced by another. This concept was further discussed in [107], [308], [406], which formally bring the idea of user’s computer at premises is only an input-output equipment and computing resource can be purchased in the same way with household purchasing water from utility company. Indeed, after water, electricity, gas and telephony, Cloud is already seen as the fifth utility which has been applied to many scenarios [31], [531].

Computing time-sharing does shed the light of Cloud computing, but they do have intrinsic differences. At the very beginning, computing time-sharing is only available to computing giants like IBM, Honeywell and GE [107], [406]. One reason of that

is computer is very expensive and very slow at that time, and there are no private computer users at all [479]. What is more, computer memory size was too small to allow multi-states concurrently exist in one system [406]. That is, time-sharing users have to take queue to enter the system one by one to use the centralized computing resources. All users only share the time among all users so that computing resources can be consumed and ran all the time [429].

By similar token, Cloud computing is to distribute centralized mega computing resources to all end users directly, which enable multi-states coexistence in one system [284]. This is due to computer memory size is dramatically improved and the invention of hypervisor. Hewlett Packard Enterprise has announced their world's largest single memory computing system has 160 terabytes [505]. To put more in detail, hypervisor [141] is a software that is implemented between computer hardware and computer virtual machines, which can pull computing resources out of computer hardware and distribute the resources to virtual machines. The deployment of both hypervisor and large memory size guarantees the implementation of Cloud computing, which makes data centre mega computing resources can be distributed to millions of private computer users through internet [32].

Cloud has three different types. Specifically, they are public Cloud, private Cloud, and hybrid Cloud [284].

- Public Cloud is open to all public users through public internet; therefore, all the public have access to it.
- Private Cloud is the Cloud infrastructure operated solely for one single tenant, and therefore only permitted users can access to private Cloud.
- Hybrid Cloud is a combination of public and private Cloud, which usually is to put public Cloud infrastructure resources on top of a private Cloud.

Indeed, there are not too much differences in the regard of architecture a public Cloud or a private Cloud, but the security and cost are substantially different [339]. Private Cloud has the highest security, as all access to it can be controlled properly and there is only one tenant in the system. However, it costs the most because there are heavy infrastructure costs and maintenance cost. On the opposite, public Cloud costs the least and has the lowest security. Public Cloud users do not need to pay infrastructure costs at all but only pay on-demand service fee, which makes it especially cost-saving [304]. Public Cloud suffers highest security risk from public internet as the audience are totally unknown. Hackers, intruders and vicious competitors can fairly easy to damage and tamper public Cloud data, which put all users in a vulnerable position [339].

In a more general sense, Cloud can be described as distributing data-centre resources as service to all on-demand users through internet. That is, Cloud provider

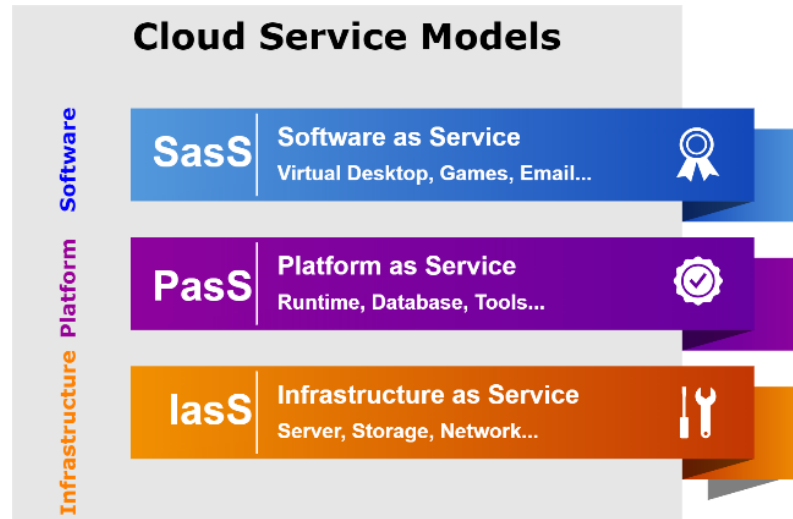


Figure 6.2: A demonstration of Cloud service models.

owns the data centre infrastructure and takes the responsibility to maintain data centre on behalf of all users. Therefore, Cloud users do not need to pay upfront infrastructure cost any more but directly purchase computing resources as utility from Cloud provider. There are three established Cloud service models. The name suffix ‘as-a-service’ indicates that there is no contract between Cloud user and provider, but user uses Cloud as self and on-demand service [197], [339]. See **Figure 6.2** for Cloud service models demonstration.

- **Infrastructure as a service (IaaS)**: IaaS typically provide datacentre infrastructure like server, storage, and networking as services to users. In this model, IaaS provider uses hypervisor pull resources out of bare-metal hardware and then directly allocate these virtualized resources to users. IaaS users can configure the infrastructure as much as they want, but apparently it needs more professional management at user side as well.
- **Platform as a service (PaaS)**: PaaS provider offers platforms to users so that they can develop and manage their own applications directly on the platform. There are many supporting tools and application programming interface (API) available to PaaS users as well. It is easier to use and do not need to manage infrastructure at all; however, it is nearly impossible to move your applications from one PaaS provider to another.
- **Software as a service (SaaS)**: Common example of SaaS are games, email, and virtual desktops like Citrix etc. Users do not have to install it on your machine or manually upgrade the software. SaaS provider will do them all on

user's behalf so it is the easiest service model to use. It is usually charged based on a subscription model rather than one-off fee.

These Cloud service models have two features in common. That is, multi-tenancy and highly scale [339]. When user demand is unpredictable or potentially has big fluctuation, Cloud is able to auto scale up or down the resources assigned to the user. Like AWS Elastic Computing 2 (AWS EC2), user has the option to setup auto scaling and thresholds based on traffic predictions. Once the threshold is past, AWS EC2 creates a new instance and auto run to the load balance pool. When the traffic drops down below the threshold, AWS EC2 will remove that instance server out of the pool. Therefore, no matter how large the traffic is or how intense the computing is required, Cloud can always support the workloads and fulfil the requirements. As for multi-tenancy, cloud provider owns all infrastructure and offer the services to all users. Each user is called a tenant, and all tenants share the same infrastructure. Multi-tenancy is especially favourable when Cloud users need to be differentiated into multiple categories or multiple roles [141].

Considering all these properties that Cloud computing has, Cloud computing is thought for well complementing IoT functionality and creates the most value for IoT system [400]. IoT devices are connected through internet, which guarantees the availability of Cloud computing to all IoT devices. Although IoT device is designed to be portable, small, and low energy consumption, its computational intense applications like AI, machine learning and aggregated big data analysis can be well supported by Cloud. Like traffic cameras in IoT smart city system [145], [278], traffic cameras only capture real time traffic information and it is Cloud does real time aggregated information processing and makes smart traffic control possible.

However, Cloud computing does have flaws. Security is one of them and it is the biggest concern for the adoption of Cloud computing. In accordance with [339], the top three Cloud security concerns are insecure interface and API, data loss and leakage, and hardware failure in data centre. The common causation of the security concern are [339]:

1. Cloud supplier takes full control of user's data. Even though there is service level agreement (SLA) and data usage agreement between Cloud supplier and user, users still may not be able to be fully aware and understand the SLA because of lacking professional knowledge. Therefore, it could be a reason for Cloud supplier to disguise and rationalize their behaviour and sell user data to other vendors for extra money.
2. Cloud service is distributed through internet, which means common internet threats are shared with Cloud system. Cloud data are easy to tamper, modifiable and can be attacked by hackers. These drawbacks stop confidential and private mattered information system from adopting Cloud.

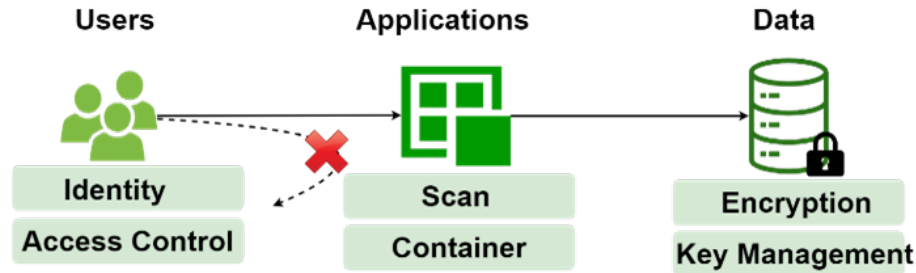


Figure 6.3: Information flow in Cloud environment.

There do exist methods to set up a safer and securer Cloud environment. See **Figure 6.3** for the information flow from users to applications and data storage infrastructure. Security measures can be taken at all levels. In Cloud computing system, access control can be implemented in all Cloud service models. It requires user identity and access can be denied accordingly, which can effectively prevent unknown intruders. Similarly, data can be scanned before sending to application and API so that virus or attacks can be stopped. By the same token, containerization [192], encryption [339] and key management [301] can be used as well.

6.3 Edge

With the widespread of large-scale data analysis and artificial intelligence IoT applications, the demand for Cloud computing substantially goes up. It pushes Cloud provider to build larger data centre in case of the adverse effect on the quality of service and user experience. Besides that, dynamic real time processing IoT system requires low latency computing is also hard to fulfil by Cloud [339]. To tackle the problems that Cloud is not able to sort, which include bandwidth, latency and connectivity, Edge computing comes up [132]. Edge computing does not only deal with all these existing issues in Cloud, but also can decentralize Cloud functionality and manage its traffic in a more flexible and efficient way [308], [429].

Edge indeed is very similar with Cloud. They both provide computing resources as service to multi-tenants through internet. However, the major difference is that Edge normally brings the computing resources physically closer to the place where the data is generated [308].

Cloud directly connects with users and IoT devices, but Edge devices and users have to pass through Edge nodes first and then get to datacentre infrastructure. Edge nodes can perform some of the computation workload and manage traffic. Edge and Cloud computing have very similar pros and cons. However, compared with Cloud, the benefits of having Edge computing are three folds [429]:

1. **Offload datacentre workload.** In Cloud system, user full workload has to be performed by Cloud provider at datacentre, which becomes a heavy burden when demand is exclusive high. Edge nodes can share the workload with other datacentre, which significantly alleviates the intense demand [308].
2. **Low latency.** As Edge nodes locates closer to user and Edge device, data do not have to travel all the way through to the data centre to be processed. Instead, data can be processed faster and more in time at Edge nodes where are physically closer to data source. This feature is very favourable when applications need real time response or dynamic processing as in auto-drive car [340] and real time monitor in traffic system [550].
3. **Securer.** Edge computing has decentralized infrastructure that all Edge devices and nodes have the capability to perform computation. Therefore, private and confidential data do not have to travel far away to Cloud data centre to be processed and stored [367]. It effectively excludes the possibility of information leakage and hacker attack. Plus, Edge computing can effectively avoid single point of failure issue by switching from one data centre to another in seconds.

Typically, there are two common network edges to perform computations in Edge computing [367]. That is, edge device and Edge server. Indeed, the popularization of Edge is indispensable with the implementation of powerful computer processing unit (CPU). CPU gives devices the power to perform computations and process data. Nowadays, each car typically has about 50 CPU, which guarantees car can meet its computing requirements as an Edge device. what is more, Edge device as an equipment with computing power that is usually built for some purposes, like car for driving, mobile phone for phone calls and wearable health device for monitor health conditions etc. They have computational functions build inside of the device itself [306].

In the other case of Edge server, it is an equipment that is used for computing IT workloads, such as a small data centre or a business IT infrastructure. It usually has more computing power compared with Edge devices but physically locate further than Edge device [385].

6.4 Fog Computing

Fog computing has been seen as the highest evolution among Cloud and Edge, which is an extension on top of both [132]. To put more in detail, NIST defines Cloud as “...a model...offers a shared pool of configurable computing resources...[336]”, which essentially turn computing infrastructure into “utility” and can be purchased as on-demand basis [31]. A Cloud provider normally has one mega data centre to

support all users' demand, which both the computing resource and Cloud system are centralized. The initial customer of Cloud utility are organizations such as university and commercial organizations; however, later on Cloud main consumer are individual car owner for GPS and auto-driving, computer game players, and millions of other computing intense, latency critical IoT devices owners. That change triggers the emergent of Edge computing, which is to bring computing infrastructure physically closer to end user through the implementation of Edge nodes or Edge server. By doing that, Cloud evolves to Edge, which is federated data centre, faster bandwidth, low latency, and high computing performance [264].

Fog computing is to provide computing resources even closer to end user in a distributed manner. That is, instead of requesting a mega data centre to distribute its computing resources to individual users, Fog computing constructs a layer to bridge individual users and all distributed computing resources providers, which fully fulfil the gap between Cloud and IoT [132]. Fog computing aims for making improvements in interaction and integration between IoT and Cloud. To be more precise, Edge computing process data at Edge server or Edge nodes, but Fog computing is able to pre-process data at IoT device first for small, non-latency critical, computing un-intense task first, and then send partially processed data to Edge server or Edge nodes for further process. A good example of that is sensor and actuators. Both sensor and actuator process first hand data at their own CPU first and then pushes pre-processed data to Edge server for further processing [360].

In accordance with [132], two distinctive features of Fog computing are:

1. **Distributed.** Aiming to conquer the drawback of Cloud centralized architecture, Fog is designed to be distributed system, which connects all possible devices to make the most use of available computing resources.
2. **Anywhere.** Fog can exist in *anywhere* between IoT and Cloud infrastructure [132].

The advantages of imposing Fog rather than Cloud has been deeply researched, which can be summarised as security, cognition, agility, latency, and efficiency [108], [109], [132].

Most important, Fog computing normally is content aware, which is able to support fine-graded data workload offloading and processing [360].

6.5 Chapter Summary

In this chapter, it aims to explain and understand how Fog computing enables much lower latency response in computational intense tasks compared with Cloud and Edge computing. Therefore, in **Chapter 6.1 Internet of Things** IoT definitions, IoT

features, and pros and cons of an IoT are all well discussed. After that, in **Chapter 6.2 Cloud Chapter 6.3 Edge Chapter 6.4 Fog Computing**, Cloud, Edge and Fog computing are discussed independently.

Chapter 7

Privacy-aware Biometric Blockchain-based e-Passport System for Automatic Border Control

In this section, privacy-aware biometric Blockchain(BBC) based e-passport system for automatic border control(ABC) solution is proposed. To put more in detail, the proposed e-passport is a digital verifiable biometric identity credential, which is not only an equivalent of tangible passport book but also enjoys the convenience of internet communication. That is to say, it can be used for biometric identity authentication over open internet with strong assurance. Most important, due to a sharp increase in border-crossing and shortage in border body guards during Covid-19 pandemic and recent war affair, border checkpoints are always jams with long queue of lorries, so does the airport passengers. It indicates that current border e-gate with e-passport book embedded microchip cannot meet border crossing demand any more. Therefore, the proposed new e-passport is designed particularly for boosting border clearance speed, which aims to make fully automatic border control solution possible through remote monitor over Fog, digital twin, and Blockchain technology.

7.1 System Overview

To start with, it has to be emphasized that automatic remote border control solution is computational intense and latency critical task; therefore, Fog computing is adopted to speed up and offload the whole workload as much as possible. That is to say, instead of pushing all workload in one centralized data centre(Cloud) or offload some workload to Edge where the data centre is closer to end user, Fog computing takes end

user device and IoT computational power into account, which allow the whole process to be decentralized distributed into small pieces in accordance with the computational power of that device. By doing that, two benefits are obtained:

1. **Resources friendly.** Nowadays IoT's computational power will definitely shock your mind. Taking smart phone as an example, rear camera generally has a resolution of 8K(7680 x 4320) compared with a normal ID photo requirement of 600 x 750 ¹. New Android mobile with Qualcomm Snapdragon 8 Gen 2 processor has a clock rate ² of 3.2GHz for 64 bit applications and 8MB cache [357], compared with HP ProBook 450 that has Intel Core i5 processor but only 2.4GHz clock rate for 64 bit applications and 8MB cache [356]. Therefore, taking user end device into account for ABC is to take the most usage of resources, and saves border agency infrastructure input costs at the same time.
2. **Efficiency.** As discussed above, user's mobile phone's processing power is no less than a ProBook, which allows end user to accomplish the vast majority of the border clearance workload by himself through smart phone applications. Fog computing offloads most of the workloads to IoTs, which leaves as little as possible to be finished at the border checkpoints so that the whole process can be very efficiently and fast accomplished.

To add on, the proposed system consists of two separate Blockchain: one is for e-passport privacy-preserving biometric Blockchain based digital verifiable credential, and the other is the automatic border control Metaverse DAO. They both are built upon private Hyperledger Fabric Blockchain, which aims for obtaining more flexibility and extensibility. Plus, private Hyperledger Blockchain can save identity authentication cost significantly, and it is very easy and cheap to scale up. Most important, private Blockchain can easily implement private data access control. By simply changing "modifier" in smart contract, adding 'collection' terms, and creating 'channels', privacy on private Blockchain can be well preserved.

What is more, the e-passport digital verifiable credential Blockchain is named as biometric Blockchain virtual identity(BBCVID). The BBCVID is designed particularly for generating the proposed e-passport and other border crossing required verifiable credentials(and/or documentations). BBCVID immutably records it on the Blockchain, which significantly improves the risk-based digital identity authentication assurance. The proposed e-passport is formatted as .json file which follows W3C digital identity credential data model standard [121] to strength its interoperability.

By similar token, the automatic border control solution is named as virtual stamping(VS) Blockchain. It combines BBCVID, road surveillance system, Fog

¹UK passport digital photo requirement from UK Passport Office [202].

²Clock rate is an indicator of processor's speed. The higher clock rate, the faster a process is.

computing, e-gates, smart contracts, and user end device to efficiently and automatically record every border crossing events as a Metaverse DAO, which aims for improving the border crossing workload efficiency to a new high level. Most important, VS is not only suitable for passenger border crossing at airports but also suitable for vehicles(both private and commercial vehicle) border crossing at land border checkpoints. Therefore, to improve data processing time and capability, Fog computing is used to collect data from road IoTs such as:

1. **Road cameras.** Road cameras read vehicle number plate so that the border crossing vehicle's identity can be identified by computing deep learning algorithms.
2. **Road sensors.** For vehicle border crossing, clearing customs duty is crucial. Learning from Norway-Sweden and UK high-tech highway border settings [203], [282], seals that carry Customs declared products without loading, which saves massive time. Seal has to meet customs regulations to be qualified as customs seal[322]. Plus, weight sensor can be used to roughly weight pass through vehicle so that it can be used as a criteria to eliminate smuggling risk.
3. **E-Gates.** E-gate booth-like access control is deemed necessary as the proposed system is only able to effectively identify traveller who has the legitimate reason to cross the border; however, travellers who *has not* the legitimate reason to cross the border is not able to be identified at all. To improve the border control workload efficiency, e-Gate uses quick response(QR) code to fetch fingerprint template from VS, and does fingerprint identity authentication for border crossing legitimacy verification.

7.2 BBCVID for e-Passport

The proposed privacy-aware biometric Blockchain based e-passport is a non-tangible legal identity credential that is endorsed by government organization. It is machine readable, third-party verifiable, and remotely verifiable, which is deemed as a digital twin of the real-world e-passport. As an IdM system, we will define IdM four major components first. That is, the repository, key authentication centre, policy control, and auditing. After that, main entities in the system, the specific system architecture, and biometric identity remote authentication algorithm will be proposed.

It has been discussed in Chapter 2 that self-sovereign identity management system is significantly related with W3C decentralized identifier(DID) [460] and decentralized systems. It is admittedly true that Blockchain is not the only decentralized system in current literature, there are more than 90 registered methods that can be used to

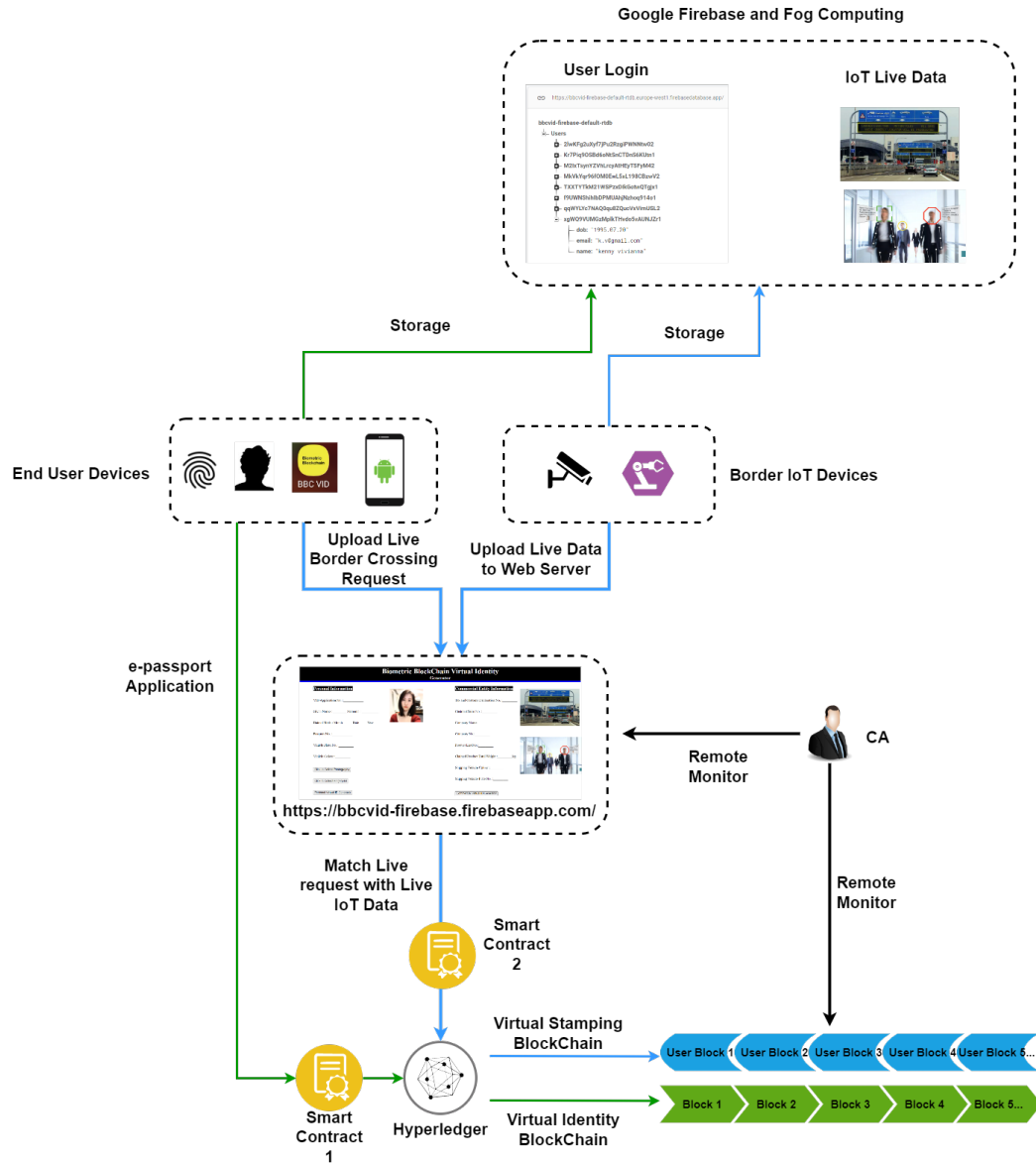


Figure 7.1: Full workflow of the proposed BBC ABC system.

generate a DID [462]. However, Blockchain innate properties make it the best suit for our proposal. To start with, we claim the proposed BBCVID has below three features:

1. **Self-sovereign identity (SSI)**. In currently literature, self-sovereign identity requires trust from identity issuer, as every participant in SSI IdM can be an identity issuer to issue all kinds of identity for himself. To tackle this trust issue,

pretty good privacy [415] and web of trust [14] has been proposed. However, they both require the identity is non-private and can be publicly resolvable, such as the public key infrastructure. In our proposal that is biometric e-passport, it is contradictory with public resolvable requirement. Therefore, a governmental organization is introduced as the central authority in BBCVID(BBCVID CA).

2. **Biometric identity awareness.** BBCVID is aware of user's biometric identity, and the biometric identity template is only available to permitted person based on requests, and is stored only in BBCVID CA's trusted database. Obviously, some established IdM protocol can fulfil similar task, such as FIDO, Fast Identity Online[227]. However, it requires to register the hardware based authenticator at remote authentication server, which imposes more costs for users. In our proposal, apart from mobile phone, there is no extra costs for new hardware at all. To put more in detail, the central arguments in remote biometric identity authentication are derived from two points:

- (a) Who conducts the authentication. That is to say, since the digital authentication is conducted remotely, the entity who authenticates the identity intuitively does not trust nor accept anyone else's identity authentication outcome apart from the entity does the authentication itself, because it is very easy to be cheated about authentication outcome.
- (b) What the identity credential is. As digital identity authentication is based on risk-based assurance, which the knowledge and/or the possession of the identity credential is crucial in the regard of judging how much assurance of the corresponding identity authentication has. Therefore, remote identity authentication server normally has to make sure the correctness and integrity of the credential, which is to maintain and store user's credential by remote authentication server itself.

That is to say, by deploying Blockchain, biometric template and the biometric identity authentication outcome can be completely trusted. To put more details in,

- (a) Atomic and immutable smart contract. Blockchain smart contract is designed to conduct the digital identity authentication, which guarantees that both the authentication mechanism integrity and the outcome are immutable due to smart contract atomic nature [503].
- (b) Immutable recording of the biometric identity template. Biometric identity template as digital credential has the feature of inexact input and inexact output; therefore, it is impossible to create an integrity checksum between two biometric identity credentials that are collected at different time even

though they are collected from the same person. Therefore, Blockchain is employed to record the biometric template hash output as a checksum for original biometric identity template, which is also recorded on the Blockchain as a reference. That is to say, in the future case where biometric identity template is required, one can make a hash over the biometric identity template they received and verify if it is the same as the Blockchain ledger records. In this way, the biometric identity template is reliable if its integrity can be guaranteed by the Blockchain ledger.

3. **Improved interoperability.** The e-passport are formatted in W3C defined digital identity credential data model v 1.1 [121] which aims for improving the credential's interoperability to the most.

7.2.1 Main IdM Components in BBCVID

An IdM system is defined as ‘...the set of processes, tools, and social contracts surrounding the creation, maintenance, and termination of a digital identity. ...to enable secure access to an expanding set of systems and applications [423].’ To fulfil those functions, four major components are defined in the proposed e-passport IdM system.

1. **Repository.** The data model, logical data storage, and a policy which governs access to, and the use of this repository is kept in this component. All credential issued by BBCVID is formatted as .json, which is stored at issuer's own digital wallets(user end device), so does the biometric identity information. Only hashed output of encrypted data are recorded on-BBCVID for immutability.
2. **Authentication centre.** Authentication centre takes responsibility for conducting primary identity authentication, and then generates authentication token to allow other components and third parties to recognize the primary authentication has been performed. In our proposed authentication centre, smart contracts conducts the whole life cycle of all e-passports including but not limited to CRUD but requires a user to initiate it.
3. **Policy control.** Policy control defines the scenario where and how identity information can be revealed, and how identity authentication protocol can be initiated. It clearly lists the terms and conditions if an audit event will be triggered so that identity data misuse can be constraint as much as possible. In our proposal, policy control as defined by the term of information use conducts.
4. **Auditing.** An audit mechanism overlooks all conducts related with the data in the repository, which enables policy control terms can be executed

and circumvented accordingly. In this proposal, e-passport has immutable checksum and CA signature attached to all e-passports. Whenever policy control defined irregular event being detected, such as incorrect checksum and digital signature can not be authenticated. Auditing events will be initiated into the corresponding digital wallets to investigate. Most important, e-passport CRUD operations are executed by smart contracts and reflected in ‘version’ corresponding floating number value. The authentication centre has to enquiry the most latest version to conduct biometric identity authentication in case of CRUD operations. Auditing events and mechanism also defines the terms that what if a previous version of a passport is being used for identity authentication.

7.2.2 Main Participants in BBCVID

To enable above component functions, BBCVID has participant who takes responsibility in accordance with its assigned role within the system so that the system function can be fulfilled accordingly. The main participants in the system include:

1. **Central Authority (BBCVID CA).** CA is defined as a trusted authority in BBCVID system, who take the responsibility to order transactions as orderer, and validate transactions occurred on BBCVID. BBCVID CA’s identity is public on BBCVID’s channel built block, where all participants can read it. Because the proposal is particularly for e-passport, CA can be the border agency or passport office of the country. Most important, as the only trusted entity within the BBCVID system, BBCVID CA maintains Blockchain consensus. CA periodically issuing a signed data structure called a certificate revocation list (CRL) [231], which requires all users to enquiry this list before identity authentication is conducted.
2. **BBCVID Users.** BBCVID user uses developed Android mobile application to interact with other BBCVID users and BBCVID Blockchain. User initiates e-passport application transaction proposal from the mobile client to BBCVID CA’ gateway service, which requires user to fill a form and visit BBCVID referred biometric identity template collection points. Upon receiving user’s application, a smart contract will be initiated straightaway for initial biometric identity authentication and DID document generation. If user’s identity is authenticated with its submitted biometric identity template, a DID will be returned to user’s mobile client which its corresponding DID document is only resolvable by user himself and user permitted entity.
3. **BBCVID Blockchain.** BBCVID Blockchain has three main responsibilities, which include:

- (a) User e-passport and digital verifiable border crossing documentation generation.
- (b) Record and host identity authentication events.
- (c) Host CRUD operations and record it on chain.

7.2.3 Biometric Identity Authentication and BBCVID Main Operations

BBCVID Registration

Before user is able to obtain the proposed e-passport, user has to register in the BBCVID mobile application to obtain a user account. To put more in detail, registration in the BBCVID mobile application is a very simple and straightforward process, which only an email address and a six-digit PIN are required. On registration completion, three goals are achieved, which include:

1. Access to the BBCVID mobile application is obtain. The application is a bridge between user and the BBCVID network, which enables user to raise transaction proposal to BBCVID network through BBCVID gateway service.
2. A PKI-based digital identity certificate will be generated for the user. Since BBCVID is a permissioned Blockchain, user's identity has to be identifiable by the BBCVID network before any interactions. Therefore, BBCVID Blockchain certificate authority(Fabric CA) will issue a PKI based digital identity certificate to the user as soon as registration is completed.
3. TLS based data exchange protocol can be established. Since user obtains a certificated public key from BBCVID Blockchain certificate authority, data exchange over open internet through TLS and the access to the BBCVID network can be secured by imposing data exchange protocol and digital signatures. Plus, data integrity and veracity can be guaranteed as well, see **Algorithm 1** for BBCVID registration.

Data Exchange Protocol

The identity data transmission over internet is conducted in accordance with IEEE Standard for Biometric Open Protocol where $n = 1$ that n is required $\neq 2$ [244]. To put more in detail, Fabric CA will generate one pair of one time public and private key pair for himself, so does user at his own mobile client. Fabric CA passes his one time public key to user, and user uses it to encrypt the data first, and then uses his

Algorithm 1 User Registration in BBCVID

Registration : $Join\ BBCVID \leftarrow User$
Require: email address, six – digits PIN
if Email address \notin application user database **then**
 (1) Store user credential in the mobile application system database
 (2) CA issues user X.509 public key based digital identity certificate
 (3) Return PKI certificate back to user
end if
Registration is completed

own one time private key to encrypt it again. User finally sends double encrypted biometric identity template and his own one time public key to Fabric CA. Fabric CA then can use his own one time private key and user’s one time public key to decrypt user’s biometric identity template, see **Figure 6.2** for the IEEE BOPS biometric data exchange protocol.

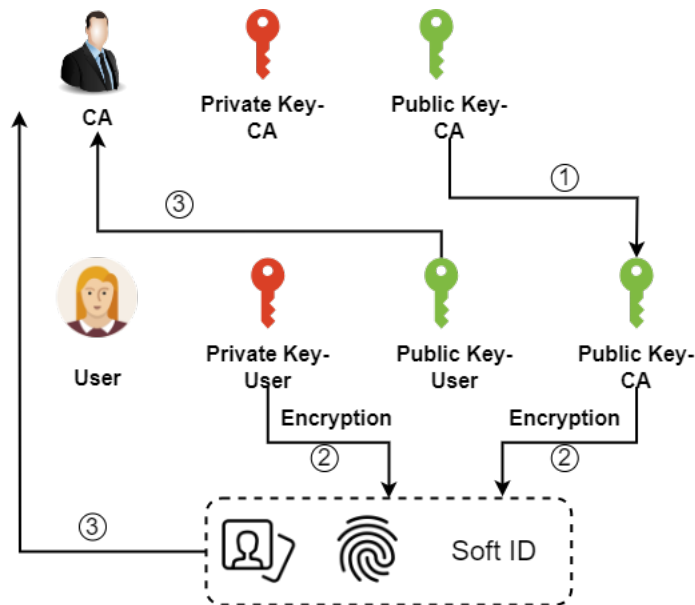


Figure 7.2: Enrolment private data exchange through Biometric Open Protocol [244].

Biometric Identity Enrolment

To enrol user’s biometric identity template in the BBCVID system, user is required to visit biometric identity collection points, which is the same as current e-passport system. At the biometric identity template collection point, both user’s facial image

and fingerprint will be collected, and then linked with user's public key that is issued by BBCVID Hyperledger Fabric Blockchain certificate authority(Fabric CA) ³.

To complete the biometric identity enrolment process, user is required to log into the mobile application after visiting the biometric identity collection points. Through the mobile client, user is able to navigate to the 'biometric identity enrolment' page and then submit facial image over it to BBCVID CA. BBCVID CA biometric authenticate user's facial image by its own database records, and then issue user a DID and DID document. The aim of biometric identity enrolment is to let user obtain a DID to referent *himself* in the BBCVID system. That is to say, user obtains a Biometric DID through biometric identity enrolment, which the DID identifies the bodily person of the user himself. DID document contains the authentication method that can be used to authenticate user's ownership of the corresponding DID.

To add on, BBCVID CA can easily decrypt user's encrypted facial image by user's one time public key and BBCVID CA's own one-time private key. Therefore, BBCVID CA then should take facial identity authentication to guarantee that the biometric identity template recorded in the DID document is veracity, and the DID identified user is the same person who is recorded in the DID document. If all above procedures are completed, BBCVID CA is able to issue user the DID and DID documents to user, which the DID document includes

1. **User's DID.** The DID method of BBCVID is named as 'bbcvid';therefore, the DID of the user follows the format of {"id": "did:bbcvid:<unique resource identifier>"}
2. **Authentication methods.** Each authentication method contains authentication material, authentication mechanism, and authentication evidence. In our proposal, three authentication methods are offered in DID document, which include public key, facial image, and fingerprint. To preserve user's biometric identity privacy, facial image and fingerprint data is not directly offered in DID document as raw biometric identity template plain-text. Specifically, facial image is hash by SHA256 and the hash output will be recorded in DID document, so does fingerprint identity template.
3. **Credential Status Lists Address.** CA maintains a credential status list which is used for broadcasting revoked identities. All entities are encouraged to consult this list before identity authentication is conducted.
4. **Proof.** Proof is the digital signature of the CA, which is used to endorse the integrity, veracity and authenticity of the DID document and the corresponding DID.

³Fabric CA refers to BBCVID Hyperledger Fabric Blockchain **Certificate** Authority, and BBCVID CA refers to BBCVID **Central** Authority

Obtaining the DID and DID documentation symbolizes a user fully enrolled in BBCVID. The DID document is resolvable from the DID, and is endorsed and digitally signed by BBCVID CA. After enrolment, three goals are achieved:

1. DID identifies a person who is the same person with DID document identified person.
2. The hash outputs of both facial and fingerprint identity template are immutably recorded on BBCVID.
3. The DID document resolved from the DID is immutable and its integrity is verifiable by CA's digital signature appended at the end of the DID document.

What is more, BBCVID CA stores user's biometric identity in BBCVID CA's own storage after encrypt user's biometric identity template in BBCVID CA's own way. The reason of BBCVID CA has to store user's biometric identity template are out of two considerations in this proposal, which both of them are the same with UK passport office's claim about retention user's biometric identity [504].

1. BBCVID CA has to store user's biometric identity in CA's own database which is not only for eliminating impersonation risks but also for the requirement of a country's immigration control conduct that biometric identity of an immigrant has to be recorded and filed as an evidence of their immigration status [504].
2. Biometric information enables BBCVID CA to identify criminals and illegal migrants effectively which aims to prevent a leave is false given[504].

Most important, as in distributed Blockchain system, the biometric identity database can be decentralized stored into BBCVID CA's peer database rather than just one single database. That is to say, since user is able to interact with any BBCVID CA's peer nodes, user's enrolment operation can be conducted by any of it. Therefore, user's biometric identity template can be stored in any of BBCVID CA's peer nodes, which eases the risk of single point attack and data breach.

Biometric Identity Authentication

It has been mentioned above that to enrol a user, BBCVID CA should conduct identity authentication, and there are three authentication methods offered in DID document altogether. These three authentication methods include public key, facial image and fingerprint. To be more precise, public key has a cryptography relationship with user's private key; therefore, to authenticate user by public key, verifier only needs to ask user to sign verifier specified data, or calling a smart contract to conduct the authentication which has exactly the same workflow with verifier.

To conduct biometric identity authentication, DID document specifies three things for verifier:

1. Hash of the facial image template.
2. Hash of the fingerprint template.
3. Biometric identity authentication smart contract address.

DID document should never store user’s biometric identity directly; therefore, the DID documents only store the hash output of user’s biometric identity template as an evidence(e.g.SHA256).

Since DID document specifies the identity authentication smart contract address already, verifier only requires to put user’s DID, and the smart contract address as input parameter to initiate the biometric authentication process. Then smart contract will take fresh biometric identity from user through Android mobile application, and collects the biometric identity template that is already stored in the CA’s database, and finally match the fresh biometric identity with biometric identity template to authenticate user. Most important, smart contract checks up if the hash output of received biometric identity template is exactly the same with DID document record first, and then give a Boolean about the authentication result(either fail or success). If the biometric identity template hash output is different from DID document, the biometric identity authentication will ‘fail’ straightaway. If the hash output is exactly the same with DID document, and biometric identity authentication is matched, smart contract will give ‘success’ as authentication output. If the hash output is exactly the same with DID document, but biometric identity authentication is failed, smart contract will give ‘fail’ as authentication output, see **Algorithm 2** for identity authentication.

Most important, as the mobile application is built upon Android, which the fingerprint collected from Android mobile touch sensor is neither accessible nor operable at all. Therefore, only facial image authentication over the Android mobile is permitted. However, fingerprint identity authentication is possible over border checkpoint e-gates system. In later section 7.3 VS for automatic border control will introduce more about it.

To the specific algorithm of biometric identity authentication, one-to-one feature vector matching is used, and the result of that depends on a cost function to be minimized. That is, $X = (x_1, x_2, \dots, x_n)^T$ is assumed as the extracted biometric template feature vector, and $X' = (x'_1, x'_2, \dots, x'_n)^T$ is a freshly extracted biometric feature vector. The matching metric rule [545] is to calculate the Euclidean distance by below formula.

$$dist(X, X') = \left(\sum_{i=1}^n (x_i - x'_i)^2 \right)^{1/2} (7.1)$$

Algorithm 2 Biometric identity authentication via e-passport

*Biometric identity authentication : JSON Web Token(JWT) \leftarrow
e-passport(EP), collected biometric sample(CBC)*

Require: *User's DID, EP, CBC*

Require: *Verifier's public key*

Verifier initiates 'Biometric Authentication' smart contract

Smart contract run :

if (1)*Digital signature in EP is authenticated* **then**
 (2)*Hash_{BT} \leftarrow SHA256(CA offered biometric template(BT))*
if (3)*Hash_{BT} == Hash output recorded in EP* **then**
 (4)*dist_i⁴ \leftarrow match CBC with BT by equation 7.1*
for *i* \leq 2
if (5)*controlled threshold $\sigma_i \geq dist_i$* **then**
 (6)*'success' \leftarrow Biometric identity is authenticated*
 (7)*Return JWT to user*
else
 (6)*'fail' \leftarrow Biometric identity is failed*
 (7)*Ask user to resubmit CBC*
 (8)*Repeat process (1) – (5)*
end if
end for
else
 (4)*'fail' \leftarrow Biometric identity is failed*
 (5)*User's BT is modified, report user to CA*
 (6)*CA ask user to update EP*
end if
else
A fake EP is found. Report user to CA and revoke this EP
end if
Biometric identity authentication is completed

To put more detail about who is able to request a particular user to conduct the biometric identity authentication through user's e-passport, it has to be emphasized that user's e-passport cannot be found passively. That is to say, only user can actively share it with target subject. Another fact to be mentioned here is apart from e-passport, a user also obtains a DID and a DID document(DID's credential), which has the same property with e-passport. Since DID, DID document, and e-passport are all endorsed by BBCVID CA, any entity who knows user's DID is able to make a 'direct' request to user to conduct (biometric) identity authentication, and the identity authentication proves a bodily person(user) has full control over the DID.

E-passport CRUD Operations

After enrolment, user obtains DID digital twin that is used to referent himself in the Metaverse, and a biometric DID document that is endorsed by CA with CA's digital signature.

C. Create

When user is ready to make proposed e-passport application to CA through user's Android mobile application client, user fills form data and submits it in together with DID to CA. A smart contract will be initiated straightaway as soon as user click "submit" button on the client. Smart contract will initiate biometric identity authentication first, and then conduct passport claim legitimacy checkups in CA's database. If all settled, smart contract will ask CA to sign a digital signature on its output. Indeed, the output of the smart contract is the proposed e-passport, which is a digital verifiable biometric credential. The e-passport has user's DID, credentials subject(e-passport), proof, and credential status. It has been discussed in Chapter 3.3.1 about the W3C basic structure of a decentralized digital verifiable credential data mode, and our proposed design is consistent with it, see **Algorithm 3** for Creating e-passport.

R. Read

BBCVID e-passport is built upon Hyperledger private Blockchain. It has been mentioned in the very beginning of this chapter about why we choose Hyperledger private Blockchain, which can be summarised as flexibility in system architects, smart contracts extensibility, easy to scale up, and very low identity authentication and maintenance costs. Indeed, another reason of that is out of preserving user's identity credential privacy concern.

To put more in detail, the proposed e-passport is digital identity biometric

Algorithm 3 Creating E-passport

Create : $E - passport \leftarrow User, DID$
Require : $User's DID$ and soft legal identity
Initiate 'Create' smart contract
Smart contract run :
if $DID \exists BBCVID$ **then**
 (1) *Biometric identity authentication*
 (2) *Soft identity checkups*
 (3) *Generate e - passport*
 (4) *BBCVID CA signs on e - passport*
else if
 then *Application is declined, and ask user to register and enrol first*
end if
Return e - passport to applicant
Creating e - passport is completed

Algorithm 4 Updating E-passport

Update : $Updated e - passport \leftarrow Previous e - passport$
Require : $Previous e - passport$ and information to be updated
Initiate 'Update' smart contract
Smart contract run :
if *information to be updated == soft identity* **then**
 (1) $BBCVID CA \leftarrow DID$ and new data form
 (2) *Biometric identity authentication*
 (3) *Soft identity checkups and updates*
 (4) *Generate new e - passport*
 (5) *BBCVID CA signs on new e - passport*
 (6) *Revoke previous e - passport*
else if *information to be updated == biometric identity template* **then**
 (1) $BBCVID CA \leftarrow Old EP$ and new biometric identity template
 (2) *New e-passport, new DID* $\leftarrow BBCVID CA$ initiates 'Enrolment' process to re-enrol user
 (3) *Revoke previous DID*
 (4) *Revoke previous e - passport*
end if
Return new e - passport to applicant
Updating e - passport is completed

credential, which is deemed as an equivalent of the government passport book. Therefore, none would like the concept of allowing everyone else to read your passport. Most important, extensively encrypted e-passport suffer some drawbacks like very complex computation, very expensive input cost in initial encryption library setups, and complicated key exchange etc. Therefore, in our proposal, it is proposed to simply limit the read access to outsiders apart from the e-passport holder himself which can be fulfilled by simply making modifier modification private collection re-definition in smart contract.

That is to say, apart from the e-passport holder and BBCVID CA who validate and endorse the e-passport, no one else can read user's e-passport without permission. To add on, proposed e-passport is generated by making a transaction on BBCVID by BBCVID CA; therefore, e-passport is a DID as well which only the passport holder can resolve it. That indicates one thing: the proposed e-passport cannot be found online passively; however, the e-passport holder is able to actively share it with any entity through proposed data exchange protocol in 7.2.3. for identity authentication purposes.

U. Update

Updating e-passport can be made by calling "Update" smart contract and input a new data form along with the DID to BBCVID CA peer node. Updating content can be made in two different components of the e-passport, which include:

1. Update soft identity. Soft identity modification and updates like given name and family name etc.
2. Updating biometric template.

Both above updates and modifications can be made through proposed Android mobile application and select corresponding subsection to redirect user to the corresponding client page to make the update.

- If soft identity needs to be updated, user refill the e-passport data form to CA and along with DID. CA will verify soft identity updates legitimacy first, if update is permitted and verified by CA, CA will initiate 'update' smart contract to conduct biometric identity authentication with user, and then issue new e-passport to user after signing on it. This process is very similar with 'creating' e-passport algorithm.
- If user requests to modify and update biometric identity template, user is required to re-visit biometric identity biometric identity collection points and then submit the old e-passport to CA. CA then needs to re-enrol the user. Most

important, CA has to make sure the old biometric identity template and the new biometric template are referent the same person as well. After re-enrolment and biometric identity authentication, CA will generate a new DID and new e-passport for user, and put the old DID and old e-passport on to ‘revoke lists’ to notify the public, see **Algorithm 4** for updating e-passport.

D. Deleting

Since there is no way to delete ledger records from Blockchain, deleting of e-passport is to raise revoking request to CA, which require user to submit e-passport DID to CA. CA will then put the corresponding e-passport into “revoke list” to notify the public. CA maintains the revoke list and it is recommended for all entities to consult this list before any identity authentication.

7.3 VS for Automatic Border Control

Internet makes remote communication possible and completely blur the concept of distance. The widespread of internet and the significant improvement in computing technology makes small IoTs such as mobile phones are capable of even more challenging tasks. Remote control is a computational intense and latency critical task, which the infrastructure development is an indispensable part of the remote control applications. Therefore, in this proposal Fog computing is employed to offload and speed up the border control workflow, which is to decentralized the whole border control procedure at where the workload is initiated as much as possible. The design of that aims to make sure the border remote surveillance system through IoTs are smooth and efficient.

Understanding the Automatic Border Customs Inspection Process

In chapter 2.5.2, it has been discussed that the main challenge of smart border customs control is efficient and secure documentation authentication and reconciliations. In current literature, there are a few on-going projects [4], [268], [425] need to tackle this challenge by Blockchain technology. Due to the fact that they are government oriented projects and are still in development, the full technical report is not available at the moment. However, one common factor is that they all built upon permissioned Blockchain, which aims to conduct document access control to preserve privacy and protect system security. In this proposal, VS Blockchain is permissioned private Blockchain, which several organizations are required to endorse VS transactions in accordance with the smart contract installed in the VS Blockchain.

The main task in remote border customs control is to ensure the *data consistency* between customs clearance documentations and the real time in-situ border crossing events through remote monitor, which includes the vehicle identity, driver identity and products declared to enter the country [276]. To put more in detail, the main target subjects and the data should be taken care of include:

- **Driver.** The identity of the driver, and driver’s legitimacy of entering.
- **Vehicle.** The identity of the vehicle and vehicle legitimacy of border crossing etc.
- **Products.** The products related documentations such as packaging list, invoices, product model and category etc, the origin of the products, import permits, licenses and certificates etc.

It is also realized and acknowledged that the customs procedure vary a lot among different countries; however, to under the inspection process, the generalized framework of the customs process is introduced. That is, upon *commercial vehicle* arriving at the border checkpoints, the country that the vehicle is going to enter should conduct customs inspections to guarantee that the vehicle, driver and the vehicle loaded products information is consistent with submitted customs clearance documentations. For *private vehicle* arriving at the border checkpoints, inspections should focus on driver’s identity, driver’s legitimacy of entry, vehicle identity, and vehicle ownership etc. Most important, normally there is booth at the border checkpoints to conduct access control, which is mainly used for preventing non-permitted entry.

That is to say, for private vehicle border crossing, identifying and authenticating the identity of both the vehicle and the driver becomes the main task, which is deemed as a task of *remote identity authentication*. However, as for commercial vehicle, apart from driver and vehicle’s identities are required to be authenticated, the product’s identity like the origin of the product and the product’s import permit related license and certificates etc. are all required to be authenticated, which is deemed as *cross platform data reconciliation and authentication*.

Understanding the Individual Immigrant Automatic Border Control Process

Border crossing for individual immigrant control has to compile the immigrant law of the country; therefore, the legitimacy of entering and the identity of the immigrant has to be confirmed and verified before the border crossing event happen in reality.

In general, border immigrant control process is to:

1. Allow immigrant obtain entering legitimacy credential(e.g. visa) and identity credential(e.g. passport).
2. Immigrants bring these credentials to the airport or land border checkpoints.
3. Upon the present of the immigrant, border crossing legitimacy credentials, and identity credentials, border officer is able to conduct:
 - (a) Credential authentication.
 - (b) Identity authentication
 - (c) Granting border crossing permit.
 - (d) Stamping and recording the border crossing event.

Automatic border immigrant control is to use computing techniques to automatize above workflow. To put more in detail, current machine readable e-passport book is to replace border officer to conduct identity authentication and credential authentication by border checkpoint e-gates, which does improve the efficiency of the whole workflow. However, nowadays when there is a sharp increase in immigrants population due to war affair but limited physical space to setting more e-gates around the checkpoint, a call is drawn for even more automatized border control solution.

7.3.1 Main Components in VS

Since our proposal is focus on airport and land border checkpoints, the proposed solution is divided into automatic immigrant(airport) and automatic customs(land border checkpoints) control system, see **Figure 6.1** for full workflow of the proposed system. To put more in detail, the proposed automatic border solution is specifically consists of three main components based on the understanding of ABC process for both immigrants and customs control:

1. **Customs documentation(and/or identity credential) sharing and authentication.** It has been discussed that both customs documentation and identity authentication is an indispensable part of ABC solution. That is, immigrants identity authentication in airport, driver and vehicle's identity authentication in land border checkpoints, and customs documentation authentication is unavoidable and compulsory.

Even though BBCVID is designed for issuing and managing user's e-passport and DID in immutable, self sovereign to user, and privacy well preserved manner. it is able to allow both governmental customs organization(GCO) and vehicle ⁵

⁵Vehicle's DID requires a DID controller who takes full control over the vehicle. That is, the owner of the vehicle.

to register, enrol in, and obtain a corresponding DID and DID document as well. By doing that, the GCO holds a CA endorsed DID, which can be authenticated by its own public key. GCO then can have peer-to-peer conversation with all the rest of users on BBCVID, therefore, to issue a verifiable customs document like import/export permit, GCO can simply allow user to submit requested data through the proposed data exchange protocol first, and then initiate identity authentication with the user. GCO then needs to audit the data submitted from user, and finally sign GCO's signature on import/export permit document to endorse it. Therefore, a customs document can be authenticated by authenticating GCO's signature. That is to say, *documentation authentication* is well sorted by BBCVID, see **Algorithm 5** for creating GCO verifiable documentation.

As for *documentation sharing*, since GCO's documentation is issued through BBCVID, the documentation shares the same feature with e-passport. That is to say, 'read' access to GCO issued documentation is constraint as well. BBCVID ledger 'Read' access is constraint to issuer and user only, and can be passively accessed by raising access request, which means GCO's document is only readable to GCO, document holder, and CA. Anyone else would like to read the document, an access request has to made to one of the entity who has already has the access, and a documentation exchange has to be conducted in accordance with data exchange protocol defined in chapter 6.2.3.

2. **Border checkpoints live in-situ data collection.** To reconcile border checkpoint live in-situ data with the corresponding customs documentations, collecting the live data from the border checkpoint through border surveillance system IoT is crucial. Border checkpoint collects below data from IoTs:
 - (a) CCTV camera for vehicle number plate. The out put is the vehicle's number plate number.
 - (b) Weight sensor to weight vehicle's weight. The output is the weight of the vehicle.
 - (c) Radio frequency identification (RFID) reader for reading RFID e-seal number on the vehicle. The output is a Boolean to suggest if further inspection is required or not: either Yes or No.

These three data elements will be grouped as a tuple, e.g. a JSON object {"vehicle number plate": "AB12CDE", "weights": 1250, "further inspections": "No"}, and then send to VS consortium Blockchain.

For the identity of the driver, it is also required to be authenticated at the border checkpoints. In this proposal, Android mobile application is used to

conduct the remote biometric identity authentication through the proposed e-passport on BBCVID. In chapter 6.3.4 more details will be introduced about the biometric identity authentication process, but e-gate as the final step in ABC, it belongs to part of the border surveillance system and is required to return decrypted information in the QR code to VS.

For a short summary, through border surveillance system and border checkpoint e-gate, *border checkpoints live in-site data* can be effectively collected, which is ready to be reconciled with the data in customs documentations.

- 3. Data reconciliation.** Data reconciliation between customs documentations and border checkpoints live in-situ data is conducted by smart contracts and recorded on VS Blockchain. To put more in detail, VS Blockchain records time stamped border crossing event as a transaction ledger with digital signature, which is designed to copy hard border control regime governmental stamp leaves at passport book for future referencing. VS is a permissioned consortium Hyperledger Blockchain, which a few governmental organizations jointly take weighted responsibility to maintain it. To be more precise, VS Blockchain (1) receives data submitted from immigrants and commercial business entity, (2) collects data from border checkpoints surveillance system for border checkpoints live in-situ data, and (3) read BBCVID chain for immigrants and driver's biometric identity authentication outcome and receive e-gate returned data from decrypted QR code, and (4) finally initiate a smart contract to generate a valid border crossing event transaction ledger in accordance with the data reconciliation outcome. That is, the immutable and automatic *data reconciliation* through remote surveillance system and customs documentation sharing and authentication is accomplished.

7.3.2 Main Participants in VS

VS has participants. In accordance with their role within the system, different participants take different responsibilities. To put more in detail,

- 1. Border Agency (BA).** BA is entity who has the most control weight in the VS consortium. To put more in detail, BA initiates 'data reconciliation' smart contract and signs on the VS transactions. Most important, BA indeed is the same entity with BBCVID's centralized authority(CA). That is to say, BA is the entity who endorse the rest of the members of VS consortium in BBCVID so that the rest of the VS consortium member is able to issue verifiable documentation. Since BA and CA is the same entity, BA is well aware of both who and when specifically an applicant's biometric identity is authenticated.

Algorithm 5 Creating GCO Verifiable Documentation

Creating: Verifiable documentation \leftarrow *applicant's DID, data form*
Require: *Applicant's DID, data form*
Require: *GCO's public key*
Initiate 'GCO Verifiable Documentation' smart contract
Smart contract run:
if *Applicant data form compiles customs regulation* **then**
 (1)*GCO initiates applicant identity authentication through e – passport*⁶
 (2)*GCO generates documentation*
 (3)*GCO signs on documentation*
else if
 then *Documentaiton creation fails and GCO asks applicant to resubmit border crossing documentation application*
end if
Return documentation to applicant
Creating GCO verifiable documentation is completed

2. **Driver and Vehicle Licensing Agency(DVLA).** DVLA is one member of the VS consortium, who takes the responsibility of verifying the relationship between vehicle and vehicle owner in vehicle's DID document after DVLA obtaining its own endorsed DID in BBCVID. DVLA also needs to be on-call for BA, in case of BA requests any vehicle information, such as the owner's information of a particular private vehicle. DVLA does not initiate any transaction nor sign any transactions on VS chain, but DVLA can make a vote toward if a border crossing permit should be granted to a specific applicant based on the information DVLA holds.
3. **Her Majesty's Revenue and Customs (HMRC).** HMRC is also a member of VS consortium, who offers BA a customs clearance documentation *list* in accordance with user submitted invoice and packaging lists. BA then can use the list as a reference to retrieve all documentations on BBCVID and then authenticate them one by one. To add on, HMRC has an absolute decisive saying about what documentation an applicant is required to obtain before entering the country with loaded product on vehicle.
4. **Border checkpoints border crossing applicants.** Every entity who would like to get border crossing permit is required to submit border crossing application to VS BA. To put more in detail, land border checkpoints border crossing applicant can be:
 - (a) **Individual person.** Individual pass through applicant is subject to border

immigrant control, which only the individual's pass through legitimacy needs to be verified. Therefore, individual person is required to make the border crossing application over the Android mobile application to generate a electronic pass(e.g. QR code).

- (b) **Private vehicle.** Both the private vehicle and the driver's border crossing legitimacy need to be verified. It is the private vehicle's driver's responsibility to submit the border crossing application to BA. Driver fills form data in mobile client about the vehicle information(e.g. ownership information and number plate etc.) and his own personal identity information, and then send to VS BA for border crossing permit.
- (c) **Commercial vehicle.** Commercial vehicle border crossing permit application can be made by commercial vehicle's business entity, who is required to fill form data about vehicle driver(e.g. driver's e-passport and DID), vehicle(vehicle ownership and number plate), and vehicle loaded product information(e.g. invoices, packaging lists, and all HMRC required border crossing documentations.) to VS BA.

5. **VS Blockchain.** VS Blockchain is a permissioned consortium Blockchain built upon Hyperledger. VS Blockchain immutably records all border crossing events as transaction ledger. Based on applicant's application, a VS transaction will be generated but without BA's signature, which means the transaction is not valid yet. Upon the applicant using the transaction to initiate data reconciliation process around the border checkpoints, BA will finally sign on the transaction after data reconciliation is completed with border surveillance systems. A VS transaction with BA's signature is time stamped at when the entity is crossing the border, and it is a valid, immutable, and legal proof of the entity's border crossing history.

6. **Surveillance System.** Surveillance system as a participant within the proposed system who is also a member of the consortium. It uses computing techniques to collect border checkpoints live in-situ data in a smart and efficient way, which the data collected is immutably recorded on BBCVID and is available for VS BA to reconcile the data with applicant's documentations. Surveillance system is very intelligent in the regard of:

- (a) IoT work mechanism and logic is controlled completely by installed smart contract which is atomic and automatic.
- (b) Data collection and record is automatically executed and controlled by on-chain smart contract, which guarantees the data is a veracious and reliable reflection of the reality.

- (c) By adopting computing deep learning algorithm, data learned from CCTV camera is very accurate, e.g. number plate.
- (d) By adopting Fog computing, the data collection and transmission process is initiated at IoT and will be completed at IoT as well so that the ABC solution can be as smooth and efficient as possible.

7.3.3 System Modelling

At system architecture level, the proposed system is Blockchain based IoT application in Fog, see **Figure 6.3** for the VS system overview. To put more in detail, the proposed system is modelled into four layers, which include:

- **Layer One: Perception layer.** Perception layer collects all sorts of data through user IoTs(e.g. mobile phone and laptops) and the border infrastructure IoTs(CCTV cameras, weight sensors, GPS, and RFID reader etc.), which makes a veracious reflection of the real physical world.
- **Layer Two: Network layer.** Network layer connects all IoTs and all participants to the proposed system, which makes secure and reliable communication between heterogeneous network possible. Network layer mainly is for the purpose of transmitting perception layer collected data to the relevant entity. Most important, since our system is based on Blockchain, network layer also enables the direct peer-to-peer communication and maintains it as secure as possible. To put more in detail, perception layer IoTs within the proposed system is mainly supported by local area network(LAN) such as wifi and local Ethernet, or mobile cellular network. On top of it, the communication between Blockchain peers, IoTs to Blockchain communications are all made by web3.js remote procedure calls.
- **Layer Three: Application layer.** Application layer provides proposed service directly to all participants through developed user interface. Plus, application layer is also able to conduct system environment monitoring in accordance with system policy, and intelligent data routing based on perception layer's data [469].

Specifically, application layer is the interface layer between perception layer and user. That is to say, application receives data from perception layer and then use it to offer service to user. To be more precise, application layer turns border crossing permit applicant's application into a valid border crossing permit which can be used in the border checkpoint to cross the border. By similar token, BA can uses application layer to record user's border crossing event on to VS Blockchain and return user a time stamped digital signed Blockchain ledger as

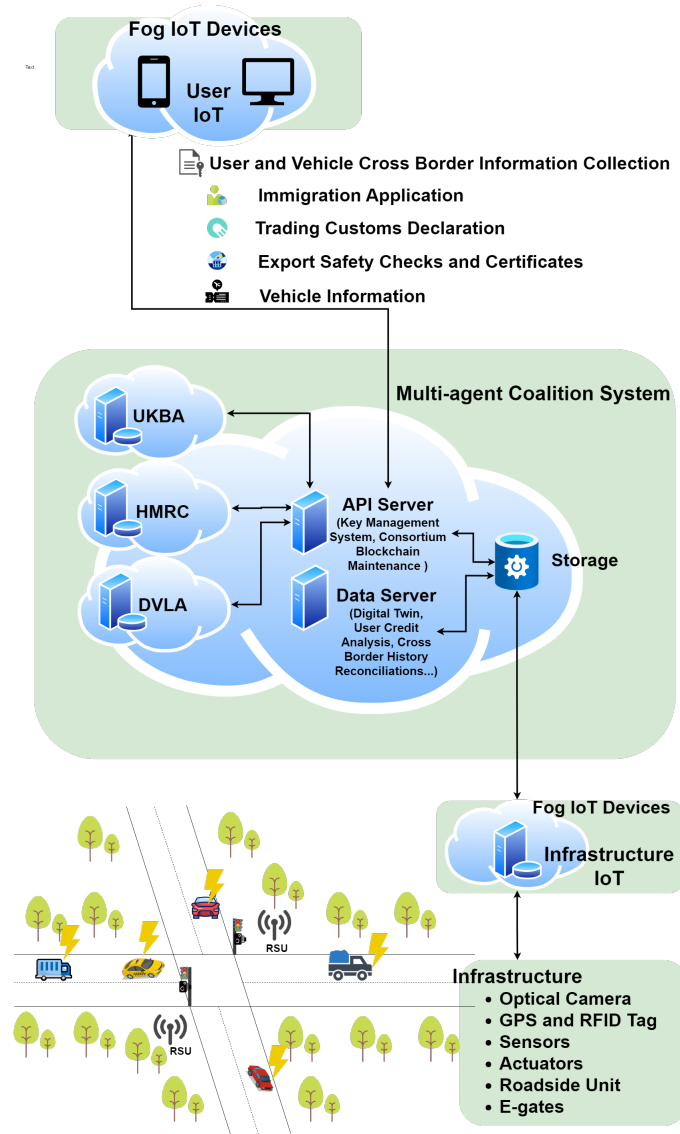


Figure 7.3: VS overview

a proof of that. Most important, this layer also offer identity service to all participants by calling on-chain smart contracts.

- **Layer Four: Inter-operation layer.** Inter-operation layer is the topmost layer in the proposed system, which offers an interface for VS consortium and CA to communicate outside of Blockchain. The inter-operation layer aims for optimizing operational procedures for proposed two permissioned Blockchain. If it is necessary, this layer can be used to build a user credit ranking system

based on all transaction history in both chains.

Most important, network security management is conducted throughout all layers of the system. Like for instance, access control is conducted for both VS and BBCVID Blockchain, and all data exchanged between Blockchain peers is fully encrypted as well.

7.3.4 VS for Automatic Immigrants Control

In chapter 6.2.3, it has been proposed and discussed in detail about how the proposed e-passport is created and how remote biometric authentication through e-passport is conducted(Algorithm 2, 3). Here in this section, the focus is how the proposed e-passport can accelerate the smart immigrant control process.

Existing micro-chip e-passport book, it requires airport traveller to conduct the biometric identity authentication at the e-gate, and e-gate is only available to traveller who does not require a visa to enter the country. That is because the e-gate is not able to record any entry event nor read any visa as it is not stored in the e-passport.

whereas as a contrast, the proposed e-passport is highly extensible. The proposed the e-passport can accelerate the existing immigrant control process in two ways:

1. E-gate is available for all traveller regardless extra visa is required or not. That is, one can easily link visa with the passport(Algorithm 5), and make both the visa and e-passport verifiable. Therefore, the e-gate can be made available to all travellers even a visa is required to enter, which significantly reduces border human labour demand so that the immigrants control efficiency can be sped up. Plus, in our proposal, immigrant's border crossing legitimacy and related documentation can be automatically verified through proposed mobile application before attending the e-gates.
2. Rather than reading micro-chip in old e-passport, the proposed e-passport generates QR code for user to pass through the e-gates. That is, QR code has favourable features like high data capacity and high speed reading. To put more in detail, QR code is generated by VS BA after immigrants border crossing legitimacy has been verified. QR code is used in the final step of border crossing control to authentication immigrants biometric identity at the e-gate. QR code contains parameters to trigger a fingerprint identity authentication at e-gate through proposed e-passport. After immigrant's fingerprint identity is authenticated, e-gates will open to allow the immigrant to cross the border and then send signed information back to VS BA to record and finish off this border crossing event.

To offer a comprehensive description about how an immigrant uses the e-passport to cross the border, the immigrant is required to submit border crossing permit application to VS BA first over the mobile application. The application indeed is a Hyperledger transaction proposal, which the application data is digitally signed by immigrant's private key. VS BA gateway service receives this transaction proposal from mobile client and then transfers it to one of the VS BA peer node to execute the transaction. To put more details in, the smart contract will initiate facial identity authentication first via e-passport over mobile application, and then verify the visa document if the immigrant has one. If the verification goes well, a signed response will be return to the immigrant's application client(QR code). The immigrant arrives at the airport e-gate or land border checkpoints e-gates, using QR code to get across the e-gate by attending fingerprint identity authentication. E-gate returns user a digitally signed response to immigrant's application client after fingerprint identity authentication is completed. Finally, the immigrants packs all responses into an envelope and digitally signs it, and then send to VS BA for ordering service. VS BA validates all responses and all signatures in the envelope, and order the envelope as a new valid transaction by order it into a new Hyperledger block. VS BA broadcasts it to his peer nodes for validating the transaction and adding it to the Hyperledger Blockchain in accordance with consensus rule. The immigrant will receive a immutable and time stamped border crossing record(validated Hyperledger leger) that can be referenced in the future.

7.3.5 VS for Automatic Customs Control

Before a vehicle driving across the border, the border crossing permit should be obtained beforehand. For a private vehicle, apart from all individual immigrant border crossing permit, vehicle's border crossing permit should be obtained from DVLA beforehand as well. For a commercial vehicle, apart from individual and vehicle border crossing permit, the loaded products on the commercial vehicle should obtain border crossing permit and RFID e-seal as well from HMRC. In this section, a comprehensive commercial vehicle border crossing event is detailed described.

To start with, in order to submit the border crossing permit application, the commercial business entity who owns the commercial vehicle is required to obtain a DID for both the vehicle and the business entity. It follows 'BBCVID Registration' and 'Biometric Identity Enrolment' procedure, which the biometric identity is owner of the commercial vehicle and the owner of the business entity. To put more in detail, the owner of the commercial vehicle and the business entity is the 'controller' of the corresponding DID, who is assumed to have full control over the subject that the DID referent to.

Then the commercial vehicle border crossing permit can be made through the

proposed mobile application that is initiated by the commercial business entity. Developed mobile application can be logged in as either an individual or as a business entity. To add more details in, the commercial border crossing permit application starts with a data form which the loaded products invoice, packaging lists, vehicle number plate, driver's visa(if required), and driver's e-passport number are to be filled. The mobile client collects those data and allow commercial entity owner to digitally sign the border crossing permit application(transaction proposal), and finally sends it to VS BA peer nodes to execute the transaction through VS gateway service. The VS BA peer node initiate the corresponding smart contract and re-direct the transaction proposal to the rest of organizations who are required to endorse this transaction in accordance with the endorsement policy in the smart contract. Specifically, by running the corresponding smart contract, below manoeuvres are taken in parallel.

1. Individual person biometric identity authentication and border crossing legitimacy verification. Driver's facial identity will be authenticated in accordance with driver's biometric e-passport, which the facial image will be collected through driver's own mobile DApp. In the case of a visa is also required for the driver to pass through the border, VS Metaverse DAO is able to make visa verification as well. If all goes well, the VS system CA peer node will digitally sign on it to make an endorsement and return it back to business entity's mobile client.
2. Products border crossing legitimacy verification. Upon receiving the products invoice number and packaging lists, HMRC is able to automatically identify what customs clearance documentations are required from the commercial business entity. HMRC then verify customs clearance documents accordingly. After all verification is completed, HMRC is required to endorse the transaction as well and then sends the endorsement back.
3. Vehicle border crossing legitimacy verification. DVLA verifies vehicle border crossing legitimacy by verifying vehicle digital twin documentation and search if the vehicle has been reported lose or any illegal records etc. If all goes well, DVLA will endorse this transaction and sends it back to the commercial business entity mobile client.
4. Border checkpoint surveillance system live data collection. There is a border road surveillance unit constructed where is about 2-3 kilometers away from the border e-gate. Surveillance system collects vehicle number plate, vehicle colour, weights, and RFID reader live data. The surveillance system then endorses the commercial business border crossing transaction proposal, and sends it back to the business entity.

The applicant mobile client collects all digitally signed endorsement and packs them into a data envelope to let applicant (commercial business owner) to digitally sign on it. The data envelope then is sent to VS BA for validation. VS BA validates all digital signatures in the envelope and then reconciled all endorsements. If all goes well, VS BA issues the border crossing permit (QR code) for the commercial business entity specified vehicle driver.

The VS generated QR code is the border crossing permit which is able to allow the commercial vehicle driver to complete fingerprint identity authentication at border e-gate. After driver's fingerprint is authenticated, driver is able to drive the vehicle to get across the e-gate. VS BA then is able to record instance as an immutable transaction on VS Blockchain.

7.4 Chapter Summary

In this chapter, the specific privacy-aware biometric Blockchain-based e-passport system for automatic border control solution is proposed. In **Chapter 6.1 System Overview**, a full proposal workflow is demonstrated and the reason of why Blockchain and Fog computing is adopted is explained. In **Chapter 6.2 BBCVID for e-Passport** the system characteristics are introduced at the very beginning, which is self-sovereign and biometric awareness. Then as an identity management system, its main components, main participants, and the biometric identity authentication and main CRUD operations are defined and explained in detail. In **Chapter 6.3 Automatic Border Control**, to make tasks and goals clear, 'understanding the automatic border customs inspection process' and 'Understanding the Automatic Border Immigrant Control Process' are well explained first. Then main components, main participants and system modelling are explained. Most important, VS for automatic immigrant control and VS for automatic customs control are separately well discussed too.

Chapter 8

Implementation and Performance

8.1 Implementation

To evaluate our proposal properly, the proposed system is implemented and simulated accordingly. For Both BBCVID and VS, they are defined as a permissioned private Blockchain by design. Therefore, Hyperledger Fabric is used as the base Blockchain network. We connect all developed user interfaces(Android mobile application, Java) to Hyperledger Fabric test network by running a full node of it from scratch. As for the consensus rule for both BBCVID and VS, we hard fork them from the test net, which both employ Raft consensus rule. Even though there are similar framework available such as Hyperledger Indy for identity management and Hyperledger Sawtooth for logistics services, we write our own smart contracts and programs as much as possible for customizing every detail of the proposed system.

Hyperledger Fabric Test Network

To put more detail in about the Hyperledger Blockchain configuration and the test network connection, some prerequisites are required before running a full node of Hyperledger Fabric. That is,

1. **Install Git.** Hyperledger Fabric is an open sourced project, which the source code is available on GitHub. Therefore, installing Git for git commands becomes a prerequisite. One of the most crucial git commands to this project is ‘git clone’.
2. **cURL.** As download Hyperledger Fabric is a distributed network for repetitive tasks in accordance with smart contracts, cURL is used for powershell scripting(windows 10 operation system).
3. **Docker and Docker compose.** Docker and Docker compose are particularly for open source project when everyone’s contribution to the project is permitted.

After prerequisites, Hyperledger Fabric is ready to build. Since Hyperledger Fabric is an open source project, its source code is available on Github.com. By following the scripts in GitHub after downloading(and/or cloning) Hyperledger samples and Docker images to local registry, a Hyperledger Fabric test network full node is able to be constructed locally.

See **Figure 7.1** for a demonstration of the initial Hyperledger fabric test network main architects, which includes two organizations: org1 and org2; one peer node in each organization: peer0.org1 and peer0.org2; CLI Docker container for initiating interactions between clients and Hyperledger network(one peer node runs on port 9051 and the other peer node runs on 7051); one orderer node: order.example.com runs on port 7050. Plus, three Docker volumes are created at the same time for independent storage, and they belongs to peer0.org1, peer0.org2, and the orderer. Most important, to this project system implementation, the local Hyperledger test network is build upon Ubuntu 20.04.3 LTS sub-system in Window 10 Version 22H2(OS Build 19045.2846) through Windows-Subsystem-for-Linux 2(WSL 2).

```

bing@inohd281986:~/go/src/github.com/fabric-samples$ cd test-network
bing@inohd281986:~/go/src/github.com/fabric-samples/test-network$ ./network.sh up
Using docker and docker-compose
Starting nodes with CLI timeout of '3' tries and CLI delay of '3' seconds and using database 'leveldb' with crypto from 'cryptogen'
LOCAL_VERSION=v2.5.0
DOCKER_IMAGE_VERSION=v2.5.0
/home/bing/go/src/github.com/fabric-samples/test-network/./bin/cryptogen
Generating certificates using cryptogen tool
Creating Org Identities
+ cryptogen generate --config=./organizations/cryptogen/crypto-config-org1.yaml --output=organizations
org1.example.com
+ test0
Creating Org2 Identities
+ cryptogen generate --config=./organizations/cryptogen/crypto-config-org2.yaml --output=organizations
org2.example.com
+ test0
Creating Orderer Org Identities
+ cryptogen generate --config=./organizations/cryptogen/crypto-config-orderer.yaml --output=organizations
+ test0
Generating CCP files for Org1 and Org2
[+] Running 8/0
   Network fabric_test                Created           0.1s
   Volume "compose_peer0_org2_example.com" Created           0.0s
   Volume "compose_orderer_example.com" Created           0.0s
   Volume "compose_peer0_org1_example.com" Created           0.0s
   Container peer0_org2_example.com    Started          32.1s
   Container orderer_example.com       Started          31.6s
   Container peer0_org1_example.com     Started          33.6s
   Container cli                       Started          39.0s
CONTAINER ID   IMAGE                                COMMAND          CREATED          STATUS          PORTS
5694a674bcf9   hyperledger/fabric-tools:latest     "/bin/bash"     39 seconds ago   Up Less than a second
849dbcf935d1   hyperledger/fabric-peer:latest      "peer node start" 40 seconds ago   Up 8 seconds    0.0.0.0:19051->19051/tcp, 7051/tcp, 0.0.0.0:9445->9445/tcp
2eaa758a11bf   hyperledger/fabric-orderer:latest   "orderer"        40 seconds ago   Up 9 seconds    0.0.0.0:7050->7050/tcp, 0.0.0.0:7053->7053/tcp, 0.0.0.0:9443->9443/tcp
8d222ef489b2   hyperledger/fabric-peer:latest      "peer node start" 40 seconds ago   Up 7 seconds    0.0.0.0:7051->7051/tcp, 0.0.0.0:9444->9444/tcp
bing@inohd281986:~/go/src/github.com/fabric-samples/test-network$

```

Figure 8.1: Hyperledger Fabric test network local builds.

Properties of BBCVID and VS Blockchain

Since both BBCVID and VS are built on top of permissioned private Hyperledger test network, they both have exactly the same Blockchain properties. That is to say, the properties of private permissioned Hyperledger Fabric Blockchain include:

- Access control to the permissioned Blockchain. As *permissioned* Blockchain, all participant's identity should be recognizable to the system before any interaction or communications. In permissioned Hyperledger Fabric Blockchain, certificate authority is the entity within the Hyperledger Fabric network who takes the responsibility of issuing and maintaining all participants' identity

certificates(e.g. x.509 public key certificates.) so that the permissioned Blockchain knows of all participants identity. In our proposal, BBCVID and VS share the same root of trust, which means the public key certificate issued by BBCVID certificate authority is also acknowledged and accepted by VS. Most important, user's permissions to resources and constraints to information of the Blockchain can be granted correctly in accordance with the public key certificate.

- Only one orderer offers ordering service. That is to say, in one particular Hyperledger Fabric transaction, it may require multiple endorsements from multiple organizations to make the transaction valid. However, there is only one organization(may have multiple peer nodes) offering ordering service for all transactions. By doing that, there will be less opportunity for the network to be forked into different chain, which is more secure.

What is more, to further preserve transaction data privacy, Hyperledger Fabric has channel that can be created. The channel is a private communication layer in Hyperledger Fabric, which only permitted and invited organizations can join the channel so that private interactions and communications can be enabled. Most important, organizations in the same channel are required to build on the same Hyperledger Blockchain.

To put more details in about the channel design for both BBCVID and VS Blockchain:

- **BBCVID.** Only one channel is constructed for three organizations below. CA node is the orderer node who offers ordering service, but the other two organizations are peers with CA and required to endorse transactions based on endorsement policy of BBCVID smart contracts. Plus, each node is also assigned one 'admin' who takes control of the node.
 1. Organization One(Orderer): CA.
 2. Organization two: DVLA.
 3. Organization three: HMRC.
- **VS.** Only one channel is constructed for the four organizations below. BA is the orderer who offers ordering service, but the rest of organizations are peers with BA and required to endorse transactions on VS based on endorsement policy of VS smart contracts. Plus, each node is also assigned one 'admin' who takes control of the node.
 1. Organization One(Orderer): BA.

2. Organization two: DVLA.
3. Organization three: HMRC.
4. Organization four: Surveillance system.

Indeed, BA and CA can be the same entity (e.g. UK Border Agency(UKBA), or UK passport office etc).

Transactions

For private Hyperledger Fabric Blockchain, the standardized transaction is made through three phases, which specifically are:

1. Phase 1: Transaction proposal and endorsement.
2. Phase 2: Transaction submission and ordering.
3. Phase 3: Transaction validation and commitment.

For transactions on BBCVID , the entire procedure are proposed in accordance with the border control workflow. To put more in detail, in *BBCVID*, a transaction is made to obtain either the proposed e-passport or other digital verifiable credential such as visa and customs documentations etc. A valid BBCVID transaction is defined as in below steps:

1. Phase 1: E-passport, visa, and customs documentation applicants submit a signed transaction proposal through our mobile application client to the relevant node by connecting to the corresponding gateway service. Like for instance, if the applicant is to obtain digital verifiable customs documentation, the gateway service should offered by HMRC's node. The gateway service will forward applicant's transaction proposal to all relevant nodes to execute the transaction in accordance with the endorsement policy¹ of the smart contract. Finally all nodes return a digitally signed response back to applicant's client.
2. Phase 2: The mobile client pack all responses obtained from Phase 1 into an envelope and let applicant to sign it. Application client submit the signed envelope to CA's gateway service, and a 'success' message will be delivered back to the client if the submission is successful. Upon receiving the response, CA will verify all the signatures in the envelope and then orders the transaction. The transaction will be ordered into a new block of the BBCVID.

¹Endorsement policy is a compulsory part of Hyperledger smart contract, and it clarifies which specific organization(s) must sign the transaction in order to make it valid.

3. Phase 3: CA will broadcast the ordered transaction to the rest of peer nodes to validate the transaction and then commit it to BBCVID. Applicant will get full access permit to this new ledger.

By similar token, in *VS*, a transaction is made to obtain a border crossing permit, and a immutable, time stamped digital record of the border crossing event for future references. To put more in detail, a valid *VS* transaction is defined as in below steps:

1. Phase 1: Border crossing permit applicants submit a signed transaction proposal to BA through applicant's client by connecting to BA's gateway service. BA executes the transaction proposal in accordance with the smart contract, and re-direct applicant's transaction proposal to the rest of relevant endorsement organizations. The rest endorsement organizations execute the smart contract as well and digitally sign a response that is going to send back to applicant's client.
2. Phase 2: Applicant's client packs all signed responses into an envelope and then let applicant to digitally sign on it. The client then submit the signed envelope to VS BA for validation, and the client receives a 'success' message if the submission is successful. BA validates all digital signatures in the envelope and send the client a QR code if all signatures are verified. QR code will initiate another smart contract at e-gate that is to conduct fingerprint identity authentication. After that, the e-gates will send a signed response back to client, and the client will submit it to BA after the applicant signs on the response.
3. Phase 3: BA then finally validate all responses and signatures of the transaction, and then order it to a new block of VS Blockchain. BA broadcast it to the rest of BA peer nodes to validate the transaction as well, and then finally commit it to the VS. The applicant will receive full access to this ledger record for future reference.

Consensus Rule

Different from the majority of the public Blockchain whose consensus algorithms are probabilistic, private Blockchain's consensus rule normally is deterministic. That is to say, both the order and the validity and correctness of a transaction is guaranteed and finalized. Therefore, consensus rule in private Hyperledger Blockchain is also called ordering service, which has the benefit of extremely difficult to fork[177], [233].

To put more in detail, Both BBCVID and VS employ *Raft*[233] as consensus rule. Specifically, Raft has followers who replicate the elected leader's decision of the Blockchain [177]. It is famous for its simplicity and efficiency compared with

other private Blockchain network consensus such as Paxos [233]. What is more, all organizational entities are publicly listed and bounded by strict smart contractual obligations to behave “in order”, and hence more efficient consensus algorithms [233]. It indicates that each organizational entity has at least one full node, which a full node can be reconfigured and deleted.

Transaction Privacy and System Security

It has been discussed in chapter 6 that since both proposed e-passport and border crossing history are private and confidential information; therefore, the privacy of the information should be well preserved. By similar token, the system security should be able to prevent impersonation and protect the Blockchain network from general threats. To fulfil these two tasks, below design and manoeuvres are taken:

- **Channels and collections for privacy data [176].** Channels in Hyperledger Fabric are private layers of communications between specific members within the network. To put more in detail, only invited and permitted entity can use the channel to interact with other permitted entities. That is to say, the access control to the *entire* Hyperledger transaction can be managed by building up channels among different organizations. Plus, all invited organization identity will be whitelisted in channel configuration file, which is publicly available to all participants of the system. Therefore, any misconduct of channel authorized entity can be distinguished straightaway, which makes the system more secure.

By similar token, collections belong to part of Hyperledger Fabric smart contract, which is used particularly to further control private data access *within* a transaction. That is to say, collection defines a subset of channel defined entities who is to further permitted to access the private data. To be more precise, collection defined private data access is only for the specific entity or peer who requires to endorse, commit, or enquiry the private data without creating a separate channel. Specifically, a smart contract collection is constructed by ‘.json’ file and inserted into smart contract by its name or ID number. To make a short summarize, two aims are fulfilled by imposing channels and collections, which are:

1. The private data either as in the entire transaction or as in a part of a transaction can be well preserved.
 2. The peer communication and peer behaviour within one channel is well protected and supervised.
- **Asymmetric encryption and ‘Salt’.** Both BBCVID and VS are deploying dissemination gossip protocol [176] for the actual private data dissemination,

and a hash(or an encrypt) of that private data will be used for endorsement, order, and recorded to Blockchain. To put more in detail,

1. Private data is not included in the transactions that submitted to system orderer; therefore, private data is not available to peers who are going to valid the transaction but only the particular node endorsed the transaction. The endorsing nodes ensure the availability of the private data in the channel or collection. That is to say, collection entity is able to share the private data based on request, which in most cases are for disputes and audits.
 2. To prevent brute force attack where a fixed string is directly encrypted by hash functions or asymmetric encryption key, a ‘salt’ random string will be generated to mix with the original string.
 3. Private data can be deleted completely from all peers so that it is not available at all after transaction is recorded on chain by calling ‘PurgePrivateData’ function in smart contract, which is to withdraw all peer’s read and write access to applicant’s private data.
- **Blockchain network access control.** Since both BBCVID and VS are permissioned Blockchain, they are only available to participants whose identity has already known to the system; therefore, data access control and data origin can be well tracked and controlled by the system.
 - **Version checkups and revoke lists enquiry.** Versioning checkups before any ledger is committed to the Hyperledger is compulsory in case of Blockchain network forking. Plus, CA maintained ‘revoke lists’ which records all revoke item details is also compulsory in case of referencing a revoked item within the system. That is, both versioning checkups and revoke list enquiry are built in smart contracts of BBCVID and VS.

8.1.1 Decentralized Application(DApp): Android Mobile Application

To start with, Android mobile operation system runs on Linux kernel that connects its hardware with its software stack [509]. Its run-time consists of Dalvik virtual machine and Core Java library, which Dalvik virtual machine enables registered applications on Android to run simultaneously on as in Java multi-threading [451]. Most important, Android requests applications is registered and signed under developer’s key, which secures the Android application in market place [509].

To preserve private data privacy and security, Android has built-in trusted execution environment, which indicates that the biometric information is encrypted

and stored in a separated part of the Android smart phone [159]. Most important, it is completely inaccessible to the regular operating system. Apart from TEE, Android has Sandbox for untested program for un-authorized access [451]. To make the system even more secure and privacy-preserving, “permissions” are used in Android whenever an access to sensitive and protected information is raised, such as GPS location and open camera etc. If user does not grant the permission, the access will be denied straightaway [451].

The developed Android mobile application is design for all border crossing permit and e-passport applicant to submit these application at their mobile end device, so that digital verifiable border crossing permit and digital verifiable biometric e-passport can be obtained. To put more details in, there are two steps to be completed before e-passport and border crossing application can be made, which specifically are:

Step One: User Registration to Obtain Access to the Mobile Application

To start with, the Android mobile application access is granted to registered user only. That is to say, user registration is compulsory so that an access to the mobile client can be obtained, see **Figure 7.2** for a demonstration of the Android mobile application user login page, **Figure 7.3** for new user registration page; **Figure 7.4** new user registration success notification, **Figure 7.5** for user database in Google Firebase, and **Figure 7.6** for Google Firebase email authentication.

To add on, user name, email address and a six-digits PIN number are required to complete the new user registration process. To secure our application, access to our application is further secured by fingerprint biometrics identity authentication by using Android mobile built-in library. Most important, apart from individual person, commercial business entity can also be a user of our mobile application, and it is assumed that it is the business owner who takes full control of our application’s user account.

Step Two: Obtain Public Key Certificate from A Certificate Authority

After obtain the access to mobile application, the access to BBCVID and VS Blockchain should be obtained in the step two. To put more details in, the Hyperledger Fabric certificate authority(Fabric CA) has three main functions in our proposed system, which include:

1. User identity connects to lightweight directory access protocol(LDAP) as the user registry.
2. Identity certificate issuance for users, and TLS certificate issuance for nodes and clients.

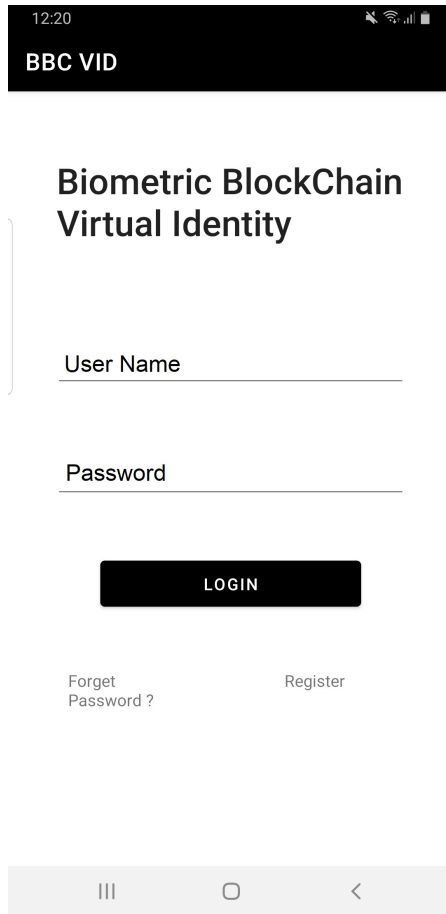


Figure 8.2: User Login.

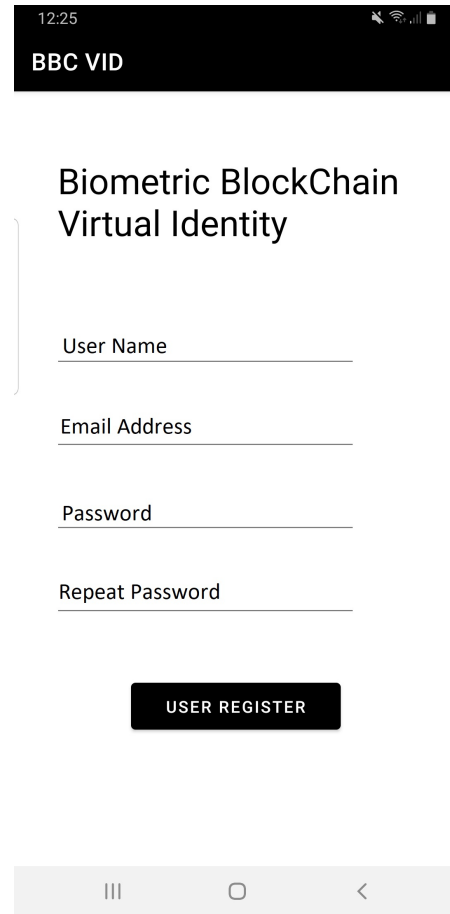


Figure 8.3: New user registration.

3. Certificate renewal, revoked, and management.

What is more, Fabric CA issued public key identity certificate does not only have public key of the user, but also have identity attributes that defines the user's role in the Blockchain. Therefore, by obtaining the public key certificate, not only user's access to the Blockchain is well controlled, but also Blockchain resources and data access control can be well managed. Plus, TLS certificate enables signed and encrypted communications between nodes and clients.

In our Android mobile application, mobile client connects to Blockchain gateway service through a Hyperledger peer node, which specifically a high performance remote procedure call(gRPC) will be dialled. That is to say, gRPC enables message transfer between mobile client and the BBCVID and VS Blockchain, and broadcasting transactions among all peers nodes etc. Specifically, border crossing permit applicants and e-passport applicants are advised to use their mobile client as Fabric CA client,

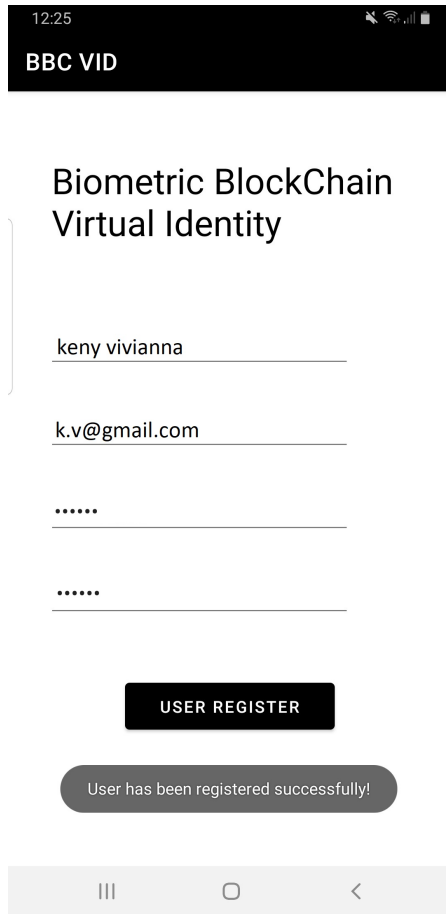


Figure 8.4: New user registration success.

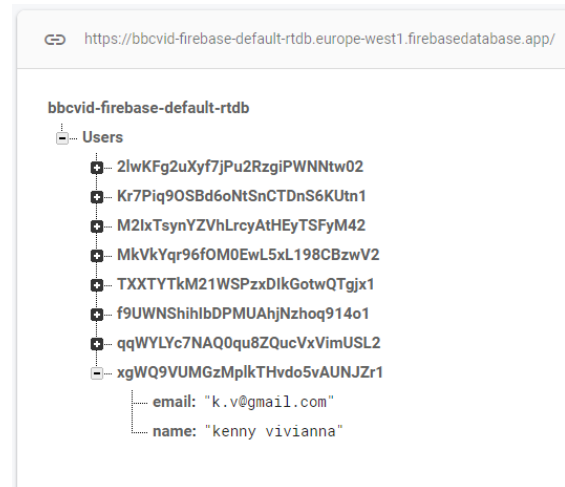


Figure 8.5: User database in Google Firebase.

which the Fabric intermediate server connects to the Fabric CA root server through the Fabric CA client. Specifically, Fabric intermediate server will use the user registration information in step one as user registry to LDAP, and issue them identity certificate accordingly.

For a short summary for both step one and two, new user inputs “userName”, “userEmail”, and “password” three variables for mobile application user registration. Mobile client connects to Google Firebase system for backend management. Mobile client then sends user information as a JSON object to BBCVID gateway service through the client. For example, one registered user information ‘{“UID”: “qqWYLYc7NAQ0qu8ZQucVxVimUSL2”, “userName”:“kennyvivianna”, “userEmail”: “k.v@gmail.com”}’ is sent to BBCVID certificate authority node admin for public key certificate to obtain access to proposed BBCVID and VS Hyperledger Blockchain. BBCVID certificate authority returns { “pubKey”: “”, “priKey”:“”

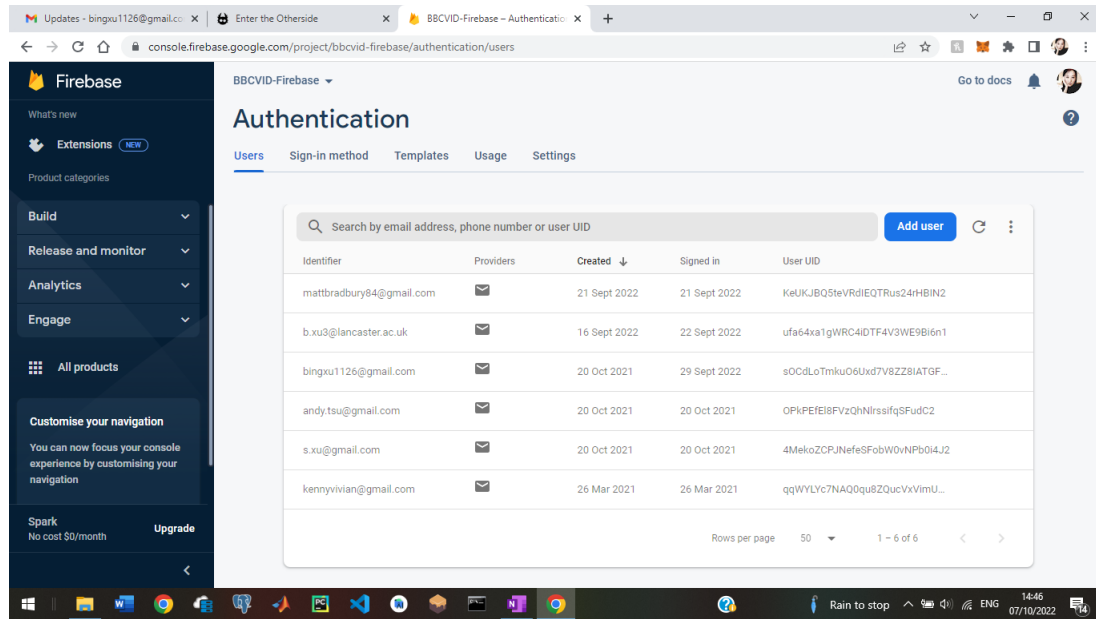


Figure 8.6: Google Firebase email authentication scheme.

back to the user and record $\{“UID”：“qqWYLYc7NAQ0qu8ZQuCVxVimUSL2”$, $“pubKey”：“ ”$, $“priKey”：“{pk}”\}$ in its own database. Fabric CA links new user’s public key certificate with mobile application user registry through “UID” variable.

The new registered user upon receiving CA’s public key identity certificate, they are required to sign on his public key by his private key and return to Fabric CA, which aims to prove that the user has control over the private key of its corresponding public key. So far, new user’s access to both mobile application and BBCVID and VS Hyperledger Blockchain are obtained.

BBCVID and VS Blockchain Configuration

In accordance with the BBCVID and VS Blockchain properties discussed above and the transaction flow defined above, BBCVID and VS Blockchain are configured accordingly as in below:

- **Consensus rule.** BBCVID consensus rule sets up as basic Raft through ‘configtx.yaml’ file configurations, which specifically EtcdRaft is adopted. EtcdRaft is a Raft package which has very established structure to define the class elements. Below is a demonstration of configured BBCVID Raft consensus through re-define ‘configtx.yaml’ file.

EtcdRaft :

Consenters :

- Host: orderer.bbcvid.com
- Port: 7050
- ClientTLSCert: ../orderer.bbcvid.com/tls/server.crt
- ServerTLSCert: ../orderer.bbcvid.com/tls/server.crt

- **Channels and Organizations.** Initial local built Hyperledger test network only has two organizations without any channel at all. To modify it into BBCVID and VS, channels and organizations are required to be re-constructed.
 - **BBCVID.** One channel “bbcvidchannel” is created to share among DVLA, HMRC and CA three independent peer organizations, see **Figure 7.7** for Ubuntu command line log records for channel ‘bbcvidchannel’ created record. Specifically, CA is the orderer node, and only one peer node is assigned to each organization, see **Figure 7.8** for Docker container demonstration of each peer nodes.
 - **VS.** One channel “vschannel” is created to share among DVLA, HMRC, Surveillance system, and BA four independent peer organizations. Specifically, BA is the orderer node, and only one peer node is assigned to each organization.
- **Certificate Authority Configuration.** Certificate authority controls the Hyperledger Fabric certificate root server, who issues digital public key certificates to participants and TLS certificate to nodes and clients through the Fabric CA client. The communication between certificate authority root server and certificate authority client is enabled by REST APIs. To decentralized certificate management, each peer node in each organization is assigned local certificate authority roles as well. Like for instance, in BBCVID, apart from the central authority who is also the certificate authority of the BBCVID Blockchain, HMRC peer0 node and DVLA peer0 are both assigned certificate authority roles. By similar token, VS orderer, HMRC peer0 and DVLA peer0 are all nominated certificate authority in VS Blockchain. Most important, BBCVID HMRC peer0 is not necessarily same node with VS HMRC peer0, neither does BBCVID DVLA peer0 and VS DVLA peer 0. See **Figure 7.9** for BBCVID HMRC and DVLA peer node tree structure after certificate authority configuration, which tls-cert.pem, ca-crt.pem, IssuerPublicKey, IssuerRevocationPublicKey, and msp are generated.

Smart Contract Deployment and Invoking


```

/home/bing/go/src/github.com/fabric-samples/test-network/./bin/configtxgen
+ configtxgen -profile TwoOrgsApplicationGenesis -outputBlock ./channel-artifacts/bbcvidchannel.block -channelID bbcvidchannel
2023-05-09 13:03:03.977 851 8001 INFO [common.tools.configtxgen] main -> Loading configuration
2023-05-09 13:03:03.991 851 8002 INFO [common.tools.configtxgen.LocalConfig] completeInitialization -> orderer type: etcdraft
2023-05-09 13:03:03.992 851 8003 INFO [common.tools.configtxgen.LocalConfig] completeInitialization -> Orderer.Start.Options unset, setting to tick_interval:"500ms" election_tick:10 heartbeat_tick:1 max_inf
light_blocks:5 snapshot_interval_size:16777216
2023-05-09 13:03:03.992 851 8004 INFO [common.tools.configtxgen.LocalConfig] Load -> Loaded configuration: /home/bing/go/src/github.com/fabric-samples/test-network/configtx/configtx.yaml
2023-05-09 13:03:03.997 851 8005 INFO [common.tools.configtxgen] doOutputBlock -> generating genesis block
2023-05-09 13:03:03.997 851 8006 INFO [common.tools.configtxgen] doOutputBlock -> Creating application channel genesis block
2023-05-09 13:03:03.999 851 8007 INFO [common.tools.configtxgen] doOutputBlock -> Writing genesis block
+ res=B
Creating channel bbcvidchannel
Using organization 1
+ osnadmin channel join --channelID bbcvidchannel --config-block ./channel-artifacts/bbcvidchannel.block -o localhost:7053 --ca-file /home/bing/go/src/github.com/fabric-samples/test-network/organizations/order
erorganizations/example.com/tlsca/tlsca.example.com-cert.pem --client-cert /home/bing/go/src/github.com/fabric-samples/test-network/organizations/ordererorganizations/example.com/orderers/orderer.example.com/t
ls/server.crt --client-key /home/bing/go/src/github.com/fabric-samples/test-network/organizations/ordererorganizations/example.com/tls/server.key
+ res=B
Status: 201
{
  "name": "bbcvidchannel",
  "url": "/participation/v1/channels/bbcvidchannel",
  "consensusRelation": "consenter",
  "status": "active",
  "height": 1
}
Channel 'bbcvidchannel' created
Joining org1 peer to the channel...
Using organization 1
+ peer channel join -b ./channel-artifacts/bbcvidchannel.block
+ res=B
2023-05-09 13:03:10.303 851 8001 INFO [channelCmd] InitCmdFactory -> Endorser and orderer connections initialized
2023-05-09 13:03:10.401 851 8002 INFO [channelCmd] executeJoin -> Successfully submitted proposal to join channel
Joining org2 peer to the channel...

```

Figure 8.7: Ubuntu command line logs for acknowledging bbcvid channel is created.

```

[+] Running 3/8
  Network fabric_test          Created           0.1s
  Volume "compose_orderer_bbcvid.com" Created           0.0s
  Volume "compose_peer0_dvla_bbcvid.com" Created           0.0s
  Volume "compose_peer0_hmrc_bbcvid.com" Created           0.0s
  Container peer0_hmrc_bbcvid.com Started           1.9s
  Container orderer_bbcvid.com Started           2.2s
  Container peer0_dvla_bbcvid.com Started           2.2s
  Container cli Started           2.0s
CONTAINER ID   IMAGE                                COMMAND          CREATED        STATUS        PORTS                NAMES
76971d4808c   hyperledger/fabric-tools:latest     "/bin/bash"     3 seconds ago Up Less than a second          cli
946556a11ff   hyperledger/fabric-peer:latest     "peer node start" 3 seconds ago Up 1 second          peer0.dvla_bbcvid.com
91e88a8e85d   hyperledger/fabric-orderer:latest  "orderer"        3 seconds ago Up 1 second          orderer_bbcvid.com
0b8568f2383e  hyperledger/fabric-peer:latest     "peer node start" 3 seconds ago Up 1 second          peer0_hmrc_bbcvid.com

```

Figure 8.8: Docker image lists for all peer nodes in BBCVID.

In Hyperledger Fabric Blockchain, smart contract is deployed in packages, which requires all organizations who are intend to validate or makes an enquiry about the transaction to deploy the corresponding smart contract on their peer node [175]. To put more in detail, a smart contract is deployed to a specific node first, and then channel members of that node belongs to are able to deploy that smart contract to the channel, which the process is also known as the Fabric chaincode life-cycle. That is to say, since smart contract is deployed to organization’s node first, it allow all organizations to agree on and approve the definition and policies in the smart contract first, and then deploying it to the channel where all members have to obey ever after.

Specifically, organization peer node is managed by node administrator(admin), who controls the node client and server. After the smart contract package installed in the peer node, the smart contract definition and policy has to be approved by the peer node affiliated organization, which “LifecycleEndorsement” policy defines who needs to approve a particular smart contract. To put more in detail, the ‘smart contract definition’ has operative parameters such as smart contract name, version, and endorsement policy, and ‘endorsement policy’ only has the organizations name who are required to endorse the smart contract initiated transaction. Plus, a ‘packageID’ will be generated and associated with approved smart contract definition,

```

bing@in9phd281906:~/go/src/github.com/fabric-samples/test-network$ tree organizations/fabric-ca/dvla
organizations/fabric-ca/dvla
├── IssuerPublicKey
├── IssuerRevocationPublicKey
├── ca-cert.pem
├── fabric-ca-server-config.yaml
├── fabric-ca-server.db
├── msp
├── cacerts
├── keystore
│   ├── 1ebcaf1dfee890c084ddf7b3ac382f199a99c54a4190be1dfc020b1f56a0a0ee_sk
│   ├── 828e135d19ce68d3abb792c6db6f076a942958959cd3f1fc17d6e40e9336edfd_sk
│   ├── IssuerRevocationPrivateKey
│   └── IssuerSecretKey
├── signcerts
├── user
└── tls-cert.pem

5 directories, 10 files
bing@in9phd281906:~/go/src/github.com/fabric-samples/test-network$ tree organizations/fabric-ca/hmrc
organizations/fabric-ca/hmrc
├── IssuerPublicKey
├── IssuerRevocationPublicKey
├── ca-cert.pem
├── fabric-ca-server-config.yaml
├── fabric-ca-server.db
├── msp
├── cacerts
├── keystore
│   ├── 43af9199859ead323ccd98ca45fbd3e86ad8d7bc29a7add42cac5618cc021d7_sk
│   ├── IssuerRevocationPrivateKey
│   ├── IssuerSecretKey
│   └── f49b199fdb614f754b67ec931fbc0b64b77f97a9436a2fd134122b1fd44e24e_sk
├── signcerts
├── user
└── tls-cert.pem

5 directories, 10 files
bing@in9phd281906:~/go/src/github.com/fabric-samples/test-network$ |

```

Figure 8.9: BBCVID HMRC and DVLA tree structure after certificate authority configuration: `tls-cert.pem`, `ca-cert.pem`, `IssuerPublicKey`, `IssuerRevocationPublicKey`, and `msp` are generated.

which can be queried by ‘peer lifecycle chaincode queryinstalled’ command.

After all organization members approved the peer node installed smart contract package, the smart contract is ready to deploy on the Blockchain channel. To put more in detail, one organization will take the responsibility to commit the approved smart contract to the channel by ‘peer lifecycle chaincode commit’ command.

As for invoking a smart contract, it normally is initiated by client application(our Android mobile application) by referencing the name of the smart contract and the channel, and sent to peer node where it was installed. In accordance with the endorsement policy of the smart contract, the client application targets sufficient number of peer transaction request being sent out.

8.2 System Simulation

In this section, we will look at in detail about how the proposed biometric e-passport is generated, and how to use it as a credential to get border crossing legitimacy approved over open internet. Most important, a real walk through of this proposed process will be simulated by writer’s own personal data.

8.2.1 BBCVID: Biometric E-passport

A: Biometric DID

After obtaining the access to the proposed mobile application and BBCVID and VS Hyperledger Blockchain, transactions can be initiated at user's client by user. For biometric e-passport, user has to submit e-passport application first and then visit biometric identity collection points where is nominated by the central authority of BBCVID (BBCVID CA) for a biometric DID.

See **Figure 7.10** for biometric e-passport application form in the proposed Android mobile application, and the "get your location" component in Figure 7.10 is used to fetch applicant's current fine location through `Android.gms.maps.GoogleMap`, which aims to reduce the risk of impersonation slightly.

BBCVID CA invites applicant to visit the biometric identity collection points if applicant's application is approved. After applicant visiting the biometric identity collection point, BBCVID is able to link applicant's application form, facial and fingerprint biometric identity template, and the applicant's public key altogether by facial identity authentication through mobile application. See **Figure 7.11** for initial biometric identity authentication between e-passport applicant and BBCVID CA, **Figure 7.12** for the BBCVID generated biometric DID documentation.

To put more in detail, the `{ "id": "did:bbcvid:2wJNgSLfLLnYTEFYzByfRM" }` is the DID in referent of the bodily person (applicant), which a biometric DID documentation is resolved from it (Figure 7.12). The applicant proves they have full control over it by the three "authentication" method defined in DID document, which include public key, facial image and fingerprint biometric identity authentication. That is, applicant's both facial and fingerprint identity templates can be resolved from the biometric DID document through URI `{ "id": "did:bbcvid:2wJNgSLfLLnYTEFYzByfRMhash1" }` and `{ "id": "did:bbcvid:2wJNgSLfLLnYTEFYzByfRMhash2" }`. Most important, the applicant biometric identity template indeed is also resolvable in accordance with `{ "id": "did:bbcvid:2wJNgSLfLLnYTEFYzByfRMhash1" }` (facial image template) and `{ "id": "did:bbcvid:2wJNgSLfLLnYTEFYzByfRMhash2" }` (fingerprint). However, these two URIs are *not* resolvable to everyone, but require DID subject to sign on the particular access request to give permission. Specifically, it is fulfilled by the smart contract private collection definition and transient data operation in Hyperledger smart contract.

What is more, the veracity and integrity of this document can be verified by "proof" component in the biometric DID document, which include the "verification-method" resolves issuer's public key, "type" specifies the verification algorithm, and the "proofvalue" is the digital signature value of this document. That is to say, the biometric DID is a digital verifiable biometric identity credential that can be used to conduct biometric identity authentication over open internet. However, biometric

DID only binds a bodily person with *a DID string only*.

Figure 8.10: Biometric e-passport application form.

Figure 8.11: Facial biometric identity authentication and applicant's current location.

B: Biometric E-passport

After obtaining the biometric DID, generating the proposed biometric e-passport is to further bind applicant's *legal soft identity* with the biometric DID through BBCVID smart contracts, see **Figure 7.13** for the proposed e-passport representation in Android mobile application and **Figure 7.14** for the corresponding e-passport DID documentation.

To put more in detail about e-passport document, the first component is the information about this biometric e-passport itself. That is, `{ "id": "did:bbccvid:1234abcdef`

```

C:\> newpro > {} bioDID.json > ...
1
2 {
3   "@context": [
4     "https://www.w3.org/ns/did/v1",
5     "https://w3id.org/security/suites/ed25519-2020/v1"
6   ],
7   "id": "did:bbcvid:2wJNgSLfLLnYTEFYzByfRM",
8   "authentication": [
9     {
10      "id": "did:bbcvid:2wJNgSLfLLnYTEFYzByfRMkey-1",
11      "type": "Ed25519VerificationKey2020",
12      "controller": "did:bbcvid:2wJNgSLfLLnYTEFYzByfRM",
13      "publickeyMultibase": "ZHC2AVvLhVgM#am3UVAjZpFkc3CvDm2n6z3uXmqPV",
14    },
15    {
16      "id": "did:bbcvid:2wJNgSLfLLnYTEFYzByfRMhash1",
17      "type": "BiometricIdentity-1",
18      "controller": "did:bbcvid:2wJNgSLfLLnYTEFYzByfRM",
19      "smartContract": "facialbioauth",
20    },
21    {
22      "id": "did:bbcvid:2wJNgSLfLLnYTEFYzByfRMhash2",
23      "type": "BiometricIdentity-2",
24      "controller": "did:bbcvid:2wJNgSLfLLnYTEFYzByfRM",
25      "smartContract": "fingerbioauth"
26    }
27  ],
28  "credentialStatus": {
29    "id": "https://bbcvid.com/status",
30    "type": "CredentialStatusList"
31  },
32
33  "proof": {
34    "type": "Ed25519Signature2020",
35    "created": "2023-01-13T08:19:39Z",
36    "verificationMethod": "https://bbcvid.com/issuers/CAKey-1",
37    "proofPurpose": "assertionMethod",
38    "proofValue": "zSDAdFfa95kq2NvPxAqpic7ndSayn1PzZs6Zjhp1CktyGesjuTsWRdohAfgFCf5bopETStjQCfFPP2oumHKtz"
39  }
40 }
41
42

```

Figure 8.12: Biometric DID document.



Figure 8.13: The proposed e-passport representation in mobile application.

g567hijk”} is the DID of this e-passport documentation, which the e-passport DID document can be resolved from it. What is more, the type, issuer, issuerName, issuance data, and expire date of the e-passport are included.

The second component of the e-passport document is the “credentialSubject”, which defines e-passport owner’s identity attributes. Specifically, {“id”: “did:bbcvid:2wJNgSLfLLnYTEFYzByfRM”} is the subject of this DID document, which refers to the bodily applicant herself. That DID is resolvable to biometric DID document(Figure 7.12). Since permission request is compulsory, access request to both biometric DID document and biometric e-passport document has to be made separately even it is made to the same subject. In this component, the owner’s soft legal identity and the hash output of owner’s biometric identity template are added. Followed by “credentialStatus” and “proof” components, they are used for referent

```

C: > newpro > {} bioepassprt.json > {} credentialSubject
1  {
2    "@context": [
3      "https://www.w3.org/2018/credentials/v1",
4      "https://www.w3.org/2018/credentials/examples/v1"
5    ],
6
7    "id": "did:bbcvid:1234abcdefg567hijk",
8    "type": ["VerifiableCredential", "passport"],
9    "issuer": "did:bbcvid:CA#key-1",
10   "issuerName": "Consulate general of P.R.China in Manchester",
11   "issuanceDate": "2023-01-01T19:23:24Z",
12   "expirationDate": "2033-01-01T19:23:24Z",
13
14   "credentialSubject": {
15
16     "id": "did:bbcvid:2wJNgSLfLLnYTEFyzByFRM",
17
18     "lang": "en",
19     "givenName": "Bing",
20     "familyName": "Xu",
21     "dateOfBirth": "Nov. 26, 1990",
22     "gender": "Female",
23     "facialImage": "hash1",
24     "fingerprint": "hash2",
25     "placeOfBirth": "Liaoning"
26   },
27
28   "credentialStatus": {
29     "id": "https://bbcvid.com/status/",
30     "type": "CredentialStatusList"
31   },
32
33   "proof": {
34     "type": "Ed25519Signature2020",
35     "created": "2023-01-13T18:19:39Z",
36     "verificationMethod": "https://bbcvid.com/issuers/CA#key-1",
37     "proofPurpose": "assertionMethod",
38     "proofValue": "z58DAdFfa9SkqZMVPxAQpic7ndSayn1PzZs6ZjWp1CktyGesjuTSwRdo
39     | | | | | WhAfGFCF5bppETStojQCrfFPP2oumHKtz"
40   }
41 }
42 }

```

Figure 8.14: The proposed e-passport DID documentation.

CA identity revoke list and the veracity and integrity authentication mechanism of the e-passport document.

Compared with biometric DID document, biometric e-passport DID document adds on e-passport owner's legal soft identity attributes such as name, date of birth, gender, and place of birth etc, which enables that a bodily person is not only bound with a DID string but also those legal soft identity. The tie between the DID string, the bodily person, and the legal soft identity is fastened by BBCVID Hyperledger

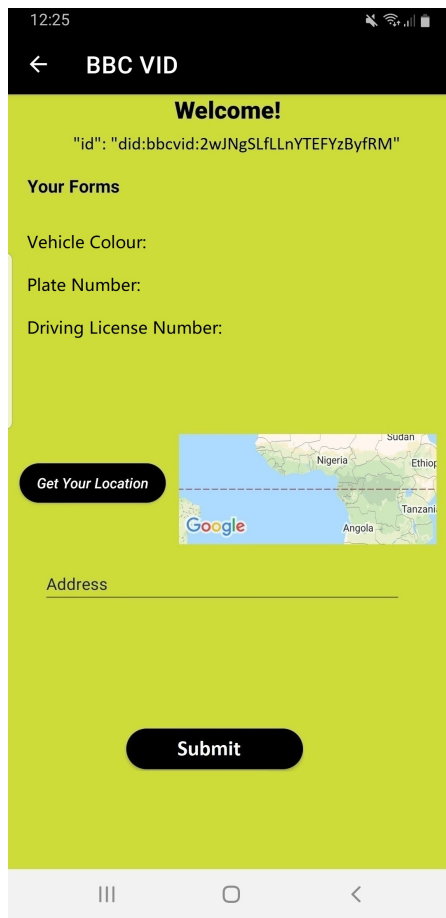


Figure 8.15: Vehicle DID application form.



Figure 8.16: The proposed vehicle DID document representation in mobile application.

Fabric Blockchain immutable transaction. The bodily person can prove her ownership toward the DID and those legal soft identity attribute by public key, facial image and fingerprint identity authentication(Figure 7.11).

C: Digital Verifiable Biometric Credentials

Obtaining biometric DID and biometric e-passport, owner of these two documents is able to make digital verifiable biometric credential applications to HMRC and DVLA, which aims to bind customs clearance and vehicle information with the proposed biometric DID.

For example, the proposed e-passport owner can make an application of vehicle

```

C: > newpro > {} vehicleDIDdox,json > {} credentialSubject
1  {
2    "@context": [
3      "https://www.w3.org/2018/credentials/v1",
4      "https://www.w3.org/2018/credentials/examples/v1"
5    ],
6
7    "id": "did:bbcvid:Eh8mWPdx23GVil65mnsePF",
8    "type": ["VerifiableCredential", "vehicleDID"],
9    "issuer": "did:bbcvid:CA#key-1",
10   "issuerName": "DVLA",
11   "issuanceDate": "2023-05-14T17:13:20Z",
12   "expirationDate": "2033-05-13T17:13:19Z",
13
14   "credentialSubject": {
15
16     "id": "did:bbcvid:2wJNgSLfLLnYTEFYzByfRM",
17
18     "lang": "en",
19     "vehicleColour": "red",
20     "plateNumber": "mw72cru",
21     "drivingLicenseNo": "xu999861267B99EA11"
22   },
23
24   "credentialStatus": {
25     "id": "https://bbcvid.com/status/",
26     "type": "CredentialStatusList"
27   },
28
29   "proof": {
30     "type": "Ed25519Signature2020",
31     "created": "2023-01-13T18:19:39Z",
32     "verificationMethod": "https://bbcvid.com/issuers/CA#key-1",
33     "proofPurpose": "assertionMethod",
34     "proofValue": "z58DAAdFfa9SkqZMVPxAQpic7ndSayn1PzZs6ZjWp1CktyGesjuTSwRdo
35     | | | | | WhAfgFCF5bPPETStojQCrfPPP2oumHKtz"
36   }
37 }
38 }

```

Figure 8.17: The vehicle DID documentation.

DID to BBCVID channel organization DVLA, see **Figure 7.15** for the vehicle DID application form, **Figure 7.16** for the vehicle DID document representation in the proposed Android mobile application, and **Figure 7.17** for the vehicle DID document.

To put more in detail, vehicle owner signs the encrypted data collected in the application form figure 7.15 by proposed data exchange algorithm in chapter 6.3.2. The mobile client sends it to DVLA node to initiate the vehicle DID generation smart contract. After all data verified, vehicle DID and vehicle DID document is generated as in Figure 7.17 and its representation is demonstrated in Figure 7.16. Specifically, the DID {“id”: “did:bbcvid:Eh8mWPdx23GVil65mnsePF”} is in referent of the vehicle DID document, and the DID {“id”: “did:bbcvid:2wJNgSLfLLnYTEFYzByfRM”}

12:23

← **BBC VID**

Welcome!

"id": "did:bbcvid:2wJNgSLfLLnYTEFYzByfRM"

Your Forms

Driver's E-passport No.: _____

Driver's Visa No.: _____

Vehicle Plate No.: _____

Commercial Invoice No.: _____

Package List No.: _____

Get Your Location

Address _____

Border Crossing Permit Application Submit

Figure 8.18: Commercial border crossing permit application form.

12:23

← **Border Crossing Records**

Welcome!

"id": "did:bbcvid:2wJNgSLfLLnYTEFYzByfRM"

Border Crossing History:

Record 1:

Application No.:

Checkpoints Attended:

Live Data Record:

Time of Border Crossing:

Issuer :

Digital Signature:

Record 2:

Application No.:

Checkpoints Attended:

Live Data Record:

Time of Border Crossing:

Issuer:

Digital Signature:

Figure 8.19: The proposed VS border crossing records representation in mobile application.

is in referent of the subject of the vehicle who is the controller of the vehicle. DVLA reconciliates the form data with its own database by checking the “driving license” out. Further DVLA requires identity authentication with the applicant by the DID recorded in the “credentialSubject” component through the mobile application. Finally, the vehicle DID and vehicle DID document can be generated.

If DVLA requires applicant’s legal soft identity such as legal name, date of birth, and gender etc, DVLA can request access to applicant’s biometric e-passport and then bind the legal soft identity information in the vehicle DID document.

Indeed, our proposal can be used as a model to generate general digital verifiable biometric credential. Starting with the biometric DID and its DID document, binding it with legal soft identity, the biometric DID document become the proposed e-

passport. By the same token, binding the biometric DID and its DID document with vehicle information, the biometric DID document becomes digital biometric verifiable vehicle DID and vehicle DID document. So does HMRC digital biometric verifiable customs clearance documentations.

8.2.2 VS: Virtual Stamping for Border Crossing Events

In this section, we will see how an individual immigrant and a commercial business entity use the proposed e-passport and other digital biometric verifiable credentials to get border crossing permit over the open internet and obtain a immutable and time stamped records of the border crossing event.

Border Crossing Permit

To start with, some key concepts in border crossing has to be emphasized. That is,

- The border crossing permit is granted at the border checkpoint e-gate, where as the final step of the entire border crossing event, fingerprint identity authentication through the proposed e-passport is required at e-gate.
- An indispensable and crucial part of the entire border control process is to reconcile data in border crossing permit application with the live in-situ data that collected by Surveillance system. Specifically, surveillance system is one organization that is authorized to join VS Hyperledger Fabric Blockchain “vschannel”; therefore, all the surveillance system’s peer node clients have public key identity certificate and TLS certificate for secure and private communications within the “vschannel”. Most important, surveillance system collected live data will packed in JSON data object, signed by surveillance system node client, and recorded on VS Blockchain. For an example of surveillance system captured vehicle live in-situ data at road border checkpoint unit, it is {“clientPubKey”:“ ”, “plateNumber”:“mw72cru”, “vehicleColour”:“red”, “weight”: 1250, “RFIDeSeal”:“permitGranted”}.

For commercial business entity applies to border crossing permit, the commercial business entity submits the border crossing permit application through mobile client(see **Figure 7.18**), and sends the signed application data to VS relevant node gateway service for initiating the transaction and request endorsement. Specifically, DVLA, HMRC, BA, and the surveillance system are all required to endorse the transaction to make it valid. To put in more detail, taking DVLA as an example, in accordance with the “vehicle plate number”, DVLA checks the vehicle out about its border crossing legitimacy and if there is any reported lost or any illegal records. Then

DVLA will authenticate driver's identity by the proposed e-passport through mobile application and driver's border crossing legitimacy by driver's visa. If everything goes well, DVLA will endorse the transaction and return it back to applicant's mobile client. By the same token, HMRC will auto verify the customs clearance documentation veracity and integrity first as soon as application's commercial invoice number and package list number available. If all those digital verifiable document is verified, HMRC will endorse the transaction and return it back to applicant's mobile client.

Similarly, the surveillance system will collect border crossing live in-situ data and sends to VS BA after endorses the transaction. Indeed, the surveillance system collects data at road border checkpoint unit where is set about 2 kilometers away from the border. It consist of road cameras to capture vehicle driving plate number and vehicle colour, then road weight sensor straightaway after the CCTV camera to capture the weight of the vehicle, finally is the RFID reader to get the customs e-seal number on the vehicle. In the process of getting border crossing permit, the road surveillance unit data is necessary. The surveillance system will captures the data, endorse the transaction, and returns it to applicant's mobile client. BA is also required to endorse the transaction but it is set to be the last one who should make the endorsement. BA will check all endorsement signatures and matching the endorsement data. If all data is conciliated, BA will offer ordering service for it, broadcast it to the rest of peer nodes to validate it, commit it to the VS Hyperledger Blockchain, and finally generate a QR code which contains driver's DID, the Hyperledger transaction ID, and BA's digital signature for driver to use it as the proof of the border crossing permit to pass through the e-gates after fingerprint identity authentication. To put it more precise, the QR code that generated by the VS BA is the final border crossing permit VS BA issued to the applicant.

That is to say, so far, the commercial vehicle loaded with commercial products is able to cross the border with verified legitimacy proof, which the process of obtaining the border crossing permit is fully automatic after the border crossing permit application is submitted over the mobile application. However, this process can be ceased at any of these steps if there is any suspicions information captured. Like for instance, if a vehicle's driving plate number does not exist in any of the border crossing permit applications, the vehicle will be stopped at the checkpoints for further inspections.

By the same token, for individual immigrants to cross the border with a private vehicle or just on her own at the airports, the border crossing permit can also be made through the mobile application to VS BA. After obtaining all endorsement from relevant organizations and validating all the signatures, VS BA is able to issue the QR code as a proof of the border crossing permit and recorded it on the VS Blockchain.

Immutable, Time-Stamped Border Crossing Event Records

Before the individual conducting fingerprint identity authentication at the e-gate system, the individual should obtain the border crossing permit already. To put more details in, the border crossing permit is represented by a QR code and recorded as a immutable ledger on the VS Blockchain.

To make it more precise, the specific border crossing event refers to individual using the border crossing permit to get across the border e-gate. To initiate the border crossing immutable time-stamped border crossing record transaction, the individual should present the border crossing permit(RQ code) that is issued and digitally signed by VS BA at the e-gate. The e-gate belongs to VA BA's organization, which reads the QR code, fetches the fingerprint template, and conducting the fingerprint identity authentication. If VS digital signature is verified and the fingerprint identity is authenticated, e-gate will be open. E-gate signs on the transaction and offers ordering service for it, broadcast it to other peers for validation, and finally commit it to the VS Blockchain. The border crossing event is time stamped as an immutable ledger on VS Blockchain.

That is so say, since this transaction does not only contains the fingerprint identity authentication outcome but also the transaction ID of the border crossing permit, VS BA is able to connect the border crossing permit information with an individual person, and represent the border crossing event records in the mobile application as in **Figure 7.19**.

8.3 Performance Evaluation

In this section, the entire system performance will be evaluated and discussed in the regard of as an automatic border crossing system and as a Blockchain identity management system respectively based on industry[328], [389] and government technical reports[365] recommended evaluation scheme and assessment criteria.

8.3.1 Performance of the BBC ABC System

In accordance with [389], soft border system can be analyzed through cost and leveraging existing system criteria. For our proposed BBC ABC system, it can also be analyzed by the same criteria. Specifically,

- **Cost.** Cost effectiveness is encouraged to take into account when analyses the performance of a new ABC system [389]. In this project, costs are occurred at three main components, which specifically are Blockchain system costs, computational cost, and infrastructure costs. To put more in detail,

- **Blockchain cost.** To build up a Blockchain system from bare metal, there are several specific costs to be considered and evaluated. For the proposed BBCVID and VS Hyperledger Blockchain system, the Blockchain cost include below two main elements, specifically,
 1. **Transaction cost.** Blockchain as a distributed network constructed from open source codes, the system security is always crucial. In chapter 5, it has been well explained that public Blockchain like Bitcoin and Ethereum Blockchain charge transaction fees and gas fee to protect system security and constrain the public resources wastage. However, Hyperledger Fabric Blockchain as private Blockchain, it uses user identity authentication tactic to secure the system security, which makes zero transaction fee possible. That is to say, transaction has zero fee for all operations on Hyperledger Fabric regardless energy costs[328], compared with Bitcoin Blockchain has an average of USD 5.10 and Ethereum Blockchain has an average of USD 4.19 per transaction fee in March 2023 [171].
 2. **Running a full node cost.** Both BBCVID and VS are Hyperledger Fabric private Blockchain. Therefore, to run a full node, there are some minimum requirements in terms of hardware and network performance [237]. Specifically, since each full nodes has a potential of making an impact on the entire Blockchain network performance in accordance with consensus rule and endorsement policy, persistent and fastest disk storage, a minimum of 1 Gbps network connection between all nodes and organizations, and 1 CPU with 2 GB of memory for orderer nodes are all encouraged[237].
- **Computational cost.** The computational costs in the proposed BBC ABC system include digital verifiable documentation verification, biometric identity authentication through proposed e-passport, border crossing permit generation, and border crossing event recording. Specifically,
 1. **Digital verifiable documentation including proposed e-passport, visa, and customs clearance documentations etc.** To be more precise, in this proposal, Edwards-Curve Digital Signature Algorithm (EdDSA) [261] is used to generate the digital signature attached to all BBCVID generated digital verifiable document, as this algorithm is recommended by W3C DID core [460] and EdDSA Cryptosuite [461] particularly for DID credentials. Specifically, in our test environment, the digital signature verification only takes an average of **0.27** seconds for accumulated 100 document(file size 1.9KB to 2.1KB) verification simulation instances.

2. Biometric identity authentication through proposed e-passport.

- * Facial image identity authentication. VGG-face recognition method [371] to conduct facial image authentication, which the image size is limited to 100 x 32 bits. It takes 9.52 seconds to complete the facial identity authentication over proposed e-passport. As facial image is captured live over mobile application camera, the majority time is spent on the facial image collection, which is estimated about 3-4 seconds minimum. The test results is achieved through author simulates 20 instances by her own mobile client.
- * Fingerprint identity authentication. Fingerprint identity authentication is only conducted at border checkpoint e-gates, which minutiae-based two dimensional feature vector matching is deployed. Since fingerprint data is *not* available in a large scale, author uses her own fingerprint data collected from U.are.u 5300 fingerprint scanner(USB 2.0, FIPS 201/PIV, FAP 30, optical, resolution: 500 dpi, 256 levels of gray) to do the test for simulating 20 instances. By initiating 'biometric identity authentication' smart contract on BBCVID Blockchain, it takes an average of 4.75 seconds.

3. Border crossing permit generation.

Starting from the border crossing applicant submits the border crossing permit application, VS BA is not able to generate the border crossing permit until all required endorsement is received. To put more in detail, there are three organizations are required to endorse a border crossing permit transaction, which include HMRC, DVLA, and the Surveillance system. Indeed, as soon as applicant submits the application, HMRC and DVLA is able to endorse the transaction straightaway; however, the surveillance system has to wait the applicant drive the declared vehicle to the border checkpoints so that the endorsement can be conducted. Therefore, the gap time of waiting the vehicle to drive-through the border checkpoint road unit will not be count into the computational cost.

Therefore, for commercial entity to obtain a border crossing permit with 20 documents to be verified altogether, the estimated computation cost is:

$$20 \times 0.27 + 9.52 + 120 + 4.02 \times 4 = 2.52 \text{ minutes} \quad (8.1)$$

which is the sum time of 20 documentation verification, facial image identity authentication over mobile application by e-passport, assumed surveillance system required time to collect live data from border

checkpoint units(120 seconds), required 3 endorsements from three different organizations, and one transaction commitment from VS BA. Particularly, the average transaction latency for 1000 bytes blind write is adopted as the transaction time.

For individual immigrant to obtain a border crossing permit with one visa documentation, the computational cost is:

$$0.27 \times 2 + 9.52 + 4.02 = 0.23 \text{ minutes} \quad (8.2)$$

Which is the sum time of visa and e-passport two documentation verification, facial identity authentication over mobile application by e-passport, and one transaction commitment from VS BA.

For individual immigrants to obtain a border crossing permit with one personal vehicle, the computational cost is:

$$0.27 \times 3 + 9.52 + 120 + 4.02 \times 3 = 2.37 \text{ minutes} \quad (8.3)$$

Which is the sum time of visa, vehicle DID documentation, e-passport three documentation verification, facial identity authentication over mobile application, assumed surveillance system required time to collect live data from border checkpoint units(120 seconds), required two endorsement from DVLA and the surveillance system, and the VS BA transaction commitment.

4. **Border crossing event recording.** The border crossing event recording refers to use the border crossing permit at the border e-gate to conduct fingerprint biometric identity authentication, and then recording it by committing this transaction on VS Blockchain. Therefore, the computational cost is:

$$0.27 + 4.75 + 1 = 6.02 \text{ seconds} \quad (8.4)$$

Which is the sum time of one e-passport verification, fingerprint identity authentication at e-gate through the e-passport, and estimated time cost to scan the QR code(1 second).

- **Infrastructure cost.** Current e-gate system reads micro-chip data and then conduct biometric identity authentication through reading the biometric identity template from the microchip. Compared with our proposal, a QR code reader is required at the e-gate and a computing Blockchain gateway service interface is also required to connect to VS Hyperledger Blockchain. Plus, weighting sensor, RFID reader, and CCTV camera are also required if it is not already installed.

- **Leveraging existing system.** Existing border crossing system has major issues such as space constraint and capped government capability. To put more in detail, the hard border checkpoint has very constraint spaces in the terms of spaces for new e-gate systems or car parking spaces for commercial lorries. Similarly, Government has limited labour input either considering the existing government organization structure and annual budget, expanding the number of government officer at a large scale seems impossible. Therefore, leveraging existing border control facility and system is crucial as well.

In our proposed BBC ABC system, the existing system is leveraged through:

1. Existing e-gate. Without too much modification, existing e-gates system can be retained at the border checkpoints for final border access control.
2. Mobile phones without extra sensors. To enable online biometric identity authentication without extra hardware, only facial identity authentication is adopted for online biometric identity authentication through the proposed e-passport, as existing mobile cameras are capable of collecting facial identity data.

- **Duration of border crossing.** In accordance with [389], existing border crossing e-gate requires 20 to 30 seconds to complete fingerprint identification and 15 to 20 second for photo facial image identity authentication. Plus, additional manual registration would be required in many occasions, which costs about 30 to 60 seconds per instance.

Therefore, a reasonable estimation of border crossing duration can be made in existing border control system. Specifically, for commercial vehicle getting across the border with 20 documents to be verified all together, the border crossing duration is 20.25 minutes($30 \times 20 + 15 + 10$ *minutes*) including the time of 20 documentation verification, facial identity authentication at e-gate, and estimated vehicle loaded products inspection time(10 minutes). For individual immigrants getting across the border, the border crossing duration is 1.25 minutes($15 + 30 \times 2$) including facial image identity authentication at border e-gate and passport and visa documents manual verification. For individual immigrants getting across the border with private vehicle, the border crossing duration duration is 5.25 minutes($15 + 30 \times 4 + 3$ *minutes*) including facial image identity authentication at border e-gate, visa, passport, driving license, and vehicle registration four document verification, and estimated vehicle inspection time at border checkpoints(3 minutes). Comparing with our proposed system which is discussed in above ‘computational cost’, the corresponding duration is 2.62 minutes, 0.33 minutes, and 2.47 minutes which including the time of obtaining the border crossing permit and obtaining the border crossing event

Duration of Border Crossing (minutes)	Individual Immigrants	Individual Immigrants with Private Vehicle	Commercial Vehicle Loaded with Products
Existing ABC	1.25	5.25	20.25
Proposed ABC	0.33	2.47	2.62
Improved Efficiency	278.79%	112.55%	672.90%

Table 8.1: A comparison between existing and the proposed border control system in the regard of border crossing duration.

immutable records, see **Table 7.1** for a comparison between existing border control system and the proposed border control system in the regard of border crossing duration.

8.3.2 Performance as Biometric Blockchain-based Identity System

[276] suggests that the Blockchain network evaluation framework consisting of several evaluation accounts such as efficiency, security, and energy and environment. Similarly, [328] also claims that below metrics can be used to evaluate Blockchain network performance. Specifically,

- Transaction latency. From a network-wide side, the amount of time that a transaction costs from its initiation to the point of its validation form is available to the whole network, which also includes the time for broadcasting. Most important, [328] also recommends to use all nodes in the system under test(SUT) to get a better evaluation of transaction latency. Specifically,

$$\text{transaction latency} = (\text{confirmation time @ network threshold}) - \text{submit time} \quad (8.5)$$

In most cases, transaction latency is represented by “...the amount of time for a percentage of the network to commit the transaction” [173].

- Transaction throughput. It is the rate of validate transaction are committed to the Hyperledger Fabric Blockchain during defined time scope. In most cases, it is represented by transactions per second. Specifically,

$$\text{transaction throughput} = \text{total committed transactions} / \text{total time in seconds} \quad (8.6)$$

- Scalability. In Chapter 5, Blockchain scalability is well-discussed. Specifically, block size, network performance, and consensus rules in general decide Blockchain scalability performance. For the proposed BBCVID system, it is built upon the Hyperledger private Blockchain. That means the proposed BBCVID has very high transaction performance scalability but low node scalability compared to Ethereum and Bitcoin public Blockchains. Currently, the BBCVID throughput is set at 3000 transactions per second, meeting the current demand for passport-based identity authentication [389]. To scale up the throughput in the proposed BBCVID, it only requires more investment in hardware facilities. However, if it were a public Blockchain, it would likely necessitate modifications to both hardware facilities and consensus rules.

To add on, Hyperledger Caliper [238], which is an established Blockchain use case based performance evaluation tool, is used to evaluate the BBCVID and VS Hyperledger Fabric Blockchain performance. Specifically, the Hyperledger Caliper parameters are:

- Hyperledger Caliper version 0.5.0;
- Hyperledger Fabric 2.4 is bound to the Hyperledger Caliper;
- There are four bare metal machines to host Caliper workers;
- Endorsement policy is 1 of Any in BBCVID but 3 of 4 in VS;
- 200 Caliper worker are used in both BBCVID and VS;
- TLS is enabled in both BBCVID and VS.

What is more, the test environment of the proposed Hyperledger Fabric Blockchain include below hardware configurations:

- Intel Core i5-8265U CPU 1.6GHz, 8GB memory.
- Intel UHD Graphics 620 with GPU 3.9GB memory.
- Ubuntu 20.04 windows subsystem Linux.
- Wifi Intel(R) Wireless-AC 9260 160MHz.

Most important, some block cutting parameters are also introduced to both BBCVID and VS. Specifically,

- Block cut time. ²: 2 seconds.

²Cut time: Cut time is the upper bound for how long a new block has to be cut even it is still not full by then.

Name	Send Rate (TPS)	Max Latency (Seconds)	Min Latency (Seconds)	Average Latency (Seconds)	Throughput TPS
Blind Write 1000 byte	2989.2	3.9	0.46	2.16	2984.8
Blind Write 100 byte	2992.3	2.8	0.34	1.58	2989.5

Table 8.2: Performance comparison of blind write 1000 and 100 byte key value on BBCVID.

- Block size. ³: 500.
- Preferred maximum bytes: 2Mb.

The proposed Android mobile application uses Google Firebase as user database and LevelDB as Hyperledger Fabric state databases, gateway service concurrency limit manually sets to 20,000 per second.

With above testing environment being declared, since Hyperledger Fabric private Blockchain does not suffer scalability issue and transaction throughput can be manually adjusted as a system parameter, only transaction latency is assessed and discussed. To put more in detail, since both BBCVID and VS Hyperledger Blockchain transactions are mostly to write blind key value in its transaction; therefore, assessed blind write per 1000 and per 100 byte performance is listed as in **Table 7.2** for BBCVID Blockchain and **Table 7.3** for VS Blockchain with a fixed TPS at 3000.

That is to say, since both BBCVID and VS Hyperledger Blockchain runs on test network with Raft consensus rule, the transaction is finalized as soon as the orderer validates the transaction, which refers to the orderer has immediate finality. Therefore, the latency performance is evaluated for each peer node in the terms of how long it takes to commit a validated transaction, which leads to a “minimum latency”, “maximum latency”, and “average latency”. Even though Hyperledger Fabric test network has pre-defined concurrency limit for 20,000 transaction per second(TPS), this assessment fixes it to 3000 to make it compatible with our hardware system. Therefore, the “send rate(TPS)” and “throughput” are both around 3000 in both cases. However, the general performance of “blind write 100 byte” outperforms the “blind write 1000 byte”.

Indeed, the performance in BBCVID and VS is closely related with the peer node size, which each node has the potential to make an impact on the entire system performance based on endorsement policy. Therefore, the average transaction latency performance is assessed based on 1, 4, 7, and 10 nodes. Specifically, see **Figure 7.20** for the average latency performance in terms of the number of the BBCVID orderer

³Block size: How many transactions per block should be ordered before the block is cut.

Name	Send Rate (TPS)	Max Latency (Seconds)	Min Latency (Seconds)	Average Latency (Seconds)	Throughput TPS
Blind Write 1000 byte	2988.7	4.21	3.75	4.02	2986.8
Blind Write 100 byte	2991.3	3.23	2.25	2.78	2990.1

Table 8.3: Performance comparison of blind write 1000 and 100 byte key value on VS.

node changes, and **Figure 7.21** is for VS Blockchain. Similarly, **Figure 7.22** for the minimum and maximum transaction latency performance in terms of 1, 4, 7, and 10 orderer nodes in BBCVID, and **Figure 7.23** is for VS Blockchain.

To put in more detail, in this test environment, the number of orderer nodes does not have any major impact on the transaction latency performance at all. The reason of that is mainly because our test network does not exist any nodes drop-off or offline at all. All nodes are always switched on whenever they are assessed; however, in reality, the increasing number of nodes will cause transaction latency significantly as it will take longer time to select a lead orderer to finalize the transaction. Most important, nodes are often offline which can be seen as fault because it could cause major transaction latency especially the orderer node number is very little initially. In both case, the network size is not a causation for transaction latency at all. However, VS Blockchain has a longer transaction latency overall because VS requires more endorsements than BBCVID Blockchain.

Last but not least, packet loss is another relevant assessment criteria to be evaluated in Blockchain network. In BBCVID, the packet loss is 0.15% by **Equation 7.7** blind write per 1000 bytes compared with 0.09% per 100 bytes. In VS, the packet loss is 0.06% blind write per 1000 bytes compared with 0.04% per 100 bytes. That is to say, blind write per 100 bytes has lower packet loss rate in both BBCVID and VS Blockchain, and VS packet loss performance is better than BBCVID overall. That is because VS has four peer nodes collecting packets from message sender but BBCVID has one node only, see **Table 7.4** for a demonstration of the packet loss rate comparison between BBCVID and VS.

$$packet\ loss = (1 - throughput\ TPS \div send\ rate\ TPS) \times 100\% \quad (8.7)$$

Packet Loss Rate	Blind Write per 1000 Bytes	Blind Write per 100 Bytes
BBCVID	0.15%	0.09%
VS	0.06%	0.04%

Table 8.4: Blind write packet loss rate comparison between BBCVID and VS.

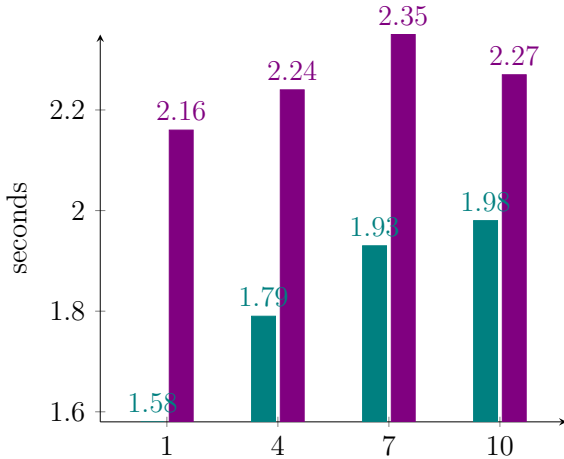


Figure 8.20: BBCVID: Average latency comparison among 1, 4, 7, and 10 orderer nodes.

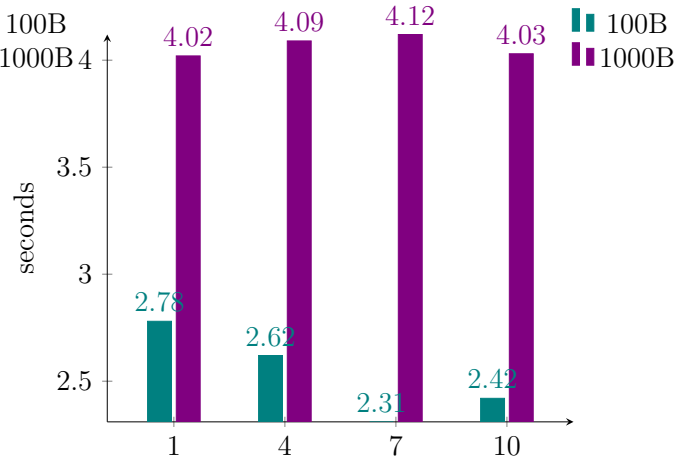


Figure 8.21: VS: Average latency comparison among 1, 4, 7, and 10 orderer nodes.

8.4 System Security Assessment

8.4.1 System Security Evaluation

In this subsection, the BBCVID and VS system security will be evaluated in terms of anti-risk and anti-impersonation risk. Specifically, the BBCVID and VS as Blockchain system, their capability of preventing Blockchain system threats will be discussed.

To put more in detail, Sybil attack[151] is one of the most famous attacks in Blockchain system, which is to create a large number of fake account or identity in the Blockchain system so that unauthorized operations can be made possible especially in a voting based consensus Blockchain by taking control over the entire Blockchain. It is a violation of the Blockchain system rule, which may damage the entire system security and data privacy. In our proposed BBC ABC system, both BBCVID and VS are Hyperledger Fabric Blockchain; therefore, they are subject to Sybil attack as well.

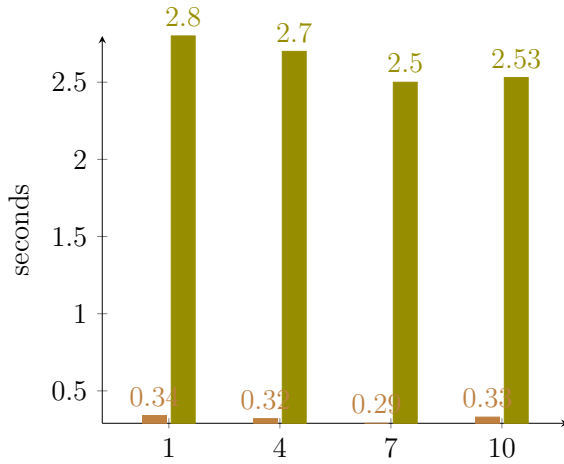


Figure 8.22: BBCVID:Min. and max.latency comparison among 1, 4, 7, and 10 orderer nodes for 100 byte blind write.

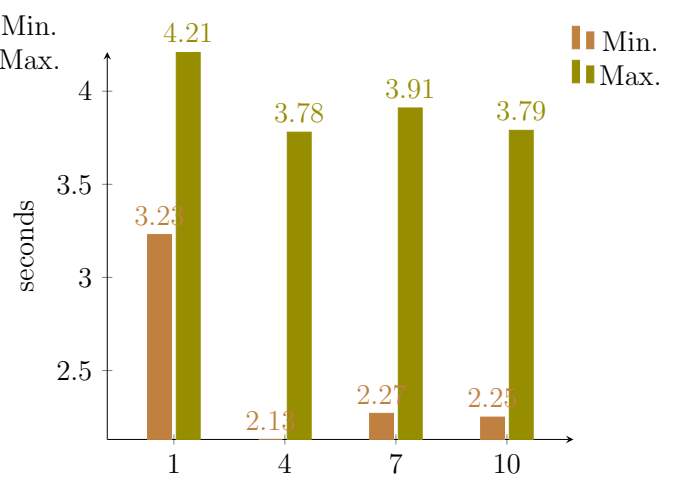


Figure 8.23: VS:Min. and max.latency comparison among 1, 4, 7, and 10 orderer nodes for 100 byte blind write.

However, in our proposal, the BBCVID system can effectively prevent Sybil attacks because of below two designs:

1. **Identity validation.** If a fake account is created, it is very difficult to validate afterwards as some information used as new user is purely faked. Therefore, identity validation can be effectively prevent Sybil attacks. In both BBCVID and VS Blockchain, new user has to obtain our Android mobile application first and then obtain the identity certificate from either Blockchain’s certificate authority. Therefore, in our proposal, there indeed is a double identity validation mechanism being introduced. To put in more detail:
 - (a) **Mobile application identity validation.** The new user identity in the Android mobile application is validated by registered emails through the Google IdM system, and a validation link has to be activated. Additionally, mobile fingerprint authentication is used to secure access to the mobile application. In accordance with [471], [510], an attacker can steal the Google IdM system identity once every four to six hours, and the state of the art in mobile fingerprint recognition achieves a 2.0% error acceptance rate. This suggests a possibility of the mobile application identity being compromised. However, the user’s identity will never be authenticated solely by the mobile application identity. The mobile identity only provides access to the BBCVID Blockchain network and does not have any access to

its stored e-passport credentials. Therefore, even if the mobile application identity is stolen, the e-passport remains secure.

- (b) **Certificate authority public key identity validation.** BBCVID and VS central authority connects to mobile application user database and uses its LDAP as registry. Fabric CA will automatically generate the public key identity certificate without any further identity validation at this point, but will request new user's biometric DID for all transactions that the user could of initiating. Because biometric identity is globally unique, the public key certificate indeed is permanently linked with new user's biometric identity, which is deemed the second identity validation in our proposal.

2. **Economic cost.** Since new user's public key certificate is immutably bind with biometric identity, faking a biometric identity will introduce a lot difficulty in terms of economic cost and technical barriers. It has been discussed in chapter 3 subsection of biometric identity that the impersonation cost for faking biometric identity is very expensive, which not only the material but also the techniques required to conduct a biometric identity impersonation. Therefore, our BBCVID and VS Blockchain can effectively prevent Sybil attack.

For other anti-risk capability evaluation, the proposed BBC ABC system is subject to forking risk. That is to say, for private Hyperledger Fabric Blockchain, since all nodes are verified participants, crash faults is more important than any other system faults, which specifically only 50 % nodes of crash fault can be tolerant, and a system forking will happen when there is more than 50 % of nodes are out of current leader's control. To put more in detail, packet loss is the main reason of network split. If the network split occurs, raft consensus algorithm would restart a new leader election process. In the meantime, the Blockchain will stop accepting new transactions. that means the Blockchain network become unavailable, which degraded the efficiency of the network [233]. Therefore, packet loss rate are meant to be monitored within the system all the time so that forking can be effectively prevented. Current packet loss rate in both BBCVID and VS are below 0.15% which is deemed in a very safe zone.

Another major risk concern is data privacy and data breaching in the proposed BBC ABC system. Specifically, we decentralize that risk from traditional centralized system into distributed system and well protect user's data by Hyperledger Fabric private data protection implementations. To put more in detail, in case of data breaching, user's data is stored into *distributed* BBCVID orderer nodes database. Therefore, the entire system data breaching risk is dramatically decentralized into a number of distributed individual nodes. To protect data privacy, all private data are well encrypted and then transmitted online. To in case of overhearing, the proposed data exchange protocol is built upon IEEE 2410-2019 BOPS protocol and

TLS is employed, which is deemed secure and effective. For nodes to valid the transaction, only the hash of the private data is recorded on in the BBC ABC system. Most important, channels and smart contract definitions are all used to secure the transaction private data. These implementations are recommended by Hyperledger Fabric and is proved to be effective and secure.

8.4.2 Anti-Impersonation Capability Discussion

As a ABC proposal, its capability of anti-impersonation will be discussed in this subsection in terms of different scenarios assumptions.

To start with, impersonation refers to someone pretend to be someone else who they actually are not. The aim of doing that in our system could be obtaining border crossing permit, considering the identity authentication and border crossing legitimate documentation verification are all conducted online. However, one should find it is nearly impossible to make an impersonation in our system for below reasons:

- **Every verifiable documentation is bind with the documentation subject's biometric DID.** Either as a commercial business entity or as an individual, all documentation issued in BBCVID can be authenticated by document owner's biometric identity(facial image or fingerprint). Plus, since the biometric identity is validated and enroled by BBCVID CA, the BBCIVD CA is able to tell all impersonation instances as in below scenarios:
 - **Instance One: Individual impersonation.** Individual impersonation refers to person A claims to be another person B through biometric identity switching, such as allow person B to attend BBCVID CA referred biometric identity collection point and enroled in the mobile application. Person A claims to be person B is because A does not have a border crossing legitimacy. Supposing person A is able to make an border crossing permit application over the mobile by person B's biometric identity. Even all that is successful, when person A arrives at the border e-gate, person A's fingerprint will be collected at the e-gate to match with BBCVID stored person B's fingerprint identity template. This is an obviously mismatch and therefore person A is not able to get through the e-gate is guaranteed. Therefore, there is no point to make an impersonation in this scenario as person A is not able to get across the border.
 - **Instance Two: Document 'impersonation'.** Documentation 'impersonation' refers to the commercial business entity uses another commercial business customs clearance documentation to apply border crossing permit. The reason why the business entity to impersonate documentation is because they do not have legitimacy to get across the border, which is

smuggling. The commercial will be loaded with products that is not consistent with the application at all. Therefore, in this scenario, it depends on the border checkpoints RFID reader read vehicle customs RFID e-seal number to do the inspections to figure this smuggling out, or the on road sensor unit to capture the weight of the vehicle and distinguish the data mismatch. Even though smuggling is possible under the condition that the documentation the business entity impersonated compiles border crossing legitimacy. However, our proposal still has two counter-measurements being introduced.

- **E-gate fingerprint access control.** E-gate system as the final step to get across the border takes the responsibility to ensure the person who pass the e-gate is only the person who has obtained the border crossing permit from BBCVID. However, there are still two scenarios where a person can attempt to get across the e-gate without the border crossing permit at all. Specifically,
 1. **Case One:** Person A takes person B's digital border crossing permit. Person A takes person B's mobile client with a valid border crossing permit issued by BBCVID CA. The person A's fingerprint will be collected at the e-gate, and the fingerprint identity template is supplied by the BBCVID CA which belong to person B. It will definitely lead to an identity mismatch; therefore, person A is not possible to get across the e-gate with person B's border crossing permit.
 2. **Case Two:** Driver X drives another company's commercial vehicle to pass through the e-gate. That is to say, the driver drives a commercial vehicle with undeclared products to get across the e-gate with company Y's valid border crossing permit. In this case, since Y's valid border crossing permit contains the vehicle number plate and vehicle colour information, if driver X drives a different car and tries to get across the border, driver X will be stopped as soon as the road unit CCTV captures his vehicle number plate and vehicle colour. Further assuming the driver driver Y's vehicle and tries to pass through the e-gate, the road surveillance unit is able to capture its vehicle weight and RFID reader for further detection. Suppose the driver even get it away from there, driver has to make fingerprint identity authentication at the e-gate with VS referred fingerprint identity template that belongs to Y's driver but not X. Therefore, driver X is definitely not able to get across the border e-gates with company Y's valid border crossing permit.

For a short summary, in all above assumed scenarios, the person who tries to get across the border with someone else valid border crossing permit is completely not

feasible.

8.5 Chapter Summary

In this chapter, the proposed system is implemented on Hyperledger Fabric test network and its performance is evaluated. In **Chapter 7.1 System Implementation**, some system prerequisites are introduced before implementing the Hyperledger Fabric Blockchain, and then an introduction about naive Hyperledger Fabric test network is offered. In accordance with our proposal introduced in chapter 6, properties of the implemented BBCVID and VS Blockchain, transaction, consensus rule, and transaction privacy preserving methods and protecting system security manoeuvres are all explained. To make the proposal functional, the border crossing individual needs an user interface. Therefore, an android mobile application(DApp) is built up. In **Chapter 7.2 System Simulation**, author uses own personal data to simulate a full walk through of the border crossing event. From obtaining access to mobile application and the Blockchain network, to get the border crossing event being recorded on the VS Hyperledger Blockchain. The representation of the proposed e-passport in moible application and the representation of the VS border crossing event records are all demonstrated, so does the corresponding Blockchain backend data. In **Chapter 7.3 System Performance Evaluation**, the performance of the system as an automatic border crossing system and as Blockchain IdM system are evaluated separately. Hyperledger Caliper tool is used to assess BBCVID and VS Blockchain network performance. Finally in **Chapter 7.4 System Security Assessment**, the system anti-Sybil attack capability, anti-forking capability, and user data privacy preserving and anti-data-breaching capability are all discussed. Most important, some impersonation cases are well discussed based on case assumptions.

Chapter 9

Conclusions and Future Work

9.1 Summary of Work

Apart from chapter one that is a short overview of this thesis, the real work of this project starts from related work review at chapter 2. To construct a sound digital twin identity management system on Blockchain, biometric IdM system is reviewed first, and then another three IdM systems that have the most potential to replace existing centralized IdM system are also reviewed. It aims to conquer the criticism of existing IdM system and know of the state of art of existing biometric IdM systems, which specifically include fast identity online(FIDO), public key infrastructure(PKI), and self-sovereign identity(SSi). Through this part of literature review, there are two major findings:

1. In current literature, *digital* identity authentication is always based on risk assurance. That is to say, digital identity authentication always bares a risk of impersonation; Therefore, multi-factors authentication is encouraged when requested assurance is high.
2. Securer and privacy more well preserved IdM system normally has more costs. That is, they are either computationally more expensive or request extra hardware. Therefore, an IdM system is always a compromised decision among cost, security, and privacy.

After that, biometric SSI IdM on Blockhain is reviewed. Specially, biometric identity template storage, raw biometric identity collection, and authentication mechanism in Blockchain network are all well reviewed. It is learnt that biometric identity template off-chain storage in permissioned Blockchain is a good balance between cost and security, and efficiency. Plus, collecting encrypted biometric data is an efficient choice if otherwise an extra hardware is required considering preserving biometric data privacy.

In the last subsection of chapter 2, the existing e-passport and smart border control system are reviewed. To put more in detail, the existing e-passport is a combination of PKI and RFID techniques, which requires a passport book to carry a microchip to embed all identity information. It clearly suffers all the inconvenience embedded with distance, which is a strong reason of why existing border control efficiency is low. What is more, in reviewing existing smart customs control solution, border crossing documentation verification is deemed as the most time consuming part in the entire border crossing process as existing verification method is still manual. Even though there does exist very limited smart documentation system, it only deals with smart documentation application but not verification at all.

Moving onto chapter 3, it still aims for researching how to construct a sound digital twin identity management system by understanding what identity is in the Metaverse. Admittedly, an identity in the Metaverse has minor difference with general digital identity, which Metaverse normally has a much wider range of audience and is used in the context of Blockchain applications. To create a digital twin of real world existing entity, a digital identifier is indispensable. Therefore, uniform resource identifier(URI) and its general syntax is studied. Particularly, decentralized identifier(DID) is also studied. To put more in detail, DID is generated particularly through decentralized Blockchain. W3C defines a DID syntax rule for all DID so that interoperability can be improved. At the final section of chapter 3, digital verifiable credential standard data model is researched, which the model normally include credential metadata, claims, and proofs three main components.

To create an IdM system, what an IdM system is has to be addressed. In chapter 4 identity management system, it is learnt that IdM has participants, participant's role, models, and principles. Plus, digital identity authentication mechanism and particularly biometric identity authentication mechanism are both well researched in this section as well. In the middle section of this chapter, established IdM protocols like HTTP authentication and single-sign-on are studied. Apart from adopting W3C DID defined data model standard to improve interoperability, a JSON data object identity token can be generally accepted by internet communications. Then IdM risk management frameworks and IdM system security and preserving data privacy are researched. Through this chapter, the framework and the best conduct of building up an IdM is well researched.

In chapter 5 Blockchain, a deep thorough research work is conducted about nearly every concern of a Blockchain system. Particularly, Blockchain DAO, smart contracts, and an comparison among Blockchain 1.0(Bitcoin Blockchain), 2.0(Ethereum), and 3.0(Hyperledger Fabric) are well researched. Hyperledger Blockchain as the third generation of Blockchain network, which has the most flexibility and extensibility; Therefore, Hyperledger is the most popular Blockchain in commercial context, especially when transaction is complex. Besides that, Blockchain major issue and

threats are also studied. This chapter is deemed as a premise and necessary work for constructing a well-built automatic Blockchain DAO.

To improve the existing border control efficiency, turning current manual work into automatic is an optimal solution. From chapter 2, the most time-consuming procedure in the entire border control process is identified on the documentation verification, such as visa and customs clearance documentation etc. Plus, border control procedure requires data reconciliation between what is being claimed in the documentation and what it is in reality. Therefore, border checkpoints surveillance system is indispensable.

In chapter 6, the full proposal of the privacy-aware biometric Blockchain based e-passport system for automatic border control is offered. Starting from the system overview, the BBCVID Hyperledger Fabric Blockchain for e-passport and VS for automatic border control are all well explained separately. Specifically, BBCVID is designed particularly for generating the digital twin of the real world e-passport book so that user's identity can be authenticated in the border control Metaverse DAO through the proposed e-passport. By similar token, the user's digital identity is generated also through BBCVID as an immutable DID. Most importantly, the algorithm of generating the proposed e-passport can be generalized into a digital verifiable documentation data model to generate real world visa and customs clearance documentations' digital twin in the border control Metaverse DAO. The aim of BBCVID is to make border crossing documentation autonomous verification possible. For VS Blockchain, it is the border control Metaverse DAO, which is used mainly for recording every border crossing event on the VS Blockchain and generating the border crossing permit for all border crossing entities. VS is constructed as a decentralized autonomous organization, which aims to offload border crossing manual labour input as much as possible so that efficiency can be improved as much as possible too.

In chapter 7, the proposed system implementation and system performance evaluation is made. The implementation is significantly depends on an Android mobile application to enable information transmission between border crossing applicant and the proposed automatic border control DAO. What is more, the BBCVID and VS Blockchain configuration and test network environment is well explained before given the risk assessment and system performance evaluation results.

9.2 Research Questions Revisited

RQ 1: How to create a digital twin for individual bodily person in the Metaverse?

To answer that question, related works, identity and identifier in the Metaverse, and the identity management system are well researched through chapter 2 to 4. Through our research, real world individual identity's digital twin into the

Metaverse requires interoperability and veracity, which the benefits of it does not only limited to immersive user experience in the Metaverse but also for the sake of Metaverse community security. Plus, identity digital twin in the Metaverse becomes decentralized identity through Blockchain operations, which is able to escape from the criticism of centralized IdM systems.

To add on, it is found that the tie between the real world individual bodily person and the digital twin identity is crucial. That is to say, if biometric identity intrinsic identity is deployed, not only the present of a bodily person over the internet can be verified, but also the digital twin identity itself is well solved. That is to say, biometric identity cannot only be used in the real world as an identity credential to authenticate a bodily person's identity, but also can be used in the digital twin as an identity credential to authenticate the digital identity. Then, how to fix a tie between the real world identity and the digital twin identity becomes crucial considering the identity credential is sorted in both reality and the Metaverse. Through our research work, our answer to that is the Blockchain immutable ledger. We aims to fix that tie through Blockchain, so that the tie is immutable and undeniable. To improve the interoperability of the real world identity digital twin, W3C digital verifiable identity credential data model is adopted.

RQ 2: How to effectively improve the border crossing efficiency overall?

The main issue of the existing border crossing system is identified as it costing too much time in manual border crossing legitimacy documentation verification, such as visa and customs clearance documentations etc. To put more in detail, current e-gates system is only available to a small group of people who do not require visa verification nor customs declarations. The individual immigrant who requires a visa has to visit border control officer checkpoints to authenticate the visa document and the immigrant's identity, which is deemed the causation of border control low efficiency. Therefore, this proposal suggests to use Fog computing and Blockchain Metaverse DAO to offload and accelerate the border crossing workloads as much as possible so that the efficiency can be improved as much as possible.

Specifically, There are three main changes are made to improve the existing border crossing workflow efficiency:

1. Digitize border crossing documentation. BBCVID Blockchain is constructed to generate the digital twin of visa and customs clearance documents so that documentation verification can be conducted by computing techniques instead of border officers. That is to say, through the application of BBCVID, the e-gate system is made available to all general public without any limitations so that border checkpoints manual labour can be significantly reduced. To put more in detail, the generated digital verifiable border crossing documentation

is secured by both issuer's digital signature and owner's biometric identity. That is to say, not only the issuer's digital signature can guarantee the digital verifiable documentation's integrity and authenticity but also the documentation's ownership can be impossibly modified.

2. Automatize identity authentication and documentation verification. Through the construction of border control Metaverse DAO on VS Blockchain, the border crossing permit and events are both generated and recorded on VS Blockchain through the execution of smart contract. That is to say, the identity authentication, border crossing documentation verification, and border crossing events recording are all automatized through Blockchain smart contracts.
3. Fog computing. Since border control task is deemed computational intense and latency critical, Fog computing is deployed whenever is possible so that the entire workflow can be offloaded from the border checkpoints to user's end device, surveillance system IoT, and Edge devices around the border checkpoint road units. It aims to decentralized entire border control workloads in terms of computational cost.

RQ 3: How secure and robust is the proposed system in terms of preserving identity privacy and protecting system security?

The proposed BBC ABC system is a Metaverse DAO combining with identity digital twin projected to it. To put more in detail, the constructed border control Metaverse DAO is decentralized autonomous organization on Blockchain, which is to use smart contract on top of Blockchain network to autonomously fulfil border control organizational purpose. Similarly, the identity digital twin projected to the border control Metaverse DAO is also enable through Blockchain(BBCVID), which is biometric-aware, privacy-preserving, and a veracity reflection of real world legal identity. Therefore, the proposed system is assessed separately between "Automatic Border Crossing(ABC) System" and "Blockchain Biometric Identity System".

Specifically, in accordance with reviewed related works, the proposed Automatic Border Crossing (ABC) system performance and security has been evaluated from several aspects, which include cost, border crossing duration, and leveraging existing system. As for assessing and evaluating the proposed BBC ABC system as Blockchain-based system, the Blockchain configuration and test network environment are introduced before simulation results are given. Blockchain transaction latency performance and packet loss rate are both well assessed. Specifically, the improved efficiency for our proposed BBC ABC border control system can be as high as 672.90% compared with the existing system. Plus, the packet loss rate in both BBCVID and

VS are well below 2% benchmark, which only has 0.15% and 0.06% highest packet loss rate respectively.

Last but not least, in the regard of protecting system security, the proposed BBC ABC system is capable of effectively preventing Sybil attack and forking through identity validation and pocket loss rate monitor. Plus, data privacy is well preserved by IEEE 2410-2019 biometric data exchange protocol and TLS protocol. Hyperledger transaction data privacy protection mechanism such as smart contract definition, channels, and privacy policy definition are all employed to add more privacy to the data. Most important, the BBC ABC anti-impersonation capability is well explained based on a scenario assumptions, which turns out that impersonation is not possible in the proposed BBC ABC system.

9.3 Contributions

In this project, sorting existing border control low efficiency issue is targeted as the main goal. Through our research, the causation of that can be summarized into two folds, which specifically are manual documentation verification and the static identity credential. To be more precise, proving border crossing legitimacy involves a good amount of documentation verification, such as identity credential passport, individual immigrant border crossing legitimacy proof visa, and commercial products border crossing legitimacy proof customs clearance documents etc. Apart from existing e-passport, all the rest of documentation verification has to be manually conducted by border control officer, which is deemed very time-consuming. Most important, current e-passport contains static information that is locked in the e-passport book microchip, which does not only suffer all inconvenience embedded with distance but also makes binding dynamic border crossing legitimacy to the passport extremely difficult.

To transform existing border control manual documentation verification into automatic process, the tangible border crossing legitimacy documentation has to be digitized. Therefore, the first contribution of this project is a real world verifiable documentation digital twin data model is constructed through Blockchain technology. Through our proposal, not only existing e-passport but also visa, customs clearance documentation, individual bodily person, and vehicle digital twin can be constructed and immersive projected into Metaverse. Most important, the constructed digital twin enables embed dynamic information to the real world entity, which makes computing machine learning and self-execution techniques applicable to it.

To further automatizing the border control process, a border control Metaverse DAO is constructed on top of Hyperledger Fabric Blockchain, which is deemed as the second contribution of this project. To be more precise, through the construction of BBC ABC Metaverse DAO system, the border crossing efficiency is improved by 354.75% on average, which specifically has 112.55% minimum improved efficiency

under individual immigrants with private vehicle border crossing scenario and 672.90% highest improved efficiency under commercial vehicle loaded with commercial products border crossing scenario.

9.4 Future Work

Real Scenario Assessment

In our project, the proposed system assessment is conducted through computing simulation and Hyperledger performance evaluation toolkit Hyperledger Caliper. Therefore, in the future work, real scenario assessment is highly recommended and encouraged. Our proposal covers all the implementation details, Android mobile application user interface, and assumed scenario border crossing case discussions; however, if real scenario case and data can be collect, it will definitely offer a better performance evaluation of the system is for sure.

Constructing BBC ABC Metaverse DAO Interface

Border crossing applicant user interface is already developed in this project, which is a decentralized application on Android mobile. It does not only transfer data between user and the proposed BBC ABC system but also make computing data presentable and readable. In our project, only the border crossing applicant user interface is developed but not any interface for the BBC ABC Metaverse DAO. All communications and data outputs are generated and achieved through Ubuntu Windows Subsystem Linux shell scripting. Therefore, in future work or in project production, it is highly recommended to construct a BBC ABC Metaverse DAO interface for scalable applications, see **Figure 8.1** for a demonstration of an initial idea of the DAO interface.

Smarter BBC ABC

Our proposed BBC ABC achieves the goal of improving existing border control efficiency at a remarkable level. We do not only automatize existing ABC system as much as possible but also construct a dynamic digital twin of real world entity that being projected to the proposed BBC ABC. It makes computing technique such as machine learning and big data analysis applicable to real world entity. Therefore, smarter BBC ABC can be another research aspect of this project in the future, such as border crossing reputation or credit systems and smart smuggling behaviour detection etc.

Biometric BlockChain Virtual Identity Generator

Personal Information	Commercial Entity
Given Name : _____	Tax and Customs Declaration No. : _____
Surname : _____	Custom Claim No. : _____
Date of Birth : Month _____ Date _____ Year _____	Company Name: _____
Passport No. : _____	Company No.: _____
Vehicle Plate No. : _____	Product List No.: _____
Vehicle Colour : _____	Claimed Product Total Weight : _____ kg
<input type="button" value="Click to Collect Photography"/>	Shipping Vehicle Colour : _____
<input type="button" value="Click to Collect Fingerprint"/>	Shipping Vehicle Plate No. : _____
<input type="button" value="Personal Virtual ID Generator"/>	<input type="button" value="Commercial Virtual ID Generator"/> <input type="button" value="Save"/>






Figure 9.1: A demonstration of an initial idea for the BBC ABC Metaverse DAO interface.

References

- [1] W. Abdul, Z. Ali, S. Ghouzali, B. Alfawaz, G. Muhammad, and M. S. Hossain, "Biometric security through visual encryption for fog edge computing," *IEEE Access*, vol. 5, pp. 5531–5538, 2017, ISSN: 2169-3536. DOI: 10.1109/access.2017.2693438.
- [2] D. H. Ackley, G. E. Hinton, and T. J. Sejnowski, "A learning algorithm for boltzmann machines," *Cognitive science*, vol. 9, no. 1, pp. 147–169, 1985.
- [3] H. Afzaal and N. A. Zafar, "Modeling of iot-based border protection system," in *2017 First International Conference on Latest trends in Electrical Engineering and Computing Technologies (INTELLECT)*, 2017, pp. 1–6. DOI: 10.1109/INTELLECT.2017.8277639.
- [4] W. A. Ahmed and A. Rios, "Digitalization of the international shipping and maritime logistics industry: A case study of tradelens," in *The Digital Supply Chain*, Elsevier, 2022, pp. 309–323.
- [5] Y. Ai, M. Peng, and K. Zhang, "Edge computing technologies for internet of things: A primer," *Digital Communications and Networks*, vol. 4, no. 2, pp. 77–86, 2018, ISSN: 23528648. DOI: 10.1016/j.dcan.2017.07.001.
- [6] Z. Akhtar, A. Hadid, M. S. Nixon, M. Tistarelli, J.-L. Dugelay, and S. Marcel, "Biometrics: In search of identity and security (q & a)," *IEEE MultiMedia*, vol. 25, no. 3, pp. 22–35, 2018.
- [7] Z. Akhtar, A. Hadid, M. S. Nixon, M. Tistarelli, J.-L. Dugelay, and S. Marcel, "Biometrics in search of identity and security," *IEEE MultiMedia: BIOMETRIC RECOGNITION AND SECURITY*, vol. JULY/SEPTEMBER 2018, pp. 22–35, 2018, ISSN: 1070-986X/18/\$33.00 2018 IEEE.
- [8] S. Albers *et al.*, *The design of alliance governance systems*. Springer, 2005.
- [9] C. Alexander, *A pattern language: towns, buildings, construction*. Oxford university press, 1977.
- [10] M. T. Alexander, "Time sharing supervisor programs," *Univ. of Michigan Computing Center, Ann Arbor, MI*, 1970.

- [11] I. Ali, M. Gervais, E. Ahene, and F. Li, “A blockchain-based certificateless public key signature scheme for vehicle-to-infrastructure communication in vanets,” *Journal of Systems Architecture*, vol. 99, 2019, ISSN: 13837621. DOI: 10.1016/j.sysarc.2019.101636.
- [12] O. Ali, M. Ally, Y. Dwivedi, *et al.*, “The state of play of blockchain technology in the financial services sector: A systematic literature review,” *International Journal of Information Management*, vol. 54, p. 102 199, 2020.
- [13] O. Ali, A. Jaradat, A. Kulakli, and A. Abuhlimeh, “A comparative study: Blockchain technology utilization benefits, challenges and functionalities,” *IEEE Access*, vol. 9, pp. 12 730–12 749, 2021.
- [14] C. Allen, A. Brock, V. Buterin, *et al.*, *Decentralized public key infrastructure. a white paper from rebooting the web of trust*, 2015.
- [15] F. Alliance, “What is fido?” *fidoalliance.org*, 2023. [Online]. Available: <https://fidoalliance.org/what-is-fido/>.
- [16] A. Allison, J. Currall, M. Moss, and S. Stuart, “Digital identity matters,” *Journal of the American Society for Information Science and Technology*, vol. 56, no. 4, pp. 364–372, 2005.
- [17] J. S. Allwood, N. Fierer, and R. R. Dunn, “The future of environmental dna in forensic science,” *Applied and environmental microbiology*, vol. 86, no. 2, e01504–19, 2020.
- [18] G. Alpár, J.-H. Hoepman, and J. Siljee, “The identity crisis. security, privacy and usability issues in identity management,” *arXiv preprint arXiv:1101.0427*, 2011.
- [19] J. Alsayed Kassem, S. Sayeed, H. Marco-Gisbert, Z. Pervez, and K. Dahal, “Dns-idm: A blockchain identity management system to secure personal data sharing in a network,” *Applied Sciences*, vol. 9, no. 15, 2019, ISSN: 2076-3417. DOI: 10.3390/app9152953.
- [20] A. A. Alshdadi, R. Mehboob, H. Dawood, M. O. Alassafi, R. Alghamdi, and H. Dawood, “Exploiting level 1 and level 3 features of fingerprints for liveness detection,” *Biomedical Signal Processing and Control*, vol. 61, 2020, ISSN: 17468094. DOI: 10.1016/j.bspc.2020.102039.
- [21] M. A. Alsmirat, F. Al-Alem, M. Al-Ayyoub, Y. Jararweh, and B. Gupta, “Impact of digital fingerprint image quality on the fingerprint recognition accuracy,” *Multimedia Tools and Applications*, vol. 78, no. 3, pp. 3649–3688, 2018, ISSN: 1380-7501 1573-7721. DOI: 10.1007/s11042-017-5537-5.

- [22] A. Alsunbul, W. Elmedany, and H. Al-Ammal, "Blockchain application in healthcare industry: Attacks and countermeasures," in *2021 International Conference on Data Analytics for Business and Industry (ICDABI)*, IEEE, 2021, pp. 621–629.
- [23] N. N. G. de Andrade, S. Monteleone, and A. Martin, "Electronic identity in europe: Legal challenges and future perspectives (e-id 2020)," *JRC Scientific and Police Reports (Europe Commission)*, 2013.
- [24] S. Angert, "Blockchain technology implementation in the us customs environment," Naval Postgraduate School Monterey United States, Tech. Rep., 2019.
- [25] Y. C. Anil Jain and M. Demirkus, "Pores and ridges fingerprint matching using level 3 features," *The 18th International Conference on Pattern Recognition (ICPR'06)*, 2016.
- [26] e. a. Anthony Lusard, "An overview of hyperledger foundation," *Hyperledger.org*, 2021. [Online]. Available: https://www.hyperledger.org/wp-content/uploads/2021/11/HL_Paper_HyperledgerOverview_102721.pdf.
- [27] A. M. Antonopoulos, *Mastering Bitcoin: unlocking digital cryptocurrencies.* "O'Reilly Media, Inc.", 2014.
- [28] F. A. Aponte-Novoa, A. L. S. Orozco, R. Villanueva-Polanco, and P. Wightman, "The 51% attack on blockchains: A mining behavior study," *IEEE Access*, vol. 9, pp. 140 549–140 564, 2021.
- [29] 3. R. K.-M. APRI SISWANTO 2NORLIZA KATUK, "Fingerprint template protection schemes a literature review," *Journal of Theoretical and Applied Information Technology*, vol. 31st May 2018. Vol.96. No 10, pp. 2764–2781, 2018, ISSN: 1992-8645.
- [30] O. A. Arigbabu, S. M. S. Ahmad, W. A. W. Adnan, and S. Yussof, "Integration of multiple soft biometrics for human identification," *Pattern Recognition Letters*, vol. 68, pp. 278–287, 2015.
- [31] M. Armbrust, A. Fox, R. Griffith, *et al.*, "A view of cloud computing," *Communications of the ACM*, vol. 53, no. 4, pp. 50–58, 2010.
- [32] M. Armbrust, A. Fox, R. Griffith, *et al.*, "A view of cloud computing," *Communications of the ACM*, vol. 53, no. 4, pp. 50–58, 2010, ISSN: 0001-0782 1557-7317. DOI: 10.1145/1721654.1721672.
- [33] L. A. Arpaia and Pasquale, "Experimental test of ecdsa digital signature robustness from timing-cattice attack," *IEEE Instrumentation and Measurement Society*, 2020.
- [34] K. J. Arrow, A. Sen, and K. Suzumura, *Handbook of social choice and welfare.* Elsevier, 2010, vol. 2.

- [35] R. A. A. H. Arshad Jama, “Blockchain based identity verification system,” 2019.
- [36] N. Asokan, J.-E. Ekberg, K. Kostiainen, *et al.*, “Mobile trusted computing,” *Proceedings of the IEEE*, vol. 102, no. 8, pp. 1189–1206, 2014.
- [37] M. H. Au, J. K. Liu, W. Susilo, and T. H. Yuen, “Constant-size id-based linkable and revocable-iff-linked ring signature,” in *Progress in Cryptology-INDOCRYPT 2006: 7th International Conference on Cryptology in India, Kolkata, India, December 11-13, 2006. Proceedings 7*, Springer, 2006, pp. 364–378.
- [38] P.-L. Aublin, S. B. Mokhtar, and V. Quéma, “Rbft: Redundant byzantine fault tolerance,” in *2013 IEEE 33rd International Conference on Distributed Computing Systems*, IEEE, 2013, pp. 297–306.
- [39] Auth0.com, “Single sign-on,” 2023. [Online]. Available: <https://auth0.com/docs/authenticate/single-sign-on>.
- [40] M. Azman and K. Sharma, “Smart boarding system with e-passports for secure and independent interoperability,” *SN Computer Science*, vol. 3, pp. 1–9, 2022.
- [41] **B. Xu**, Q. Ni, R. Jiang, A. Bouridane, and C.-T. Li, “Biometric blockchain (bbc) based e-passports for smart border control,” in *Big Data Privacy and Security in Smart Cities (Advanced Sciences and Technologies for Security Applications)*, Advanced Sciences and Technologies for Security Applications. 2022, ch. Chapter 13, pp. 235–248, ISBN: 978-3-031-04423-6 978-3-031-04424-3. DOI: 10.1007/978-3-031-04424-3_13.
- [42] **B. Xu**, T. Agbele, and R. Jiang, “Biometric blockchain a better solution for the security and trust of food logistics,” *IOP Conference Series: Materials Science and Engineering*, vol. 646, 2019. DOI: 10.1088/1757-899X/646/1/012009.
- [43] **B. Xu**, T. Agbele, Q. Ni, and R. Jiang, “Biometric blockchain a secure solution for intelligent vehicle data sharing,” *Springer Cham.*, 2020. DOI: 10.1007/978-3-030-32583-1_11.
- [44] **B. Xu**; R. Jiang; and Q. Ni, “Privacy-aware biometric blockchain based e-passport system for automatic border control,” *IEEE Transactions on Information Forensics and Security*, 2023 *Under Review*.
- [45] V. Balakrishnan, “Unconstrained biometric recognition summary of recent socia lab research,” *TECHNICAL REPORT*,
- [46] D. Bao and L. You, “Two-factor identity authentication scheme based on blockchain and fuzzy extractor,” *Soft Computing*, pp. 1–13, 2021.

- [47] S. Barman, H. P. H. Shum, S. Chattopadhyay, and D. Samanta, “A secure authentication protocol for multi-server-based e-healthcare using a fuzzy commitment scheme,” *IEEE Access*, vol. 7, pp. 12 557–12 574, 2019, ISSN: 2169-3536. DOI: 10.1109/access.2019.2893185.
- [48] M. Barni, G. Droandi, R. Lazzeretti, and T. Pignata, “Semba: Secure multi-biometric authentication,” *IET Biometrics*, vol. 8, no. 6, pp. 411–421, 2019, ISSN: 2047-4938 2047-4946. DOI: 10.1049/iet-bmt.2018.5138.
- [49] M. Al-Bassam, “Scpki: A smart contract-based pki and identity system,” in *Proceedings of the ACM Workshop on Blockchain, Cryptocurrencies and Contracts*, 2017, pp. 35–40.
- [50] F. Battistone and A. Petrosino, “Tglstm: A time based graph deep learning approach to gait recognition,” *Pattern Recognition Letters*, vol. 126, pp. 132–138, 2019.
- [51] R. Beck, C. Müller-Bloch, and J. L. King, “Governance in the blockchain economy: A framework and research agenda,” *Journal of the Association for Information Systems*, vol. 19, no. 10, p. 1, 2018.
- [52] C. Bellavitis, C. Fisch, and P. P. Momtaz, “The rise of decentralized autonomous organizations (daos): A first empirical glimpse,” *Venture Capital*, pp. 1–17, 2022.
- [53] J. Benet, *Ipfs-content addressed, versioned, p2p file system. arxiv 1407.3561*, 2020.
- [54] A. Beniiche, “A study of blockchain oracles,” *arXiv preprint arXiv:2004.07140*, 2020.
- [55] J. Bernal Bernabe, J. L. Canovas, J. L. Hernandez-Ramos, R. Torres Moreno, and A. Skarmeta, “Privacy-preserving solutions for blockchain: Review and challenges,” *IEEE Access*, vol. 7, pp. 164 908–164 940, 2019, ISSN: 2169-3536. DOI: 10.1109/access.2019.2950872.
- [56] M. Berners-Lee, *Mccahill; request for comments: 1738; uniform resource locators (url); retrieved from internet on nov. 1, 2009*.
- [57] I. Beschastnikh, P. Liu, A. Xing, P. Wang, Y. Brun, and M. D. Ernst, “Visualizing distributed system executions,” *ACM Transactions on Software Engineering and Methodology*, vol. 29, no. 2, pp. 1–38, 2020, ISSN: 1049-331X 1557-7392. DOI: 10.1145/3375633.
- [58] Binance.com, “Proof of authority explained,” *Academy Binance*, 2018. [Online]. Available: <https://academy.binance.com/en/articles/proof-of-authority-explained>.

- [59] E. Birrell and F. B. Schneider, “Federated identity management systems: A privacy-based characterization,” *IEEE security & privacy*, vol. 11, no. 5, pp. 36–48, 2013.
- [60] C. Bisogni, G. Iovane, R. E. Landi, and M. Nappi, “Ecb2: A novel encryption scheme using face biometrics for signing blockchain transactions,” *Journal of Information Security and Applications*, vol. 59, 2021, ISSN: 22142126. DOI: 10.1016/j.jisa.2021.102814.
- [61] G. Bissias and B. N. Levine, “Bobtail: Improved blockchain security with low-variance mining,” in *NDSS*, 2020.
- [62] Bitcoin.org, “Bitcoin developer guide: Wallets,” *Bitcoin.org*, 2020. [Online]. Available: <https://developer.bitcoin.org/devguide/wallets.html>.
- [63] Bitcoin.org, “What is bitcoin cash?” *Bitcoin.org*, 2023. [Online]. Available: <https://www.bitcoin.com/get-started/what-is-bitcoin-cash/#:~:text=While%5C%20Bitcoin%5C%20typically%5C%20processes%5C%20between%5C%20transaction%5C%20speed%5C%20and%5C%20reliability..>
- [64] bitcoin.org, “Bitcoin developer guide,” *Bitcoin.org*, 2020. [Online]. Available: <https://developer.bitcoin.org/devguide/index.html>.
- [65] bitcoin.org, “Bitcoin developer guide: Blockchain,” *Bitcoin.org*, 2020. [Online]. Available: https://developer.bitcoin.org/devguide/block_chain.html.
- [66] bitcoin.org, “Bitcoin developer guide: Contract,” *Bitcoin.org*, 2020. [Online]. Available: <https://developer.bitcoin.org/devguide/contracts.html>.
- [67] bitcoin.org, “Bitcoin developer guide: Mining,” *Bitcoin.org*, 2020. [Online]. Available: <https://developer.bitcoin.org/devguide/mining.html>.
- [68] bitcoin.org, “Bitcoin developer guide: Reference transactions,” *Bitcoin.org*, 2020. [Online]. Available: <https://developer.bitcoin.org/reference/transactions.html>.
- [69] bitcoin.org, “Bitcoin developer guide: Transactions,” *Bitcoin.org*, 2020. [Online]. Available: <https://developer.bitcoin.org/devguide/transactions.html>.
- [70] bitcoin.org, “Bitcoin developer reference: Block chain,” *Bitcoin.org*, 2020. [Online]. Available: https://developer.bitcoin.org/reference/block_chain.html?highlight=block%5C%20version.
- [71] Bitcoincash.org, “The history of bitcoin cash,” *Bitcoincash.org*, 2023. [Online]. Available: <https://bitcoincash.org/>.
- [72] C. Blenkinsop, “Blockchain’s scaling problem explained,” *Cointelegraph*, 2018.

- [73] U. Bodkhe, D. Mehta, S. Tanwar, P. Bhattacharya, P. K. Singh, and W.-C. Hong, "A survey on decentralized consensus mechanisms for cyber physical systems," *IEEE Access*, vol. 8, pp. 54 371–54 401, 2020, ISSN: 2169-3536. DOI: 10.1109/access.2020.2981415.
- [74] U. Bodkhe, S. Tanwar, K. Parekh, *et al.*, "Blockchain for industry 4.0: A comprehensive review," *IEEE Access*, vol. 8, pp. 79 764–79 800, 2020.
- [75] D. Boneh and M. Franklin, "Efficient generation of shared rsa keys," in *Advances in Cryptology—CRYPTO'97: 17th Annual International Cryptology Conference Santa Barbara, California, USA August 17–21, 1997 Proceedings 17*, Springer, 1997, pp. 425–439.
- [76] K. Bosworth, M. G. Lee, S. Jaweed, and T. Wright, "Entities, identities, identifiers and credentials—what does it all mean?" *BT Technology Journal*, vol. 23, no. 4, p. 25, 2005.
- [77] M. Bouguerra, "Fingerprint recognition and classification," Ph.D. dissertation, Larbi Tebessi University-Tebessa, 2022.
- [78] H. Al-Breiki, M. H. U. Rehman, K. Salah, and D. Svetinovic, "Trustworthy blockchain oracles: Review, comparison, and open research challenges," *IEEE Access*, vol. 8, pp. 85 675–85 685, 2020.
- [79] A. Broaddus and C. Gertz, "Tolling heavy goods vehicles," *Transportation Research Record: Journal of the Transportation Research Board*, vol. 2066, no. 1, pp. 106–113, 2008, ISSN: 0361-1981 2169-4052. DOI: 10.3141/2066-12.
- [80] S. Brooks, S. Brooks, M. Garcia, N. Lefkowitz, S. Lightman, and E. Nadeau, *An introduction to privacy engineering and risk management in federal systems*. US Department of Commerce, National Institute of Standards and Technology MD . . . , 2017.
- [81] D. R. L. Brown, "Recommended elliptic curve domain parameters," *Certicom Research, Canada*, 2010.
- [82] M. Brown, E. Peköz, and S. Ross, "Blockchain double-spend attack duration," *Probability in the Engineering and Informational Sciences*, vol. 35, no. 4, pp. 858–866, 2021. DOI: 10.1017/S0269964820000212.
- [83] S. Browne, "Digital epidermalization: Race, identity and biometrics," *Critical Sociology*, vol. 36, no. 1, pp. 131–150, 2010.
- [84] R. Brubaker and F. Cooper, "Beyond "identity"," *Theory and society*, vol. 29, no. 1, pp. 1–47, 2000.

- [85] J. Buchmann, E. Dahmen, and M. Schneider, “Merkle tree traversal revisited,” in *Post-Quantum Cryptography: Second International Workshop, PQCrypto 2008 Cincinnati, OH, USA, October 17-19, 2008 Proceedings 2*, Springer, 2008, pp. 63–78.
- [86] C. Busch, “Iso/iec standard 24745–biometric information protection,” in *BTP Workshop, Paris*, vol. 13, 2012.
- [87] V. Buterin, “Bootstrapping a decentralized autonomous corporation: Part i,” *Bitcoin Magazine*, vol. 19, 2013.
- [88] V. Buterin *et al.*, “A next-generation smart contract and decentralized application platform,” *white paper*, vol. 3, no. 37, pp. 2–1, 2014.
- [89] V. Buterin, “Daos, dacs, das and more: An incomplete terminology guide,” *Ethereum Blog*, vol. 6, p. 2014, 2014.
- [90] V. Buterin, “Merkling in ethereum,” 2015. [Online]. Available: <https://blog.ethereum.org/2015/11/15/merkle-in-ethereum>.
- [91] V. Buterin, “Ethereum whitepaper,” *Ethereum.org*, 2023. [Online]. Available: <https://ethereum.org/en/whitepaper/>.
- [92] A. O. Callahan and John, “The horcrux protocol a method for decentralized biometric based self sovereign identity,” *International Joint Conference on Neural Networks (IJCNN)*, 2018, ISSN: 978-1-5090-6014-6/18/\$31.00 ©2018 IEEE.
- [93] cambridge.org, “Identity,” *dictionary.cambridge.org*, 2023. [Online]. Available: <https://dictionary.cambridge.org/dictionary/english/identity>.
- [94] J. Camp, “Digital identity,” *IEEE Technology and society Magazine*, vol. 23, no. 3, pp. 34–41, 2004.
- [95] K. Cao and A. K. Jain, “Automated latent fingerprint recognition,” *IEEE transactions on pattern analysis and machine intelligence*, vol. 41, no. 4, pp. 788–800, 2018.
- [96] Y. Cao and L. Yang, “A survey of identity management technology,” in *2010 IEEE International Conference on Information Theory and Information Security*, IEEE, 2010, pp. 287–293.
- [97] R. Casado-Vara, A. González-Briones, J. Prieto, and J. M. Corchado, “Smart contract for monitoring and control of logistics activities: Pharmaceutical utilities case study,” in *International Joint Conference SOCO’18-CISIS’18-ICEUTE’18: San Sebastián, Spain, June 6-8, 2018 Proceedings 13*, Springer, 2019, pp. 509–517.

- [98] M. Castro and B. Liskov, “Practical byzantine fault tolerance,” in *Proceedings of the Third Symposium on Operating Systems Design and Implementation*, ser. OSDI '99, New Orleans, Louisiana, USA: USENIX Association, 1999, pp. 173–186, ISBN: 1880446391.
- [99] L. Cavallaro, J. Kinder, and J. Domingo-Ferrer, “Privacy protection in distributed fingerprint-based authentication,” *WPES '19*, 2019. DOI: 10.1145/3338498.
- [100] K. Chan and A. Milne, “The global legal entity identifier system: How can it deliver?” *Journal of Risk and Financial Management*, vol. 12, no. 1, 2019, ISSN: 1911-8074. DOI: 10.3390/jrfm12010039.
- [101] K. Chandra, M. Mushtaq, *et al.*, “Digital passport and visa asset management using private and permissioned blockchain,” *arXiv preprint arXiv:2107.06849*, 2021.
- [102] D. Chang, S. Garg, M. Hasan, and S. Mishra, “Cancelable multi-biometric approach using fuzzy extractor and novel bit-wise encryption,” *IEEE Transactions on Information Forensics and Security*, vol. 15, pp. 3152–3167, 2020, ISSN: 1556-6013 1556-6021. DOI: 10.1109/tifs.2020.2983250.
- [103] Y. Chang, E. Iakovou, and W. Shi, “Blockchain in global supply chains and cross border trade: A critical synthesis of the state-of-the-art, challenges and opportunities,” *International Journal of Production Research*, vol. 58, no. 7, pp. 2082–2099, 2020.
- [104] N. Chaudhry and M. M. Yousaf, “Consensus algorithms in blockchain: Comparative analysis, challenges and opportunities,” in *2018 12th International Conference on Open Source Systems and Technologies (ICOSST)*, IEEE, 2018, pp. 54–63.
- [105] B. Chen, J. Wan, A. Celesti, D. Li, H. Abbas, and Q. Zhang, “Edge computing in iot-based manufacturing,” *IEEE Communications Magazine*, vol. 56, no. 9, pp. 103–109, 2018, ISSN: 0163-6804 1558-1896. DOI: 10.1109/mcom.2018.1701231.
- [106] T. Chen and Y.-C. Lin, “A digital equipment identifier system,” *Journal of Intelligent Manufacturing*, vol. 28, no. 5, pp. 1159–1169, 2015, ISSN: 0956-5515 1572-8145. DOI: 10.1007/s10845-015-1071-3.
- [107] D. Cheriton, “The v distributed system,” *Communications of the ACM*, vol. 31, no. 3, pp. 314–333, 1988, ISSN: 0001-0782 1557-7317. DOI: 10.1145/42392.42400.
- [108] M. Chiang, B. Balasubramanian, F. Bonomi, *et al.*, *Fog for 5G and IoT*. Wiley Online Library, 2017, vol. 288.

- [109] M. Chiang and T. Zhang, “Fog and iot: An overview of research opportunities,” *IEEE Internet of things journal*, vol. 3, no. 6, pp. 854–864, 2016.
- [110] G. Chollet, A. Jimenez, D. Petrovska-Delacrétaz, and B. Raj, “Privacy preserving biometric identity verification,” in *COST IC1206 Training School*, 2017.
- [111] G. Chollet and A. Jiménez, “Privacy preserving biometric identity verification,” *Carnegie Mellon Lecture PPT*, 2017.
- [112] M. S. Choudhry, V. Jetli, S. Mathur, and Y. Saini, “A review on behavioural biometric authentication,” in *2022 International Conference on Computing, Communication, Security and Intelligent Systems (IC3SIS)*, IEEE, 2022, pp. 1–6.
- [113] S. S. Chow, Z. Lai, C. Liu, E. Lo, and Y. Zhao, “Sharding blockchain,” in *2018 IEEE International Conference on Internet of Things (iThings) and IEEE Green Computing and Communications (GreenCom) and IEEE Cyber, Physical and Social Computing (CPSCom) and IEEE Smart Data (SmartData)*, 2018, pp. 1665–1665. DOI: 10.1109/Cybermatics_2018.2018.00277.
- [114] K. Christidis and M. Devetsikiotis, “Blockchains and smart contracts for the internet of things,” *IEEE Access*, vol. 4, pp. 2292–2303, 2016, ISSN: 2169-3536. DOI: 10.1109/access.2016.2566339.
- [115] L. L. Claudia Loebbecke and D. Niederle, “Blockchain technology impacting the role of trust in transactions reflections in the case of trading diamonds,” *ECIS 2018 Proceedings*, vol. 11-29-2018, 2018.
- [116] T. Coates, “Uris, urls, and urns: Clarifications and recommendations 1.0,” *Report from the joint W3C/IETF URI Planning Interest Group*, 2001. [Online]. Available: <https://www.w3.org/TR/uri-clarification/#RFC2717>.
- [117] P. Coli, G. L. Marcialis, and F. Roli, “Fingerprint silicon replicas: Static and dynamic features for vitality detection using an optical capture device,” *International Journal of Image and Graphics*, vol. 8, no. 04, pp. 495–512, 2008.
- [118] E. Commission, “Smart borders - background,” *Migration and Home Affairs*, 2023. [Online]. Available: https://home-affairs.ec.europa.eu/pages/page/smart-borders-background_en.
- [119] *Confirmation*, Bitcoin open source guidance, 2018. DOI: <https://en.bitcoin.it/wiki/Confirmation>.
- [120] L. W. Cong and Z. He, “Blockchain disruption and smart contracts,” *The Review of Financial Studies*, vol. 32, no. 5, pp. 1754–1797, 2019, ISSN: 0893-9454 1465-7368. DOI: 10.1093/rfs/hhz007.

- [121] W. W. W. Consortium *et al.*, “Verifiable credentials data model 1.0: Expressing verifiable information on the web,” <https://www.w3.org/TR/vc-data-model/?# core-data-model>, 2019.
- [122] D. H. Craft, “Resource management in a decentralized system,” *ACM. Computer Laboratory, University of Cambridge.*, 1983, ISSN: 0-89791-115-6/83/010/0011.
- [123] A. Crespo, D. Levi, and L. Garcia, *A decentralized oracle network protocol*, 2017.
- [124] crypto.com, “A deep dive into blockchain scalability,” *crypto.com*, 2020. [Online]. Available: <https://crypto.com/university/blockchain-scalability#:~:text=While%20Visa%20can%20process%20up,capability%20to%20achieve%20mass%20adoption..>
- [125] V. C. D. and Akila, “A survey on biometric authentication systems in cloud to combat identity theft,” *Journal of critical reviews*, vol. 7, no. 03, 2020, ISSN: 23945125. DOI: 10.31838/jcr.07.03.97.
- [126] B. Da, P. P. Esnault, S. Hu, and C. Wang, “Identity identifier-enabled networks (ideas) for iot,” pp. 412–425, 2018. DOI: 978-1-4673-9944-9/18/\\$31.002018IEEE.
- [127] W. Dahea and H. Fadewar, “Multimodal biometric system: A review,” *International Journal of Research in Advanced Engineering and Technology*, vol. 4, no. 1, pp. 25–31, 2018.
- [128] H.-N. Dai, Z. Zheng, and Y. Zhang, “Blockchain for internet of things: A survey,” *IEEE Internet of Things Journal*, vol. 6, no. 5, pp. 8076–8094, 2019, ISSN: 2327-4662 2372-2541. DOI: 10.1109/jiot.2019.2920987.
- [129] A. Dantcheva, P. Elia, and A. Ross, “What else does your biometric data reveal? a survey on soft biometrics,” *IEEE Transactions on Information Forensics and Security*, vol. 11, no. 3, pp. 441–467, 2015.
- [130] S. Davidson, P. De Filippi, and J. Potts, “Blockchains and the economic institutions of capitalism,” *Journal of Institutional Economics*, vol. 14, no. 4, pp. 639–658, 2018.
- [131] S. De Angelis, L. Aniello, R. Baldoni, F. Lombardi, A. Margheri, and V. Sassone, “Pbft vs proof-of-authority: Applying the cap theorem to permissioned blockchain,” 2018.
- [132] M. De Donno, K. Tange, and N. Dragoni, “Foundations and evolution of modern computing paradigms: Cloud, iot, edge, and fog,” *Ieee Access*, vol. 7, pp. 150 936–150 948, 2019.

- [133] P. De Filippi and G. McMullen, “Governance of blockchain systems: Governance of and by distributed infrastructure,” Ph.D. dissertation, Blockchain Research Institute and COALA, 2018.
- [134] E. Deirmentzoglou, G. Papakyriakopoulos, and C. Patsakis, “A survey on long-range attacks for proof of stake protocols,” *IEEE Access*, vol. 7, pp. 28 712–28 725, 2019.
- [135] O. Delgado-Mohatar, J. Fierrez, R. Tolosana, and R. Vera-Rodriguez, “Biometric template storage with blockchain: A first look into cost and performance tradeoffs,” in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition workshops*, 2019.
- [136] O. Delgado-Mohatar, J. Fierrez, R. Tolosana, and R. Vera-Rodriguez, “Blockchain meets biometrics: Concepts, application to template protection, and trends,” *arXiv preprint arXiv:2003.09262*, 2020.
- [137] K. Delmolino, M. Arnett, A. Kosba, A. Miller, and E. Shi, “Step by step towards creating a safe smart contract: Lessons and insights from a cryptocurrency lab,” in *Financial Cryptography and Data Security: FC 2016 International Workshops, BITCOIN, VOTING, and WAHC, Christ Church, Barbados, February 26, 2016, Revised Selected Papers 20*, Springer, 2016, pp. 79–94.
- [138] B. Demil and X. Lecocq, “Neither market nor hierarchy nor network: The emergence of bazaar governance,” *Organization studies*, vol. 27, no. 10, pp. 1447–1466, 2006.
- [139] X. Deng, C. Tian, F. Chen, H. Xian, and V. Sharma, “Designated-verifier anonymous credential for identity management in decentralized systems,” *Mobile Information Systems*, vol. 2021, pp. 1–15, 2021, ISSN: 1875-905X 1574-017X. DOI: 10.1155/2021/2807395.
- [140] R. Dhamija and L. Dusseault, “The seven flaws of identity management: Usability and security challenges,” *IEEE Security & Privacy*, vol. 6, no. 2, pp. 24–29, 2008.
- [141] Y. Al-Dhuraibi, F. Paraiso, N. Djarallah, and P. Merle, “Elasticity in cloud computing: State of the art and research challenges,” *IEEE Transactions on Services Computing*, vol. 11, no. 2, pp. 430–447, 2018, ISSN: 1939-1374. DOI: 10.1109/tsc.2017.2711009.
- [142] A. Dhurat, P. Magal, M. Chheda, and D. Ingle, “Gateless electronic toll collection using rfid,” *IOSR Journal of Computer Engineering*, vol. 16, no. 2, pp. 73–80, 2014, ISSN: 22788727 22780661. DOI: 10.9790/0661-16267380.

- [143] O. Dib, K.-L. Brousmiche, A. Durand, E. Thea, and E. B. Hamida, "Consortium blockchains: Overview, applications and challenges," *International Journal on Advances in Telecommunications*, vol. vol 11 no 1 2, pp. 51–64, 2018.
- [144] T. Dierks and E. Rescorla, "The transport layer security (tls) protocol," 2008. [Online]. Available: <https://datatracker.ietf.org/doc/html/rfc5246>.
- [145] R. K. Dinakaran, P. Easom, A. Bouridane, *et al.*, "Deep learning based pedestrian detection at distance in smart cities," in *Intelligent Systems and Applications: Proceedings of the 2019 Intelligent Systems Conference (IntelliSys) Volume 2*, Springer, 2020, pp. 588–593.
- [146] A. D. Dinesh, C. D. P. Reddy, G. V. Gopi, R. Jain, and T. Shankar, "A durable biometric authentication scheme via blockchain," in *2021 International Conference on Advances in Electrical, Computing, Communication and Sustainable Technologies (ICAECT)*, IEEE, 2021, pp. 1–5.
- [147] W. Ding, J. Hou, J. Li, *et al.*, "Desci based on web3 and dao: A comprehensive overview and reference model," *IEEE Transactions on Computational Social Systems*, vol. 9, no. 5, pp. 1563–1573, 2022.
- [148] A. Dmitrienko, C. Liebchen, C. Rossow, and A.-R. Sadeghi, "On the (in) security of mobile two-factor authentication," in *Financial Cryptography and Data Security: 18th International Conference, FC 2014, Christ Church, Barbados, March 3-7, 2014, Revised Selected Papers 18*, Springer, 2014, pp. 365–383.
- [149] C. K. Dominicini, M. A. Simplício, R. R. M. Sakuragui, T. C. M. B. Carvalho, M. Näslund, and M. Pourzandi, "Threat modeling an identity management system for mobile internet," 2010.
- [150] A. Dorri, S. S. Kanhere, and R. Jurdak, "Towards an optimized blockchain for iot," in *Proceedings of the second international conference on Internet-of-Things design and implementation*, 2017, pp. 173–178.
- [151] J. R. Douceur, "The sybil attack," in *Peer-to-Peer Systems: First International Workshop, IPTPS 2002 Cambridge, MA, USA, March 7–8, 2002 Revised Papers 1*, Springer, 2002, pp. 251–260.
- [152] N. Dragoni, S. Giallorenzo, A. L. Lafuente, *et al.*, "Microservices: Yesterday, today, and tomorrow," *Present and ulterior software engineering*, pp. 195–216, 2017.

- [153] N. Dragoni, I. Lanese, S. T. Larsen, M. Mazzara, R. Mustafin, and L. Safina, “Microservices: How to make your application scale,” in *Perspectives of System Informatics: 11th International Andrei P. Ershov Informatics Conference, PSI 2017, Moscow, Russia, June 27-29, 2017, Revised Selected Papers 11*, Springer, 2018, pp. 95–104.
- [154] P. Drozowski, C. Rathgeb, A. Dantcheva, N. Damer, and C. Busch, “Demographic bias in biometrics: A survey on an emerging challenge,” *IEEE Transactions on Technology and Society*, vol. 1, no. 2, pp. 89–103, 2020, ISSN: 2637-6415. DOI: 10.1109/tts.2020.2992344.
- [155] T. Duong, L. Fan, J. Katz, P. Thai, and H.-S. Zhou, “2-hop blockchain: Combining proof-of-work and proof-of-stake securely,” in *Computer Security—ESORICS 2020: 25th European Symposium on Research in Computer Security, ESORICS 2020, Guildford, UK, September 14–18, 2020, Proceedings, Part II*, Springer, 2020, pp. 697–712.
- [156] Q. DuPont, “Experiments in algorithmic governance: A history and ethnography of “the dao,” a failed decentralized autonomous organization,” in *Bitcoin and beyond*, Routledge, 2017, pp. 157–177.
- [157] R. Dwivedi, S. Dey, M. A. Sharma, and A. Goel, “A fingerprint based cryptobiometric system for secure communication,” *Journal of Ambient Intelligence and Humanized Computing*, vol. 11, pp. 1495–1509, 2020.
- [158] J. Eberhardt and S. Tai, “Zokrates-scalable privacy-preserving off-chain computations,” in *2018 IEEE International Conference on Internet of Things (iThings) and IEEE Green Computing and Communications (GreenCom) and IEEE Cyber, Physical and Social Computing (CPSCoM) and IEEE Smart Data (SmartData)*, IEEE, 2018, pp. 1084–1091.
- [159] J.-E. Ekberg, K. Kostiaainen, and N. Asokan, “Trusted execution environments on mobile devices,” in *Proceedings of the 2013 ACM SIGSAC conference on Computer & communications security*, 2013, pp. 1497–1498.
- [160] Y. El Faqir, J. Arroyo, and S. Hassan, “An overview of decentralized autonomous organizations on the blockchain,” in *Proceedings of the 16th international symposium on open collaboration*, 2020, pp. 1–8.
- [161] C. Elliott and P. Brodwin, “Identity and genetic ancestry tracing,” *Bmj*, vol. 325, no. 7378, pp. 1469–1471, 2002.
- [162] A. I. S. D. Enríquez and Á. Martín, “Digital identity the current state of affairs,” *Digital Identity: the current state of affairs*, vol. 18-01, 2018.
- [163] ensembl.org, “Stable ids,” 2023. [Online]. Available: https://www.ensembl.org/info/genome/stable_ids/index.html.

- [164] Ethereum.org, “Decentralized autonomous organizations (daos),” 2020. [Online]. Available: <https://ethereum.org/en/dao/#main-content>.
- [165] Ethereum.org, “Blocks,” *Ethereum.org*, 2023. [Online]. Available: <https://ethereum.org/en/developers/docs/blocks/>.
- [166] Ethereum.org, “Consensus mechanisms,” *Ethereum.org*, 2023. [Online]. Available: <https://ethereum.org/en/developers/docs/consensus-mechanisms/>.
- [167] Ethereum.org, “Ethereum virtual machine,” *Ethereum.org*, 2023. [Online]. Available: <https://ethereum.org/en/developers/docs/evm/>.
- [168] Ethereum.org, “Gas and fees,” *Ethereum.org*, 2023. [Online]. Available: <https://ethereum.org/en/developers/docs/gas/>.
- [169] Ethereum.org, “Prove of stake (pos),” *Ethereum.org*, 2023. [Online]. Available: <https://ethereum.org/en/developers/docs/consensus-mechanisms/pos/>.
- [170] Ethereum.org, “The merge,” *Ethereum.org*, 2023. [Online]. Available: <https://ethereum.org/en/upgrades/merge/#merge-and-scaling>.
- [171] Etherscan.io, “Average transaction fee chart,” 2023. [Online]. Available: <https://etherscan.io/chart/avg-txfee-usd>.
- [172] ETIASEU, “Travelling to europe from the uk with a biometric passport,” 2023. [Online]. Available: <https://www.etiaseu.co.uk/etias-news/travelling-europe-from-uk-biometric-passport>.
- [173] I. Eyal, A. E. Gencer, E. G. Sirer, and R. Van Renesse, “Bitcoin-ng: A scalable blockchain protocol,” in *13th {USENIX} symposium on networked systems design and implementation ({NSDI} 16)*, 2016, pp. 45–59.
- [174] J. Ezeobijesi and B. Bhanu, “Latent fingerprint image segmentation using deep neural network,” *Deep Learning for Biometrics*, pp. 83–107, 2017.
- [175] H. Fabric, “Deploying a smart contract to a channel,” in *Hyperledger fabric documentation: Key concepts*, 2023. [Online]. Available: https://hyperledger-fabric.readthedocs.io/en/release-2.5/deploy_chaincode.html#:~:text=End%5C%20users%5C%20interact%5C%20with%5C%20the,a%5C%20chaincode%5C%20on%5C%20their%5C%20peers..
- [176] H. Fabric, “Private data,” in *Hyperledger fabric documentation: Key concepts*, 2023. [Online]. Available: <https://hyperledger-fabric.readthedocs.io/en/latest/private-data/private-data.html>.
- [177] H. Fabric, “The ordering service,” in *Hyperledger fabric documentation: Key concepts*, 2023. [Online]. Available: https://hyperledger-fabric.readthedocs.io/en/release-2.5/orderer/ordering_service.html.

- [178] S. B. Far and A. I. Rad, “Applying digital twins in metaverse: User interface, security and privacy challenges,” *Journal of Metaverse*, vol. 2, no. 1, pp. 8–15, 2022.
- [179] G. Farahani, “Dynamic and robust method for detection and locating vehicles in the video images sequences with use of image processing algorithm,” *EURASIP Journal on Image and Video Processing*, vol. 2017, no. 1, 2017, ISSN: 1687-5281. DOI: 10.1186/s13640-017-0230-1.
- [180] C. Ferhaoui Cherifi, M. Deriche, and K.-W. Hidouci, “An improved revocable fuzzy vault scheme for face recognition under unconstrained illumination conditions,” *Arabian Journal for Science and Engineering*, vol. 44, no. 8, pp. 7203–7217, 2019, ISSN: 2193-567X 2191-4281. DOI: 10.1007/s13369-019-03916-5.
- [181] M. V. Ferreira, D. J. Moroz, D. C. Parkes, and M. Stern, “Dynamic posted-price mechanisms for the blockchain transaction-fee market,” in *Proceedings of the 3rd ACM Conference on Advances in Financial Technologies*, 2021, pp. 86–99.
- [182] D. Fett, R. Küsters, and G. Schmitz, “A comprehensive formal security analysis of oauth 2.0,” in *Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security*, 2016, pp. 1204–1215.
- [183] U. Fiege, A. Fiat, and A. Shamir, “Zero knowledge proofs of identity,” in *Proceedings of the nineteenth annual ACM symposium on Theory of computing*, 1987, pp. 210–217.
- [184] R. T. Fielding, “Relative uniform resource locators,” *IETF REC 1808*, 1995. [Online]. Available: <https://datatracker.ietf.org/doc/html/rfc1808>.
- [185] H. Finney, *The finney attack (the bitcoin talk forum)*, 2011.
- [186] U. Food and D. Administration, “Product codes and product code builder,” 2021. [Online]. Available: <https://www.fda.gov/industry/import-program-tools/product-codes-and-product-code-builder>.
- [187] J. T. Force, “Risk management framework for information systems and organizations,” *NIST Special Publication*, vol. 800, p. 37, 2018.
- [188] J. Franks, P. Hallam-Baker, J. Hostetler, *et al.*, “Http authentication: Basic and digest access authentication,” 1999. [Online]. Available: <https://datatracker.ietf.org/doc/html/rfc2617>.

- [189] J.-S. Fu, Y. Liu, H.-C. Chao, B. K. Bhargava, and Z.-J. Zhang, "Secure data storage and searching for industrial iot by integrating fog computing and cloud computing," *IEEE Transactions on Industrial Informatics*, vol. 14, no. 10, pp. 4519–4528, 2018, ISSN: 1551-3203 1941-0050. DOI: 10.1109/tii.2018.2793350.
- [190] Y. Fu and J. Zhu, "Operation mechanisms for intelligent logistics system: A blockchain perspective," *IEEE Access*, vol. 7, pp. 144 202–144 213, 2019, ISSN: 2169-3536. DOI: 10.1109/access.2019.2945078.
- [191] B. Fuller, L. Reyzin, and A. Smith, "When are fuzzy extractors possible?" *IEEE Transactions on Information Theory*, vol. 66, no. 8, pp. 5282–5298, 2020, ISSN: 0018-9448 1557-9654. DOI: 10.1109/tit.2020.2984751.
- [192] K. Gai, M. Qiu, and H. Zhao, "Privacy-preserving data encryption strategy for big data in mobile cloud computing," *IEEE Transactions on Big Data*, pp. 1–1, 2017, ISSN: 2332-7790. DOI: 10.1109/tbdata.2017.2705807.
- [193] J. Galbally, R. Haraksim, and L. Beslay, "A study of age and ageing in fingerprint biometrics," *IEEE Transactions on Information Forensics and Security*, vol. 14, no. 5, pp. 1351–1365, 2019, ISSN: 1556-6013 1556-6021. DOI: 10.1109/tifs.2018.2878160.
- [194] E. Garcia Ribera, "Design and implementation of a proof-of-stake consensus algorithm for blockchain," B.S. thesis, Universitat Politècnica de Catalunya, 2018.
- [195] J. M. Gauch, "Image segmentation and analysis via multiscale gradient watershed hierarchies," *IEEE transactions on image processing*, vol. 8, no. 1, pp. 69–79, 1999.
- [196] M. Ghafourian, B. Sumer, R. Vera-Rodriguez, *et al.*, "Combining blockchain and biometrics: A survey on technical aspects and a first legal analysis," *arXiv preprint arXiv:2302.10883*, 2023.
- [197] M. H. Ghahramani, M. Zhou, and C. T. Hon, "Toward cloud computing qos architecture: Analysis of cloud systems and cloud services," *IEEE/CAA Journal of Automatica Sinica*, vol. 4, no. 1, pp. 6–18, 2017, ISSN: 2329-9266 2329-9274. DOI: 10.1109/jas.2017.7510313.
- [198] K. Gilani, E. Bertin, J. Hatin, and N. Crespi, "A survey on blockchain-based identity management and decentralized privacy for personal data," 2020.
- [199] A. Goel, A. Agarwal, M. Vatsa, R. Singh, and N. Ratha, "Securing cnn model and biometric template using blockchain," *10th IEEE International Conference on Biometrics Theory, Applications and Systems, BTAS 2019*, 2019, ISSN: 978-1-7281-1522-1/19/\$31.00 2019 IEEE.

- [200] E. Gonzalez-Sosa, J. Fierrez, R. Vera-Rodriguez, and F. Alonso-Fernandez, “Facial soft biometrics for recognition in the wild: Recent works, annotation, and cots evaluation,” *IEEE Transactions on Information Forensics and Security*, vol. 13, no. 8, pp. 2001–2014, 2018, ISSN: 1556-6013 1556-6021. DOI: 10.1109/tifs.2018.2807791.
- [201] I. Gorton, J. Klein, and A. Nurgaliev, “Architecture knowledge for evaluating scalable databases,” in *2015 12th Working IEEE/IFIP Conference on Software Architecture*, IEEE, 2015, pp. 95–104.
- [202] U. Gov.com, “Get a passport photo,” *Passport, travel and living abroad*, [Online]. Available: <https://www.gov.uk/photos-for-passports>.
- [203] Gov.uk, “Guidance: Official customs seals and trader sealing,” 2023. [Online]. Available: <https://www.gov.uk/guidance/official-customs-seals-and-trader-sealing-notice-205--2#find-out-if-your-goods-need-sealing>.
- [204] P. Grassi, M. Garcia, J. Fenton, *et al.*, “Nist digital identity guidelines,” <https://csrc.nist.gov/publications/detail/sp/800-63/3/final>, 2020.
- [205] A. Greenberg, J. Hamilton, D. A. Maltz, and P. Patel, *The cost of a cloud: Research problems in data center networks*, 2008.
- [206] A. Guarino, “Information security standards in critical infrastructure protection,” in *Information Security Solutions Europe*, 2015.
- [207] J. Guerrero-Ibanez, S. Zeadally, and J. Contreras-Castillo, “Sensor technologies for intelligent transportation systems,” *Sensors (Basel)*, vol. 18, no. 4, 2018, ISSN: 1424-8220 (Electronic) 1424-8220 (Linking). DOI: 10.3390/s18041212. [Online]. Available: <https://www.ncbi.nlm.nih.gov/pubmed/29659524>.
- [208] H. Gupta and D. Janakiram, “Cdag: A serialized blockdag for permissioned blockchain,” *arXiv preprint arXiv:1910.08547*, 2019.
- [209] N. Hackius and M. Petersen, “Blockchain in logistics and supply chain: trick or treat,” *Proceedings of the Hamburg International Conference of Logistics*, vol. Vol. 23, ISBN 978-3-7450-4328-0, pp. 3–18, 2017. DOI: 10.15480/882.1444.
- [210] J. Häkli, “Biometric identities,” *Progress in Human Geography*, vol. 31, no. 2, pp. 139–141, 2007.
- [211] N. Haller, C. Metz, P. Nesser, and M. Straw, “A one-time password system,” Tech. Rep., 1998.
- [212] H. Halpin, “Vision a critique of immunity passports and w3c decentralized identifiers,” 2020. DOI: [arXiv:2012.00136v1](https://arxiv.org/abs/2012.00136v1) [cs.CR] 30Nov2020.

- [213] T. Hamer, K. Taylor, K. S. Ng, and A. Tiu, "Private digital identity on blockchain," *CEUR_{WS}*, vol. 1-2599, 2019.
- [214] K. Hamilton-Duffy, R. Grant, and A. Gropper, "Use cases and requirements for decentralized identifiers," 2021. [Online]. Available: <https://www.w3.org/TR/did-use-cases/>.
- [215] E. Hammer, "Oauth 2.0 and the road to hell," *Retrieved September*, vol. 10, p. 2012, 2012.
- [216] H. Handschub and H. Gilbert, "Evaluation report security level of cryptography-sha-256," *Journal of Women s Health*, 2002.
- [217] M. Hansen, P. Berlich, J. Camenisch, S. Clauß, A. Pfitzmann, and M. Waidner, "Privacy-enhancing identity management," *Information security technical report*, vol. 9, no. 1, pp. 35–44, 2004.
- [218] E. Hardt, "The oauth 2.0 authorization framework," Internet Engineering Task Force (IETF), 2012. [Online]. Available: <https://www.rfc-editor.org/rfc/rfc6749?https://www.rfc-editor.org/rfc/rfc6750?https://www.rfc-editor.org/rfc/rfc7662?https://www.rfc-editor.org/rfc/rfc5849?https://www.rfc-editor.org/rfc/rfc2617?https://www.rfc-editor.org/rfc/rfc7519?https://oauth.net/2/?https://www.youtube.com/watch?v=CHzERullHe8?https://datatracker.ietf.org/doc/html/draft-ietf-oauth-security-topics-13>.
- [219] S. Hassan and P. De Filippi, "Decentralized autonomous organization," *Internet Policy Review*, vol. 10, no. 2, pp. 1–10, 2021.
- [220] A. B. Hassanat, M. Alkasassbeh, M. Al-Awadi, and A. Esra'a, "Colour-based lips segmentation method using artificial neural networks," in *2015 6th international conference on information and communication systems (ICICS)*, IEEE, 2015, pp. 188–193.
- [221] D. He, W. Zhao, and S. Wu, "Security analysis of a dynamic id-based authentication scheme for multi-server environment using smart cards," *Int. J. Netw. Secur.*, vol. 15, no. 5, pp. 350–356, 2013.
- [222] N. Heijmink, E. Poll, and R. Hofboer, *Secure single sign-on*, 2015.
- [223] R. Heinrich, A. Van Hoorn, H. Knoche, *et al.*, "Performance engineering for microservices: Research challenges and directions," in *Proceedings of the 8th ACM/SPEC on International Conference on Performance Engineering Companion*, 2017, pp. 223–226.

- [224] H. Hellani, A. E. Samhat, M. Chamoun, H. El Ghor, and A. Serhrouchni, "On blockchain technology: Overview of bitcoin and future insights," in *2018 IEEE International Multidisciplinary Conference on Engineering Technology (IMCET)*, IEEE, 2018, pp. 1–8.
- [225] B. Hendricks, "Bitcoin what is in the whitepaper," 2017.
- [226] B. Hill and D. Baghdasaryan, "Fido metadata service," *fidoalliance.org*, 2018. [Online]. Available: <https://fidoalliance.org/specs/fido-v2.0-id-20180227/fido-metadata-service-v2.0-id-20180227.html>.
- [227] B. Hill, D. Balfanz, and D. Baghdasaryan, "Fido appid and facet specification," *fidoalliance.org*, 2017. [Online]. Available: <https://fidoalliance.org/specs/fido-uaf-v1.1-ps-20170202/fido-appid-and-facets-v1.1-ps-20170202.pdf>.
- [228] J. E. Hopcroft, R. Motwani, and J. D. Ullman, "Introduction to automata theory, languages, and computation," *Acm Sigact News*, vol. 32, no. 1, pp. 60–65, 2001.
- [229] S. Hossain and G. Chetty, "Human identity verification by using physiological and behavioural biometric traits," *International Journal of Bioscience, Biochemistry and Bioinformatics*, vol. 1, no. 3, p. 199, 2011.
- [230] R. Housley, "Public key infrastructure (pki)," *The internet encyclopedia*, 2004.
- [231] R. Housley, W. Ford, W. Polk, and D. Solo, "Internet x. 509 public key infrastructure certificate and crl profile," Tech. Rep., 1999.
- [232] J. J. Howard, Y. B. Sirotin, and A. R. Vemury, "The effect of broad and specific demographic homogeneity on the imposter distributions and false match rates in face recognition algorithm performance," in *2019 IEEE 10th International Conference on Biometrics Theory, Applications and Systems (BTAS)*, IEEE, 2019, pp. 1–8.
- [233] D. Huang, X. Ma, and S. Zhang, "Performance analysis of the raft consensus algorithm for private blockchains," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 50, no. 1, pp. 172–181, 2019.
- [234] H. G. Huang and Chin-Tser, "Self-sovereign dynamic digital identities based on blockchain technology," *IEEE Southeast Conference 2019*, 2019.
- [235] J. Hübner, D. Vuckovac, E. Fleisch, and A. Ilic, "Fintechs and the new wave of financial intermediaries," 2019. [Online]. Available: <http://www.alexandria.unisg.ch/259703/>.
- [236] T. Huynh-The, T. R. Gadekallu, W. Wang, *et al.*, "Blockchain for the metaverse: A review," *Future Generation Computer Systems*, 2023.

- [237] hyperledger-fabric.readthedocs.io, “Hyperledger fabric: Performance considerations,” 2023. [Online]. Available: <https://hyperledger-fabric.readthedocs.io/en/latest/performance.html>.
- [238] Hyperledger.github.io, “Hyperledger caliper: Getting started,” 2023. [Online]. Available: <https://hyperledger.github.io/caliper/v0.5.0/getting-started/>.
- [239] IBM, “Namespaces of identifiers,” *IBM documentation*, 2021. [Online]. Available: <https://www.ibm.com/docs/en/zos/2.4.0?topic=scope-namespaces-identifiers>.
- [240] IBM.com, “Http basic authentication,” 2023. [Online]. Available: <https://www.ibm.com/docs/en/cics-ts/5.4?topic=concepts-http-basic-authentication>.
- [241] IBM.com, “What is single sign-on?,” 2023. [Online]. Available: ibm.com/uk-en/topics/single-sign-on.
- [242] M. H. Ibrahim, I. Ali, I. Ibrahim, and A. El-Sawi, “A robust threshold elliptic curve digital signature providing a new verifiable secret sharing scheme,” in *2003 46th Midwest Symposium on Circuits and Systems*, IEEE, vol. 1, 2003, pp. 276–280.
- [243] S. Z. S. Idrus, E. Cherrier, C. Rosenberger, and J.-J. Schwartzmann, “A review on authentication methods,” *Australian Journal of Basic and Applied Sciences*, vol. 7, no. 5, pp. 95–107, 2013.
- [244] IEEE, “Ieee standard for biometric open protocol,” *IEEE Std 2410-2019 (Revision of IEEE Std 2410-2017)*, pp. 1–25, 2019. DOI: 10.1109/IEEESTD.2019.8751181.
- [245] S. Ilankumaran and C. Deisy, “Multi-biometric authentication system using finger vein and iris in cloud computing,” *Cluster Computing*, vol. 22, no. S1, pp. 103–117, 2018, ISSN: 1386-7857 1573-7543. DOI: 10.1007/s10586-018-1824-9.
- [246] D. Impedovo and G. Pirlo, “Automatic signature verification: The state of the art,” *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, vol. 38, no. 5, pp. 609–635, 2008.
- [247] M. R. Islam, M. M. Rahman, M. Mahmud, M. A. Rahman, M. H. S. Mohamad, *et al.*, “A review on blockchain security issues and challenges,” in *2021 IEEE 12th Control and System Graduate Research Colloquium (ICSGRC)*, IEEE, 2021, pp. 227–232.

- [248] I. J. Jacob, P. Betty, P. E. Darney, S. Raja, Y. H. Robinson, and E. G. Julie, "Biometric template security using dna codec based transformation," *Multimedia Tools and Applications*, vol. 80, pp. 7547–7566, 2021.
- [249] N. Jahan and S. Reno, "Utilizing hyperledger-based private blockchain to secure e-passport management," in *Data Intelligence and Cognitive Informatics: Proceedings of ICDICI 2022*, Springer, 2022, pp. 579–593.
- [250] A. K. Jain, A. Ross, and S. Prabhakar, "An introduction to biometric recognition," *IEEE Transactions on circuits and systems for video technology*, vol. 14, no. 1, pp. 4–20, 2004.
- [251] F. Jansen, J. Sánchez-Monedero, and L. Dencik, "Biometric identity systems in law enforcement and the politics of (voice) recognition: The case of siip," *Big Data & Society*, vol. 8, no. 2, p. 20 539 517 211 063 604, 2021.
- [252] M. Jansen, F. Hdhili, R. Gouiaa, and Z. Qasem, "Do smart contract languages need to be turing complete?" In *Blockchain and Applications: International Congress*, Springer, 2020, pp. 19–26.
- [253] P. Jarupunphol, "A critical analysis of 3-d secure," *Proceedings of the 3rd Electronic Commerce Research and Development (E-COM-03)*, pp. 87–94, 2003.
- [254] C. Jentsch, "Decentralized autonomous organization to automate governance," *White paper, November*, 2016.
- [255] P. Jesus, C. Baquero, and P. S. Almeida, "Id generation in mobile environments," 2006.
- [256] L. Jiang, T. Zhao, C. Bai, A. Yong, and M. Wu, "A direct fingerprint minutiae extraction approach based on convolutional neural networks," in *2016 International Joint Conference on Neural Networks (IJCNN)*, IEEE, 2016, pp. 571–578.
- [257] D. Johnson, A. Menezes, and S. Vanstone, "The elliptic curve digital signature algorithm (ecdsa)," *International Journal of Information Security*, vol. 1, no. 1, pp. 36–63, 2014, ISSN: 1615-5262. DOI: 10.1007/s102070100002.
- [258] D. Johnston, S. O. Yilmaz, J. Kandah, *et al.*, "The general theory of decentralized applications, dapps," 2014.
- [259] S. Jokić, A. S. Cvetković, S. Adamović, N. Ristić, and P. Spalević, "Comparative analysis of cryptocurrency wallets vs traditional wallets," *ekonomika*, vol. 65, no. 3, pp. 65–75, 2019.
- [260] K. P. Jørgensen and R. Beck, "Universal wallets," *Business & Information Systems Engineering*, pp. 1–11, 2022.

- [261] S. Josefsson and I. Liusvaara, “Edwards-curve digital signature algorithm (eddsa),” Tech. Rep., 2017.
- [262] A. Juels, “Rfid security and privacy: A research survey,” *IEEE journal on selected areas in communications*, vol. 24, no. 2, pp. 381–394, 2006.
- [263] H. Jung and Y. Heo, “Fingerprint liveness map construction using convolutional neural network,” *Electronics Letters*, vol. 54, no. 9, pp. 564–566, 2018.
- [264] V. P. Kaffle, Y. Fukushima, and H. Harai, “Internet of things standardization in itu and prospective networking technologies,” *IEEE Communications Magazine*, vol. 54, no. 9, pp. 43–49, 2016.
- [265] Y. Kaga, M. Fujio, K. Naganuma, *et al.*, “A secure and practical signature scheme for blockchain based on biometrics,” in *Information Security Practice and Experience* (Lecture Notes in Computer Science), Lecture Notes in Computer Science. 2017, ch. Chapter 55, pp. 877–891, ISBN: 978-3-319-72358-7 978-3-319-72359-4. DOI: 10.1007/978-3-319-72359-4_55.
- [266] A. Kahveci, “Implications of blockchain on customs and customs’ procedures,” *BLOCKCHAIN IN FINANCE, MARKETING AND OTHERS*, p. 203, 2022.
- [267] S. G. Kanade, D. Petrovska-Delacrétaz, and B. Dorizzi, “Cancelable biometrics for better security and privacy in biometric systems,” in *Advances in Computing and Communications: First International Conference, ACC 2011, Kochi, India, July 22-24, 2011, Proceedings, Part III 1*, Springer, 2011, pp. 20–34.
- [268] T. I. Kang, “Korea pilots blockchain technology as it prepares for the future,” *World Customs Organization news*, 2023. [Online]. Available: <https://mag.wcoomd.org/magazine/wco-news-88/korea-pilots-blockchain-technology-as-it-prepares-for-the-future/>.
- [269] A. K. Kasgar, J. Agrawal, and S. Shahu, “New modified 256-bit md 5 algorithm with sha compression function,” *International Journal of Computer Applications*, vol. 42, no. 12, 2012.
- [270] N. K. G. Kaur and Manjeet, “A robust and secure multitrait based fuzzy extractor,” *8th ICCCNT, July 3 -5, 2017, IIT Delhi*, 2017.
- [271] Y. Kawase and S. Kasahara, “Transaction-confirmation time for bitcoin: A queueing analytical approach to blockchain mechanism,” in *Queueing Theory and Network Applications: 12th International Conference, QTNA 2017, Qinhuangdao, China, August 21-23, 2017, Proceedings 12*, Springer, 2017, pp. 75–88.
- [272] N. Khan and R. State, “Lightning network: A comparative review of transaction fees and data analysis,” in *Blockchain and Applications: International Congress*, Springer, 2020, pp. 11–18.

- [273] S. N. Khan, F. Loukil, C. Ghedira-Guegan, E. Benkhelifa, and A. Bani-Hani, "Blockchain smart contracts: Applications, challenges, and future trends," *Peer-to-peer Networking and Applications*, vol. 14, pp. 2901–2925, 2021.
- [274] Z. A. Khattak, S. Sulaiman, and J.-L. Ab Manan, "A study on threat model for federated identities in federated identity management system," in *2010 International Symposium on Information Technology*, IEEE, vol. 2, 2010, pp. 618–623.
- [275] S. N. Khoshafian and G. P. Copeland, "Object identity," *ACM SIGPLAN Notices*, vol. 21, no. 11, pp. 406–416, 1986.
- [276] M. K. Khoshons, C. C. Lim, and T. Sayed, "Simulation and evaluation of international border crossing clearance systems: A canadian case study," *Transportation research record*, vol. 1966, no. 1, pp. 1–9, 2006.
- [277] A. Kiayias, A. Russell, B. David, and R. Oliynykov, "Ouroboros: A provably secure proof-of-stake blockchain protocol," in *Advances in Cryptology—CRYPTO 2017: 37th Annual International Cryptology Conference, Santa Barbara, CA, USA, August 20–24, 2017, Proceedings, Part I*, Springer, 2017, pp. 357–388.
- [278] T.-h. Kim, C. Ramos, and S. Mohammed, *Smart city and iot*, 2017.
- [279] G. R. Kirti Chawla Christopher McFarland and W. Thomason, "An accurate real-time rfid-based location system," *Int. J. Radio Frequency Identification Technology and Applications, Vol. 5, No. 1, 2018*, pp. 48–76, 2018.
- [280] L. Kleinrock, "Distributed systems," *Communications of the ACM*, vol. 28, no. 11, pp. 1200–1213, 1985, ISSN: 0001-0782 1557-7317. DOI: 10.1145/4547.4552.
- [281] G. Klyne, "Uniform resource identifier (uri) schemes," *IANA Assignments*, 2023. [Online]. Available: <https://www.iana.org/assignments/uri-schemes/uri-schemes.xhtml>.
- [282] B. T. Koca *et al.*, "Bordering practices across europe: The rise of "walls" and "fences"," *Migration Letters*, vol. 16, no. 2, pp. 183–194, 2019.
- [283] G. Kondova and R. Barba, "Governance of decentralized autonomous organizations," *Journal of Modern Accounting and Auditing*, vol. 15, no. 8, pp. 406–411, 2019.
- [284] N. Kratzke and P.-C. Quint, "Understanding cloud-native applications after 10 years of cloud computing - a systematic mapping study," *Journal of Systems and Software*, vol. 126, pp. 1–16, 2017, ISSN: 01641212. DOI: 10.1016/j.jss.2017.01.001.

- [285] K. K. Kulwinder Singh and A. Sardana, "Fingerprint feature extraction," *International Journal of Computer SCIEncE and teChnology*, vol. Vol. 2, ISSue 3, September, pp. 237–241, 2011, ISSN: 0976-8491.
- [286] V. Kumar and A. Bhardwaj, "Identity management systems: A comparative analysis," *International Journal of Strategic Decision Sciences (IJSDS)*, vol. 9, no. 1, pp. 63–78, 2018.
- [287] R. Kundargi J.; Karandikar, "Fingerprint liveness detection using wavelet-based completed lbp descriptor," *In Proceedings of the 2nd International Conference on Computer Vision and Image Processing, Roorkee, India*, vol. 9–12 September 2017; Springer: Berlin, Germany, pp. 187–202. 2018.
- [288] S. Kundra, A. Dureja, and R. Bhatnagar, "The study of recent technologies used in e-passport system," in *2014 IEEE global humanitarian technology conference-South Asia Satellite (GHTC-SAS)*, IEEE, 2014, pp. 141–146.
- [289] R. C. Labong, "Identity theft protection strategies a literature review," *Journal of Academic Research*, vol. 04:2(2019), pp. 1–12, 2019.
- [290] K. Laeeq, "Metaverse: Why, how and what," *How and What*, 2022.
- [291] L. Lamport, "Password authentication with insecure communication," *Communications of the ACM*, vol. 24, no. 11, pp. 770–772, 1981.
- [292] K. A. Lantz, J. L. Edighoffer, and B. L. Hitson, "Towards a universal directory service," *ACM SIGOPS Operating Systems Review*, vol. 20, no. 2, pp. 43–53, 1986.
- [293] A. Laurent, L. Brotcorne, and B. Fortz, "Transaction fees optimization in the ethereum blockchain," *Blockchain: Research and Applications*, vol. 3, no. 3, p. 100 074, 2022.
- [294] M. Laurent and S. Bouzeffrane, *Digital identity management*. Elsevier, 2015.
- [295] J.-H. Lee, "Bidaas: Blockchain based id as a service," *IEEE Access*, vol. 6, pp. 2274–2278, 2018, ISSN: 2169-3536. DOI: 10.1109/access.2017.2782733.
- [296] L. Lee, T. Braud, P. Zhou, *et al.*, "All one needs to know about metaverse: A complete survey on technological singularity, virtual ecosystem, and research agenda.," *arXiv preprint arXiv:2110.05352*, 2021.
- [297] M. Lenco, "Digital identity as a key enabler for e-government services," *Mob. Connect-GSMA*, pp. 1–8, 2016.
- [298] Y. Lewenberg, Y. Sompolinsky, and A. Zohar, "Inclusive block chain protocols," in *Financial Cryptography and Data Security: 19th International Conference, FC 2015, San Juan, Puerto Rico, January 26-30, 2015, Revised Selected Papers 19*, Springer, 2015, pp. 528–547.

- [299] J. Lewis and M. Fowler, "Microservices: A definition of this new architectural term," *MartinFowler.com*, vol. 25, no. 14-26, p. 12, 2014.
- [300] H. Li, K. Ota, and M. Dong, "Learning iot in edge: Deep learning for the internet of things with edge computing," *IEEE Network*, vol. 32, no. 1, pp. 96–101, 2018, ISSN: 0890-8044 1558-156X. DOI: 10.1109/mnet.2018.1700202.
- [301] P. Li, J. Li, Z. Huang, C.-Z. Gao, W.-B. Chen, and K. Chen, "Privacy-preserving outsourced classification in cloud computing," *Cluster Computing*, vol. 21, no. 1, pp. 277–286, 2017, ISSN: 1386-7857 1573-7543. DOI: 10.1007/s10586-017-0849-9.
- [302] S. Li, B. Li, L. Li, and S. Wang, "Research on the application of blockchain technology in the field of cross-border logistics for small and medium-sized enterprises," 2020.
- [303] W. Li and C. J. Mitchell, "Addressing threats to real-world identity management systems," Wiesbaden: Springer Fachmedien Wiesbaden, 2015, pp. 251–259.
- [304] X. Li, Y. Li, T. Liu, J. Qiu, and F. Wang, "The method and tool of cost analysis for cloud computing," in *2009 IEEE International Conference on Cloud Computing*, IEEE, 2009, pp. 93–100.
- [305] Y. Li, W. Susilo, G. Yang, Y. Yu, T. V. X. Phuong, and D. Liu, "Non-equivocation in blockchain: Double-authentication-preventing signatures gone contractual," in *Proceedings of the 2021 ACM Asia Conference on Computer and Communications Security*, New York, NY, USA: Association for Computing Machinery, 2021, pp. 859–871. DOI: 10.1145/3433210.3437516. [Online]. Available: <https://doi.org/10.1145/3433210.3437516>.
- [306] Y. Li, A.-C. Orgerie, I. Rodero, B. L. Amersho, M. Parashar, and J.-M. Menaud, "End-to-end energy models for edge cloud-based iot platforms: Application to data stream analysis in iot," *Future Generation Computer Systems*, vol. 87, pp. 667–678, 2018.
- [307] J. Light, "The differences between a hard fork, a soft fork, and a chain split, and what they mean for the future of bitcoin," *online*. *September*, 2017.
- [308] L. Lin, X. Liao, H. Jin, and P. Li, "Computation offloading toward edge computing," *Proceedings of the IEEE*, vol. 107, no. 8, pp. 1584–1607, 2019, ISSN: 0018-9219 1558-2256. DOI: 10.1109/jproc.2019.2922285.
- [309] X. Lin and P. R. Spence, "Identity on social networks as a cue: Identity, retweets, and credibility," *Communication Studies*, vol. 69, no. 5, pp. 461–482, 2018.

- [310] J. Liu, S. Huo, and Y. Wang, "A hierarchical mapping system for flat identifier to locator resolution based on active degree," *Future Internet*, vol. 10, no. 8, 2018, ISSN: 1999-5903. DOI: 10.3390/fi10080075.
- [311] J. Liu, Y. Deng, T. Bai, Z. Wei, and C. Huang, "Targeting ultimate accuracy: Face recognition via deep embedding," *arXiv preprint arXiv:1506.07310*, 2015.
- [312] M. Liu, K. Wu, and J. J. Xu, "How will blockchain technology impact auditing and accounting: Permissionless versus permissioned blockchain," *Current Issues in auditing*, vol. 13, no. 2, A19–A29, 2019.
- [313] S.-G. Liu, W.-Q. Chen, and J.-L. Liu, "An efficient double parameter elliptic curve digital signature algorithm for blockchain," *IEEE Access*, vol. 9, pp. 77 058–77 066, 2021, ISSN: 2169-3536. DOI: 10.1109/access.2021.3082704.
- [314] Y. Liu, D. He, M. S. Obaidat, N. Kumar, M. K. Khan, and K.-K. R. Choo, "Blockchain-based identity management systems: A review," *Journal of network and computer applications*, vol. 166, p. 102 731, 2020.
- [315] E. Lombrozo, J. Lau, and P. Wuille, "Bip 141: Segregated witness (consensus layer)," *Github.com/Bitcoin*, 2015. [Online]. Available: <https://github.com/bitcoin/bips/blob/master/bip-0141.mediawiki>.
- [316] D. Longley, D. Zagidulin, and M. Sporny, "The did:key method v0.7," 2022. [Online]. Available: <https://w3c-ccg.github.io/did-method-key/>.
- [317] J. C. López-Pimentel, M. Alcaraz-Rivera, R. Granillo-Macias, and E. Olivares-Benitez, "Traceability of mexican avocado supply chain: A microservice and blockchain technological solution," *Sustainability*, vol. 14, no. 21, p. 14 633, 2022.
- [318] N. Lu, B. Wang, Y. Zhang, W. Shi, and C. Esposito, "Neucheck: A more practical ethereum smart contract security analysis tool," *Software: Practice and Experience*, vol. 51, no. 10, pp. 2065–2084, 2019, ISSN: 0038-0644 1097-024X. DOI: 10.1002/spe.2745.
- [319] K. Ludlow, "Genetic identity concerns in the regulation of novel reproductive technologies," *J Law Biosci*, vol. 7, no. 1, lsaa004, 2020, ISSN: 2053-9711 (Print) 2053-9711 (Linking). DOI: 10.1093/jlb/lsaa004. [Online]. Available: <https://www.ncbi.nlm.nih.gov/pubmed/34221417>.
- [320] C. Lundkvist, R. Heck, J. Torstensson, Z. Mitton, and M. Sena, "Uport: A platform for self-sovereign identity," *URL: https://whitepaper.uport.me/u-Port_whitepaper_DRAFT20170221.pdf*, 2017.

- [321] L. Luu, D.-H. Chu, H. Olickel, P. Saxena, and A. Hobor, “Making smart contracts smarter,” in *Proceedings of the 2016 ACM SIGSAC conference on computer and communications security*, 2016, pp. 254–269.
- [322] M. Lux, *European union customs code*, 2016.
- [323] Z. A. Lux, D. Thatmann, S. Zickau, and F. Beierle, “Distributed-ledger-based authentication with decentralized identifiers and verifiable credentials,” in *2020 2nd Conference on Blockchain Research & Applications for Innovative Networks and Services (BRAINS)*, IEEE, 2020, pp. 71–78.
- [324] H. Ma, S. Han, and H. Lei, “Optimized paillier’s cryptosystem with fast encryption and decryption,” in *Annual Computer Security Applications Conference*, 2021, pp. 106–118.
- [325] J. Ma, B. Qi, and K. Lv, “Bsa: Enabling biometric-based storage and authorization on blockchain,” in *2021 IEEE 20th International Conference on Trust, Security and Privacy in Computing and Communications (TrustCom)*, IEEE, 2021, pp. 1077–1084.
- [326] S. Machani, R. Philpott, S. Srinivas, J. Kemp, and J. Hodges, “Fido uaf architectural overview,” *fidoalliance.org*, 2020. [Online]. Available: <https://fidoalliance.org/specs/fido-uaf-v1.2-ps-20201020/fido-uaf-overview-v1.2-ps-20201020.pdf>.
- [327] D. Macrinici, C. Cartoceanu, and S. Gao, “Smart contract applications within blockchain technology: A systematic mapping study,” *Telematics and Informatics*, vol. 35, no. 8, pp. 2337–2354, 2018.
- [328] e. a. Mahesh Adulla (Reddy), “Hyperledger blockchain performance metrics,” *Hyperledger.org*, 2018. [Online]. Available: <https://www.hyperledger.org/learn/publications/blockchain-performance-metrics>.
- [329] S. Maitra, V. P. Yanambaka, A. Abdelgawad, D. Puthal, and K. Yelamarthi, “Proof-of-authentication consensus algorithm: Blockchain-based iot implementation,” in *2020 IEEE 6th World Forum on Internet of Things (WF-IoT)*, IEEE, 2020, pp. 1–2.
- [330] M. A. Manolache, S. Manolache, and N. Tapus, “Decision making using the blockchain proof of authority consensus,” *Procedia Computer Science*, vol. 199, pp. 580–588, 2022.
- [331] F. G. Mármol, J. Girao, and G. M. Pérez, “Trims, a privacy-aware trust and reputation model for identity management systems,” *Computer Networks*, vol. 54, no. 16, pp. 2899–2912, 2010.
- [332] T. Matsuda, K. Takahashi, and G. Hanaoka, “On the security of linear sketch schemes against recovering attacks,” in *ICETE (2)*, 2018, pp. 242–253.

- [333] R. McCabe and E. Newton, “Data format for the interchange of fingerprint, facial, and other biometric information,” *ANSI/NIST*, 2007.
- [334] P. McCorry, S. F. Shahandashti, and F. Hao, “A smart contract for boardroom voting with maximum voter privacy,” in *Financial Cryptography and Data Security: 21st International Conference, FC 2017, Sliema, Malta, April 3-7, 2017, Revised Selected Papers 21*, Springer, 2017, pp. 357–375.
- [335] F. Mehboob, M. Abbas, R. Jiang, A. Rauf, S. A. Khan, and S. Rehman, “Trajectory based vehicle counting and anomalous event visualization in smart cities,” *Cluster Computing*, vol. 21, pp. 443–452, 2018.
- [336] P. Mell, T. Grance, *et al.*, “The nist definition of cloud computing,” 2011.
- [337] R. C. Merkle, “A digital signature based on a conventional encryption function,” in *Advances in Cryptology—CRYPTO’87: Proceedings 7*, Springer, 1988, pp. 369–378.
- [338] P. Mockapetris and K. J. Dunlap, “Development of the domain name system,” in *Symposium proceedings on Communications architectures and protocols*, 1988, pp. 123–133.
- [339] J. G. Mohamed Al Morsy and I. Müller, “An analysis of the cloud computing security problem,” 2010.
- [340] M. L. D. Monache, J. Sprinkle, R. Vasudevan, and D. Work, “Autonomous vehicles from vehicular control to traffic control,” *2019 IEEE 58th Conference on Decision and Control (CDC)*, vol. December 11-13, 2019, 2019.
- [341] T. D. Moshood, G. Nawanir, S. Sorooshian, and O. Okfalisa, “Digital twins driven supply chain visibility within logistics: A new paradigm for future logistics,” *Applied System Innovation*, vol. 4, no. 2, p. 29, 2021.
- [342] K. Munir and L. A. Mohammed, “Biometric smartcard authentication for fog computing,” *International Journal of Network Security Its Applications*, vol. 10, no. 6, pp. 35–45, 2018, ISSN: 09752307 09749330. DOI: 10.5121/ijnsa.2018.10604.
- [343] S. Mystakidis, “Metaverse,” *Encyclopedia*, vol. 2, no. 1, pp. 486–497, 2022. [Online]. Available: <https://www.mdpi.com/2673-8392/2/1/31>.
- [344] K. Naganuma, T. Suzuki, M. Yoshino, K. Takahashi, Y. Kaga, and N. Kunihiro, “New secret key management technology for blockchains from biometrics fuzzy signature,” in *2020 15th Asia Joint Conference on Information Security (AsiaJCIS)*, IEEE, 2020, pp. 54–58.
- [345] A. Nagar, K. Nandakumar, and A. K. Jain, “Biometric template transformation: A security analysis,” in *Media Forensics and Security II*, SPIE, vol. 7541, 2010, pp. 237–251.

- [346] A. Najibi, “Racial discrimination in face recognition technology,” in *BLOG, SCIENCE POLICY, SPECIAL EDITION: SCIENCE POLICY AND SOCIAL JUSTICE*, Harvard University Science in The News, 2020, pp. 1–8.
- [347] S. Nakamoto, “Bitcoin a peer-to-peer electronic cash system,” 2008.
- [348] C. Nakkach, A. Zrelli, and T. Ezzedine, “Smart border surveillance system based on deep learning methods,” in *2022 International Symposium on Networks, Computers and Communications (ISNCC)*, IEEE, 2022, pp. 1–6.
- [349] K. Nandakumar, A. K. Jain, and S. Pankanti, “Fingerprint-based fuzzy vault: Implementation and performance,” *IEEE Transactions on Information Forensics and Security*, vol. 2, no. 4, pp. 744–757, 2007, ISSN: 1556-6013. DOI: 10.1109/tifs.2007.908165.
- [350] B. C. ord Neuman, “Scale in distributed systems,” *ISI/USC*, p. 68, 1994.
- [351] S. Newman, *Building microservices*. ” O’Reilly Media, Inc.”, 2021.
- [352] G.-T. Nguyen and K. Kim, “A survey about consensus algorithms used in blockchain,” *Journal of Information processing systems*, vol. 14, no. 1, pp. 101–128, 2018.
- [353] K. Nguyen, C. Fookes, S. Sridharan, M. Tistarelli, and M. Nixon, “Super-resolution for biometrics: A comprehensive survey,” *Pattern Recognition*, vol. 78, pp. 23–42, 2018.
- [354] M. S. Nixon, P. L. Correia, K. Nasrollahi, T. B. Moeslund, A. Hadid, and M. Tistarelli, “On soft biometrics,” *Pattern Recognition Letters*, vol. 68, pp. 218–230, 2015.
- [355] M. Nofer, P. Gomber, O. Hinz, and D. Schiereck, “Blockchain,” *Business Information Systems Engineering*, vol. 59, no. 3, pp. 183–187, 2017, ISSN: 2363-7005 1867-0202. DOI: 10.1007/s12599-017-0467-3.
- [356] Notebookcheck.com, “Intel core i5-1135g7,” [Online]. Available: <https://www.notebookcheck.net/Intel-Core-i5-1135G7-Processor-Benchmarks-and-Specs.467754.0.html>.
- [357] Notebookcheck.com, “Qualcomm snapdragon 8 gen 2,” [Online]. Available: <https://www.notebookcheck.net/Qualcomm-Snapdragon-8-Gen-2-Processor-Benchmarks-and-Specs.670032.0.html>.
- [358] R. Oak, “A literature survey on authentication using behavioural biometric techniques,” in *Intelligent Computing and Information and Communication: Proceedings of 2nd International Conference, ICICC 2017*, Springer, 2018, pp. 173–181.

- [359] H. S. de Ocáriz Borde, “An overview of trees in blockchain technology: Merkle trees and merkle patricia tries,” 2022.
- [360] A. Ometov, O. L. Molua, M. Komarov, and J. Nurmi, “A survey of security in cloud, edge, and fog computing,” *Sensors*, vol. 22, no. 3, p. 927, 2022.
- [361] openid.net, “Welcome to openid connect,” 2023. [Online]. Available: <https://openid.net/connect/>.
- [362] o. developer.bitcoin.org, “Bitcoin developer guides: Contracts,” *Bitcoin.org*, 2023. [Online]. Available: <https://developer.bitcoin.org/devguide/contracts.html>.
- [363] R. T. Oscar Delgado-Mohatar Julian Fierrez and R. Vera-Rodriguez, “Biometric template storage with blockchain a first look into cost and performance tradeoffs,” *CVPR*, 2017.
- [364] R. T. Oscar Delgado-Mohatar Julian Fierrez and R. Vera-Rodriguez, “Blockchain and biometrics a first look into opportunities and challenges,” 2019, ISSN: arXiv:1903.05496v1 [cs.CR].
- [365] J. Owen, M. Shephard, and A. Stojanovic, “Implementing brexit: Customs,” *Institute for UK Government*, 2017. [Online]. Available: www.instituteforgovernment.org.uk/brexit.
- [366] O. Ozdenizci, Y. Wang, T. Koike-Akino, and D. Erdogmus, “Adversarial deep learning in eeg biometrics,” *IEEE Signal Process Lett*, vol. 26, no. 5, pp. 710–714, 2019, ISSN: 1070-9908 (Print) 1070-9908 (Linking). DOI: 10.1109/LSP.2019.2906826. [Online]. Available: <https://www.ncbi.nlm.nih.gov/pubmed/31814690>.
- [367] J. Pan and J. McElhannon, “Future edge cloud and edge computing for internet of things applications,” *IEEE Internet of Things Journal*, vol. 5, no. 1, pp. 439–449, 2018, ISSN: 2327-4662. DOI: 10.1109/jiot.2017.2767608.
- [368] K. C. Panayiotis Christodoulou and andreas andreou, “A decentralized application for logistics using blockchain in real world applications,” *The Cyprus review*, vol. vol. 30:2 Fall 2018, 2018.
- [369] S. Panchamia and D. K. Byrappa, “Passport, visa and immigration management using blockchain,” in *2017 23rd annual International Conference in advanced computing and communications (ADCOM)*, IEEE, 2017, pp. 8–17.
- [370] U. Park, Y. Tong, and A. K. Jain, “Age-invariant face recognition,” *IEEE transactions on pattern analysis and machine intelligence*, vol. 32, no. 5, pp. 947–954, 2010.
- [371] O. M. Parkhi, A. Vedaldi, and A. Zisserman, “Deep face recognition,” 2015.

- [372] V. M. Patel, N. K. Ratha, and R. Chellappa, "Cancelable biometrics: A review," *IEEE signal processing magazine*, vol. 32, no. 5, pp. 54–65, 2015.
- [373] I. S. Patil and Harshawardhan, "Rfid dynamic surveillance approach," *IJCSI International Journal of Computer Science Issues*, vol. Vol. 7, Issue 3, No 7, 2010.
- [374] D. C. Paul Overell, "Augmented bnf for syntax specifications: Abnf," *IETF REC 2234*, 1997. [Online]. Available: <https://datatracker.ietf.org/doc/html/rfc2234>.
- [375] S. Pearson, "Taking account of privacy when designing cloud computing services," in *2009 ICSE Workshop on Software Engineering Challenges of Cloud Computing*, IEEE, 2009, pp. 44–52.
- [376] M. Pease, R. Shostak, and L. Lamport, "Reaching agreement in the presence of faults," *J. ACM*, vol. 27, no. 2, pp. 228–234, Apr. 1980, ISSN: 0004-5411. DOI: 10.1145/322186.322188. [Online]. Available: <https://doi.org/10.1145/322186.322188>.
- [377] A. R. S. Pedro Sousa and J. A. Marques, "Object identifiers and identity a naming issue," pp. 127–129, 1995. DOI: 1063-5351/95\04.0001995IEEE.
- [378] R. Petke and I. King, "Registration procedures for url scheme names," Tech. Rep., 1999.
- [379] A. Petrosyan, "Number of internet and social media users worldwide as of january 2023," *statista.com*, 2023. [Online]. Available: <https://www.statista.com/statistics/617136/digital-population-worldwide/>.
- [380] K. Pfeuffer, M. J. Geiger, S. Prange, L. Mecke, D. Buschek, and F. Alt, "Behavioural biometrics in vr: Identifying people from body motion and relations in virtual reality," in *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems*, 2019, pp. 1–12.
- [381] A. Pfitzmann and K. Borcea-Pfitzmann, "Lifelong privacy: Privacy and identity management for life," in *Privacy and Identity Management for Life: 5th IFIP WG 9.2, 9.6/11.4, 11.6, 11.7/PrimeLife International Summer School, Nice, France, September 7-11, 2009, Revised Selected Papers 5*, Springer, 2010, pp. 1–17.
- [382] C. Picus, D. Bauer, M. Hubner, *et al.*, "Novel smart sensor technology platform for border crossing surveillance within foldout," *J. Def. Secur. Technol*, vol. 5, pp. 44–57, 2022.
- [383] P. Porwik, "The biometric passport: The technical requirements and possibilities of using," in *2009 International Conference on Biometrics and Kansei Engineering*, IEEE, 2009, pp. 65–69.

- [384] R. Pradeep and N. Sunitha, "A reliable block-chain based biometric authentication solution for aadhar," *Indian Journal of Science and Technology*, vol. 15, no. 41, pp. 2115–2120, 2022.
- [385] G. Premsankar, M. Di Francesco, and T. Taleb, "Edge computing for the internet of things: A case study," *IEEE Internet of Things Journal*, vol. 5, no. 2, pp. 1275–1284, 2018, ISSN: 2327-4662 2372-2541. DOI: 10.1109/jiot.2018.2805263.
- [386] F. Pub, "Standards for security categorization of federal information and information systems," *NIST FIPS*, vol. 199, 2004.
- [387] D. Puthal and S. P. Mohanty, "Proof of authentication: Iot-friendly blockchains," *IEEE Potentials*, vol. 38, no. 1, pp. 26–29, 2018.
- [388] D. Puthal, S. P. Mohanty, P. Nanda, E. Kougianos, and G. Das, "Proof-of-authentication for scalable blockchain in resource-constrained distributed systems," in *2019 IEEE International Conference on Consumer Electronics (ICCE)*, IEEE, 2019, pp. 1–5.
- [389] PwC, "Technical study on smart borders: Final report," 2014. [Online]. Available: https://home-affairs.ec.europa.eu/system/files/2020-09/smart_borders_technical_study_en.pdf.
- [390] Y. A. Qadri, A. Nauman, Y. B. Zikria, A. V. Vasilakos, and S. W. Kim, "The future of healthcare internet of things: A survey of emerging technologies," *IEEE Communications Surveys Tutorials*, vol. 22, no. 2, pp. 1121–1167, 2020, ISSN: 1553-877X 2373-745X. DOI: 10.1109/comst.2020.2973314.
- [391] K. N. Qureshi, S. Din, G. Jeon, and F. Piccialli, "Internet of vehicles: Key technologies, network model, solutions and challenges with future aspects," *IEEE Transactions on Intelligent Transportation Systems*, vol. 22, no. 3, pp. 1777–1786, 2021, ISSN: 1524-9050 1558-0016. DOI: 10.1109/tits.2020.2994972.
- [392] J. Raab, "Powell (1990): Neither market nor hierarchy: Network forms of organization," *Schlüsselwerke der Netzwerkforschung*, pp. 461–463, 2019.
- [393] D. Rachmawati, J. Tarigan, and A. Ginting, "A comparative study of message digest 5 (md5) and sha256 algorithm," in *Journal of Physics: Conference Series*, IOP Publishing, vol. 978, 2018, p. 012116.
- [394] V. Radha and D. H. Reddy, "A survey on single sign-on techniques," *Procedia Technology*, vol. 4, pp. 134–139, 2012.
- [395] R. Ramachandra and C. Busch, "Presentation attack detection methods for face recognition systems: A comprehensive survey," *ACM Computing Surveys (CSUR)*, vol. 50, no. 1, pp. 1–37, 2017.

- [396] R. Ramya and T. Sasikala, “An efficient minkowski distance-based matching with merkle hash tree authentication for biometric recognition in cloud computing,” *Soft Computing*, vol. 23, no. 24, pp. 13 423–13 431, 2019, issn: 1432-7643 1433-7479. DOI: 10.1007/s00500-019-03881-z.
- [397] H. Rantzsch, H. Yang, and C. Meinel, “Signature embedding: Writer independent offline signature verification with deep metric learning,” in *Advances in Visual Computing: 12th International Symposium, ISVC 2016, Las Vegas, NV, USA, December 12-14, 2016, Proceedings, Part II 12*, Springer, 2016, pp. 616–625.
- [398] C. Rathgeb, J. Merkle, J. Scholz, B. Tams, and V. Nesterowicz, “Deep face fuzzy vault: Implementation and performance,” *Computers & Security*, vol. 113, p. 102 539, 2022.
- [399] A. Rattani and R. Derakhshani, “A survey of mobile face biometrics,” *Computers Electrical Engineering*, vol. 72, pp. 39–52, 2018, issn: 00457906. DOI: 10.1016/j.compeleceng.2018.09.005.
- [400] P. P. Ray, “A survey of iot cloud platforms,” *Future Computing and Informatics Journal*, vol. 1, no. 1-2, pp. 35–46, 2016.
- [401] G. P. Reddy, A. Narayana, P. K. Keerthan, B. Vineetha, and P. Honnavalli, “Multiple hashing using sha-256 and md5,” in *Advances in Computing and Network Communications: Proceedings of CoCoNet 2020, Volume 1*, Springer, 2021, pp. 643–655.
- [402] G. D. P. Regulation, “General data protection regulation (gdpr),” *Intersoft Consulting, Accessed in October*, vol. 24, no. 1, 2018.
- [403] S. Reno, S. Bhowmik, and M. Ahmed, “Utilizing ipfs and private blockchain to secure forensic information,” in *2021 International Conference on Automation, Control and Mechatronics for Industry 4.0 (ACMI)*, IEEE, 2021, pp. 1–6.
- [404] M. Reveilhac and M. Pasquet, “Promising secure element alternatives for nfc technology,” in *2009 First International Workshop on Near Field Communication*, IEEE, 2009, pp. 75–80.
- [405] ripple.com, “Digital asset for global economy,” *ripple.com*, 2023. [Online]. Available: <https://ripple.com/xrp/#:~:text=Scalable,1%2C500%20transactions%20per%20second%2C%2024x7..>
- [406] D. M. Ritchie, “The unix system: The evolution of the unix time-sharing system,” *ATT Bell Laboratories Technical Journal*, vol. 63, no. 8, pp. 1577–1593, 1984. DOI: 10.1002/j.1538-7305.1984.tb00054.x.

- [407] D. L. Ron Daniel, “Www names and addresses, uris, urls, urns, urcs,” 1995. [Online]. Available: <https://www.w3.org/Addressing/URL/Addressing.html>.
- [408] A. Roosendaal, “Digital personae and profiles as representations of individuals,” in *Privacy and Identity Management for Life: 5th IFIP WG*, Springer, 2010, pp. 226–236.
- [409] K. Rose, S. Eldridge, and L. Chapin, “The internet of things: An overview,” *The internet society (ISOC)*, vol. 80, pp. 1–50, 2015.
- [410] L. Roseblum, R. A. Cross, W. Earnshaw, and H. Vince, “The challenge of virtual reality,” *Visualization & modeling*, pp. 325–399, 1997.
- [411] A. Ross, “Some research problems in biometrics the future beckons,” 2019, ISSN: 978-1-7281-3640-0/19/1.00 c2019 IEEE.
- [412] A. Ross, A. Jain, and J. Reisman, “A hybrid fingerprint matcher,” *Pattern Recognition*, vol. 36, no. 7, pp. 1661–1673, 2003.
- [413] A. Ross, J. Shah, and A. K. Jain, “From template to image: Reconstructing fingerprints from minutiae points,” *IEEE transactions on pattern analysis and machine intelligence*, vol. 29, no. 4, pp. 544–560, 2007.
- [414] E. M. Rudd, M. Günther, and T. E. Boulton, “Moon: A mixed objective optimization network for the recognition of facial attributes,” in *Computer Vision—ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11–14, 2016, Proceedings, Part V 14*, Springer, 2016, pp. 19–35.
- [415] S. Ruoti, J. Andersen, D. Zappala, and K. Seamons, “Why johnny still, still can’t encrypt: Evaluating the usability of a modern pgp client,” *arXiv preprint arXiv:1510.08555*, 2015.
- [416] F. Sabena, A. Dehghantanha, and A. P. Seddon, “A review of vulnerabilities in identity management using biometrics,” in *2010 Second International Conference on Future Networks*, IEEE, 2010, pp. 42–49.
- [417] M. Sabt, M. Achemlal, and A. Bouabdallah, “Trusted execution environment: What it is, and what it is not,” in *2015 IEEE Trustcom/BigDataSE/IsPa*, IEEE, vol. 1, 2015, pp. 57–64.
- [418] M. Sahasrabudhe and A. M. Namboodiri, “Learning fingerprint orientation fields using continuous restricted boltzmann machines,” in *2013 2nd IAPR Asian Conference on Pattern Recognition*, IEEE, 2013, pp. 351–355.
- [419] N. Sakimura, J. Bradley, M. Jones, B. de Medeiros, and C. Mortimore, “Openid connect core 1.0,” 2014. [Online]. Available: https://openid.net/specs/openid-connect-core-1_0-final.html.

- [420] Y. U. Sakurai and Kouichi, “A proposal of one-time biometric authentication,”
- [421] R. Salz, “A universally unique identifier (uuid) urn namespace,” 2005.
- [422] S. S. I. Samuel, “A review of connectivity challenges in iot-smart home,” in *2016 3rd MEC International conference on big data and smart city (ICBDSC)*, IEEE, 2016, pp. 1–4.
- [423] K. Sandrasegaran and M. Li, “Identity management,” in *Handbook of Research on Wireless Security*, IGI Global, 2008, pp. 44–60.
- [424] D. P.-D. Sanjay Kanade and B. Dorizzi, “Cancelable iris biometrics and using error correcting codes to reduce variability in biometric data,” 2009, ISSN: 978-1-4244-3991-1/09/\$25.00 ©2009 IEEE.
- [425] S. C. Santamaria, “Cadena, a blockchain enabled solution for the implementation of mutual recognition arrangements/agreements,” *World Customs Organization news*, 2023. [Online]. Available: <https://mag.wcoomd.org/magazine/wco-news-87/cadena-a-blockchain-enabled-solution-for-the-implementation-of-mutual-recognition-arrangements-agreements/>.
- [426] A. Santander and A. One, *My first aragon app: Voting supercharged with daostack’s holographic consensus (part 4)*, 2019.
- [427] F. K. Santoso and N. C. Vun, “Securing iot for smart home system,” in *2015 international symposium on consumer electronics (ISCE)*, IEEE, 2015, pp. 1–2.
- [428] A. Sarkar and B. K. Singh, “A review on performance, security and various biometric template protection schemes for biometric authentication systems,” *Multimedia Tools and Applications*, vol. 79, no. 37-38, pp. 27 721–27 776, 2020, ISSN: 1380-7501 1573-7721. DOI: 10.1007/s11042-020-09197-7.
- [429] M. Satyanarayanan, “The emergence of edge computing,” *IEEE COMPUTER SOCIETY*, 2017, ISSN: 0018 - 9 1 62 / 1 7 / \$ 3 3.00 © 2 0 1 7 I E E E.
- [430] G. Sawant and V. Bharadi, “Permission blockchain based smart contract utilizing biometric authentication as a service: A future trend,” in *2020 International Conference on Convergence to Digital World - Quo Vadis (ICCDW)*, 2020, pp. 1–4. DOI: 10.1109/ICCDW45521.2020.9318715.
- [431] S. Sayeed and H. Marco-Gisbert, “Assessing blockchain consensus and security mechanisms against the 51% attack,” *Applied sciences*, vol. 9, no. 9, p. 1788, 2019.
- [432] S. Sayeed, H. Marco-Gisbert, and T. Caira, “Smart contract: Attacks and protections,” *IEEE Access*, vol. 8, pp. 24 416–24 427, 2020, ISSN: 2169-3536. DOI: 10.1109/access.2020.2970495.

- [433] C.-P. Schnorr, “Efficient identification and signatures for smart cards,” in *Advances in Cryptology—CRYPTO’89 Proceedings 9*, Springer, 1990, pp. 239–252.
- [434] S. A. Schuckers, “Spoofing and anti-spoofing measures,” *Information Security technical report*, vol. 7, no. 4, pp. 56–62, 2002.
- [435] T. B. Secretariat, “Standard on identity and credential assurance,” DOI: <https://www.tbs-sct.gc.ca/pol/doceng.aspx>, 2013.
- [436] J. Sedlmeir, R. Smethurst, A. Rieger, and G. Fridgen, “Digital identities and verifiable credentials,” *Business Information Systems Engineering*, vol. 63, no. 5, pp. 603–613, 2021, ISSN: 2363-7005 1867-0202. DOI: 10.1007/s12599-021-00722-y.
- [437] J. Sedlmeir, R. Smethurst, A. Rieger, and G. Fridgen, “Digital identities and verifiable credentials,” *Business & Information Systems Engineering*, vol. 63, no. 5, pp. 603–613, 2021.
- [438] P. L. Seijas, S. Thompson, and D. McAdams, “Scripting smart contracts for distributed ledger technology,” *Cryptology ePrint Archive*, 2016.
- [439] M. Shahriar Rahman, A. Al Omar, M. Z. A. Bhuiyan, A. Basu, S. Kiyomoto, and G. Wang, “Accountable cross-border data sharing using blockchain under relaxed trust assumption,” *IEEE Transactions on Engineering Management*, vol. 67, no. 4, pp. 1476–1486, 2020. DOI: 10.1109/TEM.2019.2960829.
- [440] A. Shamir, “How to share a secret,” *Communications of the ACM*, vol. 22, no. 11, pp. 612–613, 1979.
- [441] A. Sharif, R. Kumar, J. Ouyang, *et al.*, “Making assembly line in supply chain robust and secure using uhf rfid,” *Sci Rep*, vol. 11, no. 1, p. 18041, 2021, ISSN: 2045-2322 (Electronic) 2045-2322 (Linking). DOI: 10.1038/s41598-021-97598-5. [Online]. Available: <https://www.ncbi.nlm.nih.gov/pubmed/34508125>.
- [442] Y. Sheffer, R. Holz, and P. Saint-Andre, “Recommendations for secure use of transport layer security (tls) and datagram transport layer security (dtls),” Tech. Rep., 2015.
- [443] J. Shen, T. Zhou, X. Chen, J. Li, and W. Susilo, “Anonymous and traceable group data sharing in cloud computing,” *IEEE Transactions on Information Forensics and Security*, vol. 13, no. 4, pp. 912–925, 2018, ISSN: 1556-6013 1556-6021. DOI: 10.1109/tifs.2017.2774439.
- [444] H. Sheth and J. Dattani, “Overview of blockchain technology,” *Asian Journal For Convergence In Technology (AJCT) ISSN-2350-1146*, 2019.

- [445] R. Shirey, "Internet security glossary, version 2. request for comments," RFC 4949. <https://tools.ietf.org/html/rfc4949>. Last access 7 July, Tech. Rep., 2019.
- [446] S. ShoCard, *Travel identity of the future—white paper*, 2016.
- [447] W. Si, J. Zhang, Y.-D. Li, *et al.*, "Remote identity verification using gait analysis and face recognition," *Wireless Communications and Mobile Computing*, vol. 2020, pp. 1–10, 2020, ISSN: 1530-8677 1530-8669. DOI: 10.1155/2020/8815461.
- [448] S. M. Siegel and Josh, "Conversations with connected vehicles," *5th International Conference on the Internet of Things (IoT)*, 2015.
- [449] M. M. Silvia Morera and R. Lorini, "Blockchain and digital identity: The path to self sovereign identity," *PwC research paper*, 2019. [Online]. Available: www.pwc.com/it.
- [450] M. Singh, R. Singh, and A. Ross, "A comprehensive overview of biometric fusion," *Information Fusion*, vol. 52, pp. 187–205, 2019.
- [451] R. Singh, "An overview of android operating system and its security," *int. journal of Engineering Research and Applications*, vol. 4, no. 2, pp. 519–521, 2014.
- [452] S. Singh, A. S. Hosen, and B. Yoon, "Blockchain security attacks, challenges, and solutions for the future distributed iot network," *IEEE Access*, vol. 9, pp. 13 938–13 959, 2021.
- [453] A. Sivanathan, D. Sherratt, H. H. Gharakheili, *et al.*, "Characterizing and classifying iot traffic in smart cities and campuses," in *2017 IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS)*, IEEE, 2017, pp. 559–564.
- [454] N. Six, N. Herbaut, and C. Salinesi, "Blockchain software patterns for the design of decentralized applications: A systematic literature review," *Blockchain: Research and Applications*, p. 100 061, 2022.
- [455] S. N. G. SNG, "Identity management systems (ims): Identification and comparison study independent centre for privacy protection (icpp)/unabhängiges landeszentrum für datenschutz (uld) schleswig-holstein," 2003.
- [456] S. Solat and M. Potop-Butucaru, "Brief announcement: Zeroblock: Timestamp-free prevention of block-withholding attack in bitcoin," in *Stabilization, Safety, and Security of Distributed Systems*, P. Spirakis and P. Tsigas, Eds., Cham: Springer International Publishing, 2017, pp. 356–360.
- [457] R. Soltani, U. T. Nguyen, and A. An, "A survey of self-sovereign identity ecosystem," *Security and Communication Networks*, vol. 2021, pp. 1–26, 2021.

- [458] Y. Sompolinsky and A. Zohar, “Accelerating bitcoin’s transaction processing. fast money grows on trees, not chains,” *Cryptology ePrint Archive*, 2013.
- [459] Sovrin, “Sovrin-protocol-and-token-white-paper,” *A White Paper from the Sovrin Foundation*, 2018.
- [460] M. Sporny, D. Longley, M. Sabadello, *et al.*, “Decentralized identifiers (dids) v1.0 core architecture, data model, and representations,” 2022. [Online]. Available: <https://www.w3.org/TR/did-core/>.
- [461] O. Steele, M. Sporn, and T. Looker, *Eddsa cryptosuite v2020*, 2022. [Online]. Available: <https://www.w3.org/community/reports/credentials/CG-FINAL-di-eddsa-2020-20220724/>.
- [462] O. Steele and M. Sporny, “Did specification registries,” 2023. [Online]. Available: <https://w3c.github.io/did-spec-registries/>.
- [463] M. van Steen and A. S. Tanenbaum, “A brief introduction to distributed systems,” *Computing*, vol. 98, no. 10, pp. 967–1009, 2016, ISSN: 0010-485X 1436-5057. DOI: 10.1007/s00607-016-0508-7.
- [464] H.-R. Su, K.-Y. Chen, W. J. Wong, and S.-H. Lai, “A deep learning approach towards pore extraction for high-resolution fingerprint recognition,” in *2017 IEEE international conference on acoustics, speech and signal processing (ICASSP)*, IEEE, 2017, pp. 2057–2061.
- [465] Substrate.io, “Blockchain basics,” *docs.substrate.io*, 2023. [Online]. Available: <https://docs.substrate.io/fundamentals/blockchain-basics/>.
- [466] A. J. Sudan and Madhu, “A fuzzy vault scheme,”
- [467] D. Sunaryono, J. Siswantoro, and R. Anggoro, “An android based course attendance system using face recognition,” *Journal of King Saud University - Computer and Information Sciences*, vol. 33, no. 3, pp. 304–312, 2021, ISSN: 1319-1578. DOI: <https://doi.org/10.1016/j.jksuci.2019.01.006>. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S1319157818309406>.
- [468] K. Sundararajan and D. L. Woodard, “Deep learning for biometrics,” *ACM Computing Surveys*, vol. 51, no. 3, pp. 1–34, 2018, ISSN: 0360-0300 1557-7341. DOI: 10.1145/3190618.
- [469] H. Suo, J. Wan, C. Zou, and J. Liu, “Security in the internet of things: A review,” in *2012 international conference on computer science and electronics engineering*, IEEE, vol. 3, 2012, pp. 648–651.
- [470] J. Surowiecki, *The wisdom of crowds*. Anchor, 2005.

- [471] E. Surveys, “Google auth vulnerability,” 2020. [Online]. Available: <https://medium.com/swlh/google-firebase-authentication-vulnerability-245050cb7ceb>.
- [472] M. Svitek, T. Horak, D. Carrera, *et al.*, “Smart logistics across smart border,” in *2021 Smart City Symposium Prague (SCSP)*, IEEE, 2021, pp. 1–5.
- [473] P. Szilágyi, “Eip-225: Clique proof-of-authority consensus protocol,” *Ethereum improvement proposals*, 2017.
- [474] D. Taibi, V. Lenarduzzi, and C. Pahl, “Processes, motivations, and issues for migrating to microservices architectures: An empirical investigation,” *IEEE Cloud Computing*, vol. 4, no. 5, pp. 22–32, 2017.
- [475] D. Taibi, V. Lenarduzzi, and C. Pahl, “Architectural patterns for microservices: A systematic mapping study,” in *CLOSER 2018: Proceedings of the 8th International Conference on Cloud Computing and Services Science; Funchal, Madeira, Portugal, 19-21 March 2018*, SciTePress, 2018.
- [476] Y. Taigman, M. Yang, M. Ranzato, and L. Wolf, “Closing the gap to human-level performance in face verification. deepface,” in *Proceedings of the IEEE Computer Vision and Pattern Recognition (CVPR)*, vol. 5, p. 6.
- [477] e. a. Tamas Blummer, “An introduction to hyperledger,” *Hyperledger.org*, 2018. [Online]. Available: https://www.hyperledger.org/wp-content/uploads/2018/08/HL_Whitepaper_IntroductiontoHyperledger.pdf.
- [478] B. Tan and S. Schuckers, “Liveness detection for fingerprint scanners based on the statistics of wavelet signal processing,” in *2006 Conference on Computer Vision and Pattern Recognition Workshop (CVPRW’06)*, IEEE, 2006, pp. 26–26.
- [479] M. v. S. Tanenbaum and A. S., “Distributed system,” *Maarten van Steen*, vol. Third edition, 2018, ISSN: 978-90-815406-2-9.
- [480] A. Tešanovic, “What is a pattern,” *Dr. ing. course DT8100 Object-oriented Systems*, 2005.
- [481] L. Thomas, Y. Zhou, C. Long, J. Wu, and N. Jenkins, “A general form of smart contract for decentralized energy systems management,” *Nature Energy*, vol. 4, no. 2, pp. 140–149, 2019, ISSN: 2058-7546. DOI: 10.1038/s41560-018-0317-7.
- [482] E. Tijan, S. Aksentijević, K. Ivanić, and M. Jardas, “Blockchain technology implementation in logistics,” *Sustainability*, vol. 11, no. 4, 2019, ISSN: 2071-1050. DOI: 10.3390/su11041185.

- [483] S. Tikhomirov, “Ethereum: State of knowledge and research perspectives,” in *Foundations and Practice of Security: 10th International Symposium, FPS 2017, Nancy, France, October 23-25, 2017, Revised Selected Papers 10*, Springer, 2018, pp. 206–221.
- [484] L. M. M. Tim Berners-Lee Roy T. Fielding, “Uniform resource identifier (uri): Generic syntax,” *IETF REC 1738*, 1994. [Online]. Available: <https://datatracker.ietf.org/doc/html/rfc1738>.
- [485] L. M. M. Tim Berners-Lee Roy T. Fielding, “Uniform resource identifier (uri): Generic syntax,” *IETF REC 2396*, 1998. [Online]. Available: <https://datatracker.ietf.org/doc/html/rfc2396>.
- [486] L. M. M. Tim Berners-Lee Roy T. Fielding, “Uniform resource identifier (uri): Generic syntax,” *IETF REC 3986*, 2005. [Online]. Available: <https://datatracker.ietf.org/doc/html/rfc3986>.
- [487] K. Tinn, “Blockchain and the future of optimal financing contracts,” *SSRN*, 2019.
- [488] R. Tolosana, R. Vera-Rodriguez, J. Fierrez, and J. Ortega-Garcia, “Exploring recurrent neural networks for on-line handwritten signature biometrics,” *Ieee Access*, vol. 6, pp. 5128–5138, 2018.
- [489] R. Tonelli, M. I. Lunesu, A. Pinna, D. Taibi, and M. Marchesi, “Implementing a microservices system with blockchain smart contracts,” in *2019 IEEE international workshop on blockchain oriented software engineering (IWBOSE)*, IEEE, 2019, pp. 22–31.
- [490] J. Torres, M. Nogueira, and G. Pujolle, “A survey on identity management for the future network,” *IEEE Communications Surveys & Tutorials*, vol. 15, no. 2, pp. 787–802, 2012.
- [491] F. Toutara and G. Spathoulas, “A distributed biometric authentication scheme based on blockchain,” in *2020 IEEE International Conference on Blockchain (Blockchain)*, IEEE, 2020, pp. 470–475.
- [492] S. Troncoso and A. M. Utratel, “If i only had a heart: A disco manifesto,” *The Transnational Institute*, vol. 14, 2019.
- [493] J. Turow, L. Feldman, and K. Meltzer, “Open to exploitation: America’s shoppers online and offline,” *Departmental Papers (ASC)*, p. 35, 2005.
- [494] S. Turrina, E. Bortoletto, G. Giannini, and D. De Leo, “Monozygotic twins: Identical or distinguishable for science and law?” *Med Sci Law*, vol. 61, no. 1_{suppl}, pp. 62–66, 2021, ISSN: 2042-1818 (Electronic) 0025-8024 (Linking). DOI: 10.1177/0025802420922335. [Online]. Available: <https://www.ncbi.nlm.nih.gov/pubmed/33591870>.

- [495] F. Turrone, "Fingerprint recognition: Enhancement, feature extraction and automatic evaluation of algorithms," *PhD Thesis : Universit'a di Bologna*, 2012.
- [496] S. G. Tzafestas, "Synergy of iot and ai in modern society: The robotics and automation case," *Robot. Autom. Eng. J*, vol. 31, pp. 1–15, 2018.
- [497] R. Van Mólken, *Blockchain across Oracle: Understand the details and implications of the Blockchain for Oracle developers and customers*. Packt Publishing Ltd, 2018.
- [498] A. D. Vanitha Carmel, "A survey on biometric authentication systems in cloud to combat identity theft," *Journal of critical reviews*, vol. 7, no. 03, 2020, ISSN: 23945125. DOI: 10.31838/jcr.07.03.97.
- [499] A. A. Vazirani, O. O'Donoghue, D. Brindley, and E. Meinert, "Blockchain vehicles for efficient medical record management," *NPJ Digit Med*, vol. 3, p. 1, 2020, ISSN: 2398-6352 (Electronic) 2398-6352 (Linking). DOI: 10.1038/s41746-019-0211-0. [Online]. Available: <https://www.ncbi.nlm.nih.gov/pubmed/31934645>.
- [500] Veridocglobal, "Veridocglobal: Finished blockchain solution," 2023. [Online]. Available: <https://veridocglobal.com/>.
- [501] G. S. Veronese, M. Correia, A. N. Bessani, L. C. Lung, and P. Verissimo, "Efficient byzantine fault-tolerance," *IEEE Transactions on Computers*, vol. 62, no. 1, pp. 16–30, 2013, ISSN: 0018-9340. DOI: 10.1109/tc.2011.221.
- [502] e. a. Vipin Bharathan, "Hyperledger architecture, volume i: Introduction to hyperledger business blockchain design philosophy and consensus," *Hyperledger.org*, 2018. [Online]. Available: https://www.hyperledger.org/wp-content/uploads/2017/08/Hyperledger_Arch_WG_Paper_1_Consensus.pdf.
- [503] e. a. Vipin Bharathan, "Hyperledger architecture, volume ii: Smart contracts," *Hyperledger.org*, 2018. [Online]. Available: https://www.hyperledger.org/wp-content/uploads/2018/04/Hyperledger_Arch_WG_Paper_2_SmartContracts.pdf.
- [504] U. Visas and Immigration, "Retention and usage of biometric information (accessible)," *Gov.uk*, 2023. [Online]. Available: <https://www.gov.uk/government/publications/biometric-information/retention-and-usage-of-biometric-information-accessible>.

- [505] E. Viveash, “Hewlett packard enterprise (hpe) has built the largest single-memory computing system in the world at 160tb of ram,” 2017. [Online]. Available: [https://news.filehippo.com/2017/05/hpe-unveils-worlds-largest-single-memory-computer/#:~:text=TB\)...,Hewlett%5C%20Packard%5C%20Enterprise%5C%20\(HPE\)%5C%20has%5C%20built%5C%20the%5C%20largest%5C%20single%5C%20,80%5C%2C000%5C%20of%5C%20the%5C%20latest%5C%20iPhones.](https://news.filehippo.com/2017/05/hpe-unveils-worlds-largest-single-memory-computer/#:~:text=TB)...,Hewlett%5C%20Packard%5C%20Enterprise%5C%20(HPE)%5C%20has%5C%20built%5C%20the%5C%20largest%5C%20single%5C%20,80%5C%2C000%5C%20of%5C%20the%5C%20latest%5C%20iPhones.)
- [506] A. Volchkov, “Revisiting single sign-on: A pragmatic approach in a new context,” *IT Professional*, vol. 3, no. 1, pp. 39–45, 2001.
- [507] D. Vujičić, D. Jagodić, and S. Ranić, “Blockchain technology, bitcoin, and ethereum: A brief overview,” in *2018 17th international symposium infoteh-jahorina (infoteh)*, IEEE, 2018, pp. 1–6.
- [508] P. Wackerow, “Introduction to smart contracts,” *Ethereum.org*, 2022. [Online]. Available: <https://ethereum.org/en/developers/docs/smart-contracts/>.
- [509] C. Wang, W. Duan, J. Ma, and C. Wang, “The research of android system architecture and application programming,” in *Proceedings of 2011 International Conference on Computer Science and Network Technology*, IEEE, vol. 2, 2011, pp. 785–790.
- [510] P. Wang, L. You, G. Hu, L. Hu, Z. Jian, and C. Xing, “Biometric key generation based on generated intervals and two-layer error correcting technique,” *Pattern Recognition*, vol. 111, 2021, ISSN: 00313203. DOI: 10.1016/j.patcog.2020.107733.
- [511] Q. Wang, B. Qin, J. Hu, and F. Xiao, “Preserving transaction privacy in bitcoin,” *Future Generation Computer Systems*, vol. 107, pp. 793–804, 2020, ISSN: 0167739X. DOI: 10.1016/j.future.2017.08.026.
- [512] R. Wang, Z. Lin, and H. Luo, “Blockchain, bank credit and sme financing,” *Quality Quantity*, vol. 53, no. 3, pp. 1127–1140, 2018, ISSN: 0033-5177 1573-7845. DOI: 10.1007/s11135-018-0806-6.
- [513] R. Wang, C. Han, and T. Guo, “A novel fingerprint classification method based on deep learning,” in *2016 23rd International Conference on Pattern Recognition (ICPR)*, IEEE, 2016, pp. 931–936.
- [514] S. Wang, W. Ding, J. Li, Y. Yuan, L. Ouyang, and F.-Y. Wang, “Decentralized autonomous organizations: Concept, model, and applications,” *IEEE Transactions on Computational Social Systems*, vol. 6, no. 5, pp. 870–878, 2019.

- [515] S. Wang, L. Ouyang, Y. Yuan, X. Ni, X. Han, and F.-Y. Wang, “Blockchain-enabled smart contracts: Architecture, applications, and future trends,” *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 49, no. 11, pp. 2266–2277, 2019, ISSN: 2168-2216 2168-2232. DOI: 10.1109/tsmc.2019.2895123.
- [516] S. Wang and W. Wang, “A review of the application of digital identity in the metaverse,” 2023.
- [517] Y. Wang, Z. Su, N. Zhang, *et al.*, “A survey on metaverse: Fundamentals, security, and privacy,” *IEEE Communications Surveys & Tutorials*, 2022.
- [518] Z. Wang, H. Jin, W. Dai, K.-K. R. Choo, and D. Zou, “Ethereum smart contract security research: Survey and future research opportunities,” *Frontiers of Computer Science*, vol. 15, no. 2, 2020, ISSN: 2095-22282095-2236. DOI: 10.1007/s11704-020-9284-9.
- [519] Z. Wang, J. Yang, and Y. Zhu, “Review of ear biometrics,” *Archives of Computational Methods in Engineering*, vol. 28, no. 1, pp. 149–180, 2019, ISSN: 1134-3060 1886-1784. DOI: 10.1007/s11831-019-09376-2.
- [520] J. L. Wayman, “Biometrics in identity management systems,” *IEEE Security & Privacy*, vol. 6, no. 2, pp. 30–37, 2008.
- [521] N. Webb, “A fork in the blockchain: Income tax and the bitcoin/bitcoin cash hard fork,” *North Carolina Journal of Law & Technology*, vol. 19, no. 4, p. 283, 2018.
- [522] S. G. Weber, L. Martucci, S. Ries, and M. Mühlhäuser, “Towards trustworthy identity and access management for the future internet,” in *4th International Workshop on Trustworthy Internet of People, Things & Services*, 2010.
- [523] A. F. Westin, “Privacy and freedom,” *Washington and Lee Law Review*, vol. 25, no. 1, p. 166, 1968.
- [524] wikipedia, “Uniform resource identifier,” 2020. [Online]. Available: https://en.wikipedia.org/wiki/Uniform_Resource_Identifier.
- [525] wikipedia.org, “On the internet, nobody knows you’re a dog.” [Online]. Available: https://en.wikipedia.org/wiki/On_the_Internet,_nobody_knows_you're_a_dog.
- [526] T. Wolf, M. Babaei, and G. Rigoll, “Multi-view gait recognition using 3d convolutional neural networks,” in *2016 IEEE international conference on image processing (ICIP)*, IEEE, 2016, pp. 4165–4169.
- [527] A. Wright, “The rise of decentralized autonomous organizations: Opportunities and challenges,” *Stanford Journal of Blockchain Law & Policy*, vol. 4, no. 2, pp. 152–176, 2021.

- [528] D. C. S. Wright, “Turing complete bitcoin script white paper,” *Available at SSRN 3160279*, 2016.
- [529] L. Wu, J. Yang, M. Zhou, Y. Chen, and Q. Wang, “Lvid: A multimodal biometrics authentication system on smartphones,” *IEEE Transactions on Information Forensics and Security*, vol. 15, pp. 1572–1585, 2020, ISSN: 1556-6013 1556-6021. DOI: 10.1109/tifs.2019.2944058.
- [530] J. Xiaomeng, Z. Fan, L. Shenwen, Y. Jinglin, and H. Ketai, “Data analysis of bitcoin blockchain network nodes,” in *2020 15th IEEE Conference on Industrial Electronics and Applications (ICIEA)*, 2020, pp. 1891–1895. DOI: 10.1109/ICIEA48937.2020.9248092.
- [531] J. Xie and C.-C. Liu, “Multi-agent systems and their applications,” *Journal of International Council on Electrical Engineering*, vol. 7, no. 1, pp. 188–197, 2017, ISSN: 2234-8972. DOI: 10.1080/22348972.2017.1348890.
- [532] J. Xiong and Q. Wang, “Anonymous auction protocol based on time-released encryption atop consortium blockchain,” *International Journal of Advanced Information Technology*, vol. 09, no. 01, pp. 01–16, 2019, ISSN: 22311920 22311548. DOI: 10.5121/ijait.2019.9101.
- [533] J. Xu, K. Xue, S. Li, *et al.*, “Healthchain: A blockchain-based privacy preserving scheme for large-scale health data,” *IEEE Internet of Things Journal*, vol. 6, no. 5, pp. 8770–8781, 2019, ISSN: 2327-4662 2372-2541. DOI: 10.1109/jiot.2019.2923525.
- [534] L. L. Xu and Budong, “Research on information security technology based on blockchain,” *2018 the 3rd IEEE International Conference on Cloud Computing and Big Data Analysis*, 2018, ISSN: 978-1-5386-4301-3/18/\$31.00 ©2018 IEEE.
- [535] X. Xu, I. Weber, M. Staples, *et al.*, “A taxonomy of blockchain-based systems for architecture design,” in *2017 IEEE international conference on software architecture (ICSA)*, IEEE, 2017, pp. 243–252.
- [536] Yahoo! “Bitcoin historical data.” (2023).
- [537] R. Yang, F. R. Yu, P. Si, Z. Yang, and Y. Zhang, “Integrated blockchain and edge computing systems: A survey, some research issues and challenges,” *IEEE Communications Surveys Tutorials*, vol. 21, no. 2, pp. 1508–1532, 2019. DOI: 10.1109/COMST.2019.2894727.
- [538] W. Yang, S. Wang, H. Cui, Z. Tang, and Y. Li, “A review of homomorphic encryption for privacy-preserving biometrics,” *Sensors*, vol. 23, no. 7, p. 3566, 2023.

- [539] W. Yang, S. Wang, J. Hu, G. Zheng, and C. Valli, "Security and accuracy of fingerprint-based biometrics: A review," *Symmetry*, vol. 11, no. 2, 2019, ISSN: 2073-8994. DOI: 10.3390/sym11020141.
- [540] W. Yang, S. Wang, J. Hu, G. Zheng, J. Yang, and C. Valli, "Securing deep learning based edge finger vein biometrics with binary decision diagram," *IEEE Transactions on Industrial Informatics*, vol. 15, no. 7, pp. 4244–4253, 2019, ISSN: 1551-3203 1941-0050. DOI: 10.1109/tii.2019.2900665.
- [541] L. R. Yevgeniy Dodis and A. Smith, "Fuzzy extractors how to generate strong keys from biometric and noisy data," *EUROCRYPT*, vol. C. Cachin and J. Camenisch, pp. 523–540, 2004.
- [542] H. Yousaf, G. Kappos, and S. Meiklejohn, "Tracing transactions across cryptocurrency ledgers," in *28th USENIX Security Symposium (USENIX Security 19)*, 2019, pp. 837–850.
- [543] G. Yuen, "How blockchain-based multi-biometrics revolutionizes kyc for cryptocurrency and fintech," *Digital Transaction Limited*, 2021.
- [544] N. Zhang, A. Rector, I. Buchan, *et al.*, "A linkable identity privacy algorithm for healthgrid," *Studies in Health Technology and Informatics*, vol. 112, pp. 234–246, 2005.
- [545] X. Zhang, D. Cheng, P. Jia, Y. Dai, and X. Xu, "An efficient android-based multimodal biometric authentication system with face and voice," *IEEE Access*, vol. 8, pp. 102 757–102 772, 2020, ISSN: 2169-3536. DOI: 10.1109/access.2020.2999115.
- [546] J. Zhao, H. Xu, H. Liu, J. Wu, Y. Zheng, and D. Wu, "Detection and tracking of pedestrians and vehicles using roadside lidar sensors," *Transportation Research Part C: Emerging Technologies*, vol. 100, pp. 68–87, 2019, ISSN: 0968090X. DOI: 10.1016/j.trc.2019.01.007.
- [547] Q. Zhao, D. Zhang, L. Zhang, and N. Luo, "Adaptive fingerprint pore modeling and extraction," *Pattern Recognition*, vol. 43, no. 8, pp. 2833–2844, 2010, ISSN: 00313203. DOI: 10.1016/j.patcog.2010.02.016.
- [548] Q. Zhao, D. Zhang, L. Zhang, and N. Luo, "Adaptive fingerprint pore modeling and extraction," *Pattern Recognition*, vol. 43, no. 8, pp. 2833–2844, 2010.
- [549] W. Zhao, S. Jin, and W. Yue, "Analysis of the average confirmation time of transactions in a blockchain system," in *Queueing Theory and Network Applications: 14th International Conference, QTNA 2019, Ghent, Belgium, August 27–29, 2019, Proceedings*, Springer, 2019, pp. 379–388.

- [550] X. Zhao, F. Pu, Z. Wang, H. Chen, and Z. Xu, "Detection, tracking, and geolocation of moving vehicle from uav using monocular camera," *IEEE Access*, vol. 7, pp. 101 160–101 170, 2019, ISSN: 2169-3536. DOI: 10.1109/access.2019.2929760.
- [551] Y. Zhao, J. Zhao, L. Jiang, *et al.*, "Privacy-preserving blockchain-based federated learning for iot devices," *IEEE Internet of Things Journal*, vol. 8, no. 3, pp. 1817–1829, 2021, ISSN: 2327-4662 2372-2541. DOI: 10.1109/jiot.2020.3017377.
- [552] Z. Zheng, S. Xie, H. Dai, X. Chen, and H. Wang, "An overview of blockchain technology: Architecture, consensus, and future trends," in *2017 IEEE International Congress on Big Data (BigData Congress)*, Los Alamitos, CA, USA: IEEE Computer Society, Jun. 2017, pp. 557–564. DOI: 10.1109/BigDataCongress.2017.85.
- [553] Z. Zheng, S. Xie, H. Dai, X. Chen, and H. Wang, "An overview of blockchain technology: Architecture, consensus, and future trends," in *2017 IEEE international congress on big data (BigData congress)*, Ieee, 2017, pp. 557–564.
- [554] Q. Zhou, H. Huang, Z. Zheng, and J. Bian, "Solutions to scalability of blockchain: A survey," *Ieee Access*, vol. 8, pp. 16 440–16 455, 2020.
- [555] Ziggy, "This is why heathrow's immigration e-gates can be a joke," *Traveling-formiless.com*, 2020. [Online]. Available: <https://travelingformiles.com/this-is-why-heathrows-immigration-e-gates-can-be-a-joke/>.
- [556] R. Ziolkowski, G. Miscione, and G. Schwabe, "Exploring decentralized autonomous organizations: Towards shared interests and 'code is constitution'," 2020.
- [557] D. Zissis and D. Lekkas, "Addressing cloud computing security issues," *Future Generation computer systems*, vol. 28, no. 3, pp. 583–592, 2012.
- [558] N. Zivic, C. Ruland, and O. Ur-Rehman, "Addressing byzantine fault tolerance in blockchain technology," in *2019 8th International Conference on Modeling Simulation and Applied Optimization (ICMSAO)*, IEEE, 2019, pp. 1–5.