Zero-Knowledge Proof-of-Identity:

Sybil-Resistant, Anonymous Authentication on Permissionless Blockchains and Incentive Compatible, Strictly Dominant Cryptocurrencies

David Cerezo Sánchez
david@calctopia.com

September 16, 2024

Abstract

Zero-Knowledge Proof-of-Identity from trusted public certificates (e.g., national identity cards and/or ePassports; eSIM) is introduced here to permissionless blockchains in order to remove the inefficiencies of Sybil-resistant mechanisms such as Proof-of-Work (i.e., high energy and environmental costs) and Proof-of-Stake (i.e., capital hoarding and lower transaction volume). The proposed solution effectively limits the number of mining nodes a single individual would be able to run while keeping membership open to everyone, circumventing the impossibility of full decentralization and the blockchain scalability trilemma when instantiated on a blockchain with a consensus protocol based on the cryptographic random selection of nodes. Resistance to collusion is also considered.

Solving one of the most pressing problems in blockchains, a zk-PoI cryptocurrency is proved to have the following advantageous properties:

- an incentive-compatible protocol for the issuing of cryptocurrency rewards based on a unique Nash equilibrium
- strict domination of mining over all other PoW/PoS cryptocurrencies, thus the zk-PoI cryptocurrency becoming the preferred choice by miners is proved to be a Nash equilibrium and the Evolutionarily Stable Strategy
- PoW/PoS cryptocurrencies are condemned to pay the Price of Crypto-Anarchy, redeemed by the optimal efficiency of zk-PoI as it implements the social optimum
- the circulation of a zk-PoI cryptocurrency Pareto dominates other PoW/PoS cryptocurrencies
- the network effects arising from the social networks inherent to national identity cards and ePassports dominate PoW/PoS cryptocurrencies
- the lower costs of its infrastructure imply the existence of a unique equilibrium where it dominates other forms of payment

Keywords: zero-knowledge, remote attestation, anonymous credentials, incentive compatibility, dominant strategy equilibria, Nash equilibria, Price of Crypto-Anarchy, Pareto dominance, blockchain, cryptocurrencies

1 Introduction

Sybil-resistance for permissionless consensus comes at a big price since it needs to waste computation using Proof-of-Work (PoW), in addition to assuming that a majority of the participants must be honest. In contrast, permissioned consensus is able to overcome these issues assuming the existence of a Public-Key Infrastructure[DS83, DLS88, CL99] otherwise it would be vulnerable to Sybil attacks[Dou02]: indeed, it has been recently proved[PS18] that consensus without authentication is impossible without using Proof-of-Work. Proof-of-Stake, the alternative to PoW, is economically inefficient because participants must keep capital at stake which incentivise coin hoarding and ultimately leads to lower transaction volume.

Another major challenge in permissionless blockchains is scalability, both in number of participants and total transaction volume. Blockchains based on Proof-of-Work are impossible to scale because they impose a winner-take-all contest between rent-seeking miners who waste enormous amounts of resources, and their proposed replacements based on Proof-of-Stake don't exhibit the high decentralization desired for permissionless blockchains.

The solution proposed in this paper prevents Sybil attacks without resorting to Proof-of-Work and/or Proof-of-Stake on permissionless blockchains while additionally guaranteeing anonymous identity verification: towards this goal, zero-knowledge proofs of trusted PKI certificates (i.e., national identity cards and/or ePassports) are used to limit the number of mining nodes that a single individual could run; alternatively, a more efficient solution based on mutual attestation is proposed and demonstrated practical 4.2.5. Counterintuitively, the blockchain would still be permissionless even though using government IDs because the term "permissionless" literally means "without requiring permission" (i.e., to access, to join, ...) and governments would not be authorizing access to the blockchain; moreover, the goal is to be open to all countries of the world 4.3, thus its openness is indistinguishable from PoW/PoS blockchains (i.e., the union of all possible national blockchains equals a permissionless, open and global blockchain). Later papers in the literature [LSP24] agree with this definition of permissionless (see Definition 1 of [LSP24]) and further highlight the importance of the use of identities in blockchain protocols (see Section 5 of [LSP24]). Coincidently, the latest regulations [otCCAC18, Reu19] point to the obligation to verify and use real-world identities on blockchains, and the banning of contaminant cryptocurrency mining[BG19, DC19].

Blockchain research has focused on better consensus algorithms obviating that incentives are a central aspect of permissionless blockchains and that better incentive mechanisms would improve the adoption of blockchains much more that scalability improvements. To bridge this gap, new proofs are introduced to demonstrate that mining a new cryptocurrency based on Zero-Knowledge Proof-of-Identity would strictly dominate previous PoW/PoS cryptocurrencies, thus replacing them is proved to be a Nash equilibrium; additionally, the circulation of the proposed cryptocurrency would Pareto dominate other cryptocurrencies. Furthermore, thanks to the network effects arising from the network of users of

trusted public certificates, the proposed cryptocurrency could become dominant over previous cryptocurrencies and the lower costs of its infrastructure imply the existence of a unique equilibrium where it dominates other forms of payment.

1.1 Contributions

The main and novel contributions are:

- The use of anonymous credentials in permissionless blockchains in order to prevent Sybil attacks 4: previous works[KITD17, DYSZ17] considered the use of PKI infrastructures in blockchains (i.e., permissioned ledgers) but without transforming them into anonymous credentials in order to obtain the equivalent of a permissionless blockchain. Other works have considered anonymous credentials on blockchains[GGM13, SABBD18, CDD17, FMMO18, Res18], but requiring the issuance of new credentials and not reusing previously existing ones: verifying real-world identities and issuing their corresponding digital certificates is the most expensive part of any real-world deployment.
  - The practical implementation and its perfomance evaluation 4.2.5 for national identity cards and ePassports.
- Circumventing the impossibility of full decentralization 4.4 and the blockchain scalability trilemma.
- A protocol for an incentive-compatible cryptocurrency 1: previous blockchains mint cryptocurrencies tied to the process of reaching a consensus on the order of the transactions, but the game-theoretic properties of this mechanism is neither clear nor explicit.
- A proof that mining the proposed cryptocurrency is a dominant strategy over other PoW/PoS blockchains and a Nash equilibrium over previous cryptocurrencies 5.2.1, in addition to an Evolutionary Stable Strategy 5.2.2.
- The insight that the optimal efficiency of zk-PoI resides in that it's implementing the social optimum, unlike PoW/PoS cryptocurrencies that have to pay the Price of (Crypto-)Anarchy 5.2.3.
- A proof that the circulation of the proposed zk-PoI cryptocurrency Pareto dominates other PoW/PoS cryptocurrencies 5.2.4.
- A proof that the proposed cryptocurrency could become dominant over previous ones due to stronger network effects and the lack of acceptance of previous cryptocurrencies as a medium of payment 5.2.5.
- Finally, the lower costs of its infrastructure imply the existence of a unique equilibrium where it dominates other forms of payment 5.2.6.

2 Related Literature

This section discusses how the present paper is significantly better and more innovative than previous approaches in order to fulfill the objective of providing a Sybil-resistant and permissionless blockchain with anonymous transaction processing nodes (i.e., miners). Moreover, it's considerably cheaper than other approaches[SABBD18, BKKJ$^+$17] that would require the re-identification and issuing of new identities to the global population because the current proposal relies on the previously issued credentials of electronic national identity cards (3.5 billion issued at the time of publication) and electronic passports (1 billion issued at the time of publication).

Proof of Space[DFKP13, ABFG13] reduces the energy costs of Proof-of-Work but it's not economically efficient. Proof of Authority[Woo15](PoA) maintains a public list of previously authorised nodes: the identities are not anonymised and the blockchain is not open to everyone (i.e., the blockchain is permissioned). Proof of Personhood[BKKJ$^+$17](PoP) can be understood as an improvement over Proof of Authority in that identities are anonymised, but the parties/gatherings used to anonymise and incorporate identities into the blockchain don't scale to national/international populations and could compromise Sybil resistance because it's trivial to get multiple identities by using different disguises on different parties/gatherings (i.e., they need to be validated simultaneously and without disguises): however, the present paper produces Sybil-resistant, anonymised identities on a global scale for a permissionless blockchain. Moreover, Proof of Personhood[BKKJ$^+$17] is endogenizing all the costly process of credential verification and issuing: by contrast, Zero-Knowledge Proof of Identity is exogenizing/outsourcing this costly process to governments, thus making the entire blockchain system cheaper. More recently, Private Proof-of-Stake protocols[GOT18, KKKZ18](PPoS) achieve anonymity, but the economic inefficiencies of staking capital still remain and the identities have no relation to the real world.

A conceptually close work ("*Decentralized Multi-authority Anonymous Authentication for Global Identities with Non-interactive Proofs*", [Ana19]), concurrently developed, doesn't reuse real-world certificates and therefore it would require that governments re-issue the cryptographic credentials of their citizens: therefore, it doesn't consider neither Sybil-resistance nor blockchain integration. Pseudo-anonymous signatures[BHK$^+$18] for identity documents provide an interesting technical solution to the problem of anonymous authentication using identity documents. However, the proposed schemes present a number of shortcomings that discourage their use in the present setting: some schemes are closely tied to particular countries (i.e., the German Identity Card[BDFK12, fIS16, KHK18]), thus non-general purpose enough to include any country in the world, or flexible to adapt to future changes; they require interaction with an issuer during card initialization; they feature protocols for deanonymisation and revocation, not desired in the setting considered in this paper; the initial German scheme[fIS16] could easily be subverted[KHK16] because the formalization of pseudo-anonymous signatures is still incipient[KHK],

and improvements are being worked out[BCLP14, Klu16, KHK18].

Anonymous credentials, first envisioned by David Chaum[Cha83], and first fully realised by Camenisch and Lysyanskaya[CL01] with follow-up work improving its security/performance[BCKL09, CHL05, CG10, CL04, BL12], are a centrally important building block in e-cash. The use of anonymous credentials to protect against Sybil attacks[Dou02] has already been proposed in previous works[AKMP08, BCKL07] although with different cryptographic techniques and for different goals. The main problem with anonymous credentials is that they require a first identification step to an issuing party[Res18] and that would compromise anonymity. This problem is shared with other schemes for pseudoanymization: for example, Bitnym[FWB15] requires that a Trusted Third Party must check the real identity of a user before allowing the creation of a bounded number of valid genesis pseudonyms. Decentralized Anonymous Credentials[GGM13] was first to show how to decentralise the issuance of anonymous credentials and integrate them within a blockchain (i.e., Bitcoin), but they do not re-use previously existing credentials and they still rely on Proof-of-Work for Sybil-resistance. Decentralized Blacklistable Anonymous Credentials with Reputation[YAXY17] introduce blacklistable reputation on blockchains, but users must also publish their real-world identity (i.e., non-anonymous). QuisQuis[FMMO18] introduces the novel primitive of updatable public keys in order to provide anonymous transactions in cryptocurrencies, but it doesn't consider their Sybil-resistance. DarkID[Arn18] is a practical implementation of an anonymous decentralised identification system, but requires non-anonymous pre-authentication and doesn't consider Sybil-resistance. A previous work[AABM17] on secure identity registration on distributed ledgers achieved anonymity from a credential issuer, but the pre-authentication is non-anonymous, it doesn't consider Sybil-resistance and it doesn't re-use real-world cryptographic credentials. Recently, anonymous credentials on standard smart cards have been proved practical[CDDH19], but in a different setting where the credential issuer and the verifier are the same entity.

Previous works have also considered anonymous PKIs: for example, generating pseudonyms[RPKC07] using a Certificate Authority and a separate Private Certificate Authority; however, this architecture is not coherent for a permissionless blockchain because both certificate authorities would be open to everyone and that would allow the easy linking of anonymous identities. Another recent proposal for a decentralised PKI based on a blockchain[PSRK18] does not provide anonymity, although it improves the work on cryptographic accumulators on blockchains started by Certcoin[FVY14, RY15]; another proposals introduce privacy-aware PKIs on blockchains[AG16, OP19], but they are not Sybil-resistant and do not re-use certificates from other CAs. Previously, BitNym[FWB15] introduced Sybil-resistant pseudonyms to Bitcoin, but a Trusted-Third Party must check the real identities of users before allowing the creation of a bounded number of valid genesis pseudonyms. ChainAnchor[HSP16] wasn't Sybil-resistant and used Direct Anonymous Attestation just for anonymous authentication, but not for mutual authentication: it worked on the permissioned model, explicitly not permissionless, and the GroupOwner initially knew the

true identity of members; moreover, the Permissions Issuer is supposed not to collude with the Verifier, although it has reading access to the identity database. ClaimChain[KITD17] improves the decentralised distribution of public keys in a privacy-preserving way with non-equivocation properties, but it doesn't consider their Sybil-resistance because it's more focused on e-mail communications. Blind Certificate Authorities[WAPaas18, WPasR16] can simultaneously validate an identity and prove a certificate binding a public key to it, without ever learning the identity, which sounds perfect for the required scenario except that it requires 3 parties and it's impossible to achieve in the 2-party setting; moreover, it doesn't consider Sybil-resistance.

Other approaches to anonymous identity include: Lightweight Anonymous Subscription with Efficient Revocation[KLL$^+$18], although it doesn't consider the real-world identity of users because it's focused on the host and its Trusted Platform Module; One Time Anonymous Certificates[AC10] extends the X.509 standard to support anonymity through group signatures and anonymous credentials, although it doesn't consider Sybil-resistance and their group signatures require that users hold two group secret keys, a requisite that is not allowed in the current scenario because the user is not trusted to store them on the national identity card (for the very same reasons, Linkable Ring Signatures[LWW04] and Linkable Message Tagging[GP14] are not allowed as cryptographic tools whilst group signatures and Deniable Anonymous Group Authentication[SPW$^+$14] would require a non-allowed setup phase). Opaak[MSCS12] provides anonymous identities with Sybil-resistance based on the scarcity of mobile phone numbers: however, users must register by receiving an SMS message (i.e., the Anonymous Identity Provider knows the real identity of participants). Oblivious PRFs[JKR18] are not useful in the permissionless blockchains because the secret key of the OPRF would be known by everyone, and the forward secrecy of the scheme that would provide security even if the secret key is known would not be of any use because the object identifiers ObjID would be easily predictable (i.e., derived from national identifiers). SPARTA[BBF$^+$07] provides pseudonyms through a distributed third-party-based infrastructure; however, it requires non-anonymous pre-registration. UnlimitID[IHD16] provides anonymity to OAuth/OpenID protocols, although users must create keypairs and keep state between and within sessions, a requisite that is not allowed in the current scenario. Another proposal for anonymous pseudonyms with one Trusted-Third Party[YL] requires a division of roles between the TTP and the server that is not coherent in a permissionless blockchain. With Self-Certified Sybil-free Pseudonyms[MKAP08, AKMP08], the user must keep state (i.e., dispenser D) generated by the issuer during enrollment and the Sybil-free identification is based on unique featurs of the devices, not on the user identity. Another anonymous authentication using smart cards[TLW13] is only anonymous from an eavesdropping adversary, not from the authentication server itself. TATA provides a novel way to achieve Sybil-resistant anonymous authentication: members of an induction group must interact and keep a list of who has already been given a pseudonym; therefore, a list of participants could be collected, but they can't be linked to their real-world identities; it's not clear how to bootstrap the

initial set of trusted users to get them to blindly sign each other's certificates.

Self-sovereign identity solutions usually rely on identities from social networks, but their Sybil-resistance is very questionable because almost half of their accounts could be fake[Nic19]: in spite of this, SybilQuorum[Dan18, SD19] proposes the use of social network analysis techniques to improve their Sybil-resistance; other research projects consider privacy-preserving cryptographic credentials from federated online identities[MJW+14].

Regarding the game-theoretic aspects, most papers focus on attacking only one cryptocurrency (e.g., selfish mining[ES13], miner's dilemma[Eya14], fork after withholding[KKS+17]). For a recent survey of these topics, see[AH19]. Exceptionally, "*Game of Coins*"[SKT18] considers the competition between multiple cryptocurrencies: a manipulative miner alters coin rewards in order to move miners to other cryptocurrencies of his own interest (with a fixed cost and a finite number of steps). However, in this paper, it's the cryptocurrency issuer who changes the rewards in order to attract miners from other cryptocurrencies by producing the most efficient cryptocurrency to mine.

	PoW	PoSpace	PoS	PPoS	PoA	PoP	**zk-PoI**
(Pseudo)-Anonymity	✓	✓	×	✓	×	✓	✓
Energy-Efficient	×	✓	✓	✓	✓	✓	✓
Economically Efficient	×	×	×	×	✓	×	✓
Permissionless	✓	✓	✓	✓	×	✓(*)	✓

Table 1: Comparison of different Sybil-resistant mechanisms.
(*) Only if open to everyone, with no selective pre-invitation and no right to exclude.

2.1 Proof-of-Personhood Considered Harmful (and Illegal)

To be considered lawful in the real world, Proof-of-Personhood (PoP, [BKKJ+17]) requires the concurrence of multiple unrestricted freedoms: assembly, association, and wearing of masks. However, in most countries these freedoms are limited:

- freedom of assembly[Wik20b] and association[Wik20c]: most countries usually require previous notification and permission from the governing authorities, that may reject for multiple grounds including but not limited a breach of public order. Thus, PoP cannot be considered permissionless in these countries.

- it's forbidden to wear a mask in most countries[Wik20a], as required for the anonimity of PoP ("All parties are recorded for transparency, but attendees are free to hide their identities by dressing as they wish, including hiding their faces for anonymity.", [BKKJ+17]). Thus, PoP won't be anonymous in countries that outlaw the covering of faces.

- promoters and organizers of PoP parties may themselves be committing a crime, due to incitement, conspiracy and complicity.

The solution proposed in this paper it's the only possible lawful one according to current regulations that require the use of national IDs to register on blockchains (AMLD5[Par18], FATF[FAT19], Cyberspace Administration of China[Reu19, otCCAC18]).

3 Building Blocks

3.1 Consensus based on the Cryptographic Random Selection of Transaction Processing Nodes

The new family of consensus algorithms based on the cryptographically random selection of transaction processing nodes[PS16, DPS16, KKJG$^+$17, GHM$^+$17, HMW18] is characterised by:

Consensus algorithm	Random selection method	Sybil resistance
OmniLedger	PVSS + collective BLS/BDN signatures [SJK$^+$16, KKJG$^+$17, BDN18, KK19]	PoW/PoS
RapidChain	Performance improvements over OmniLedger[ZMR18]	PoS
Algorand	Cryptographic sortition by a unique digital signature	PoS
Dfinity	BLS threshold signature scheme[BLS01]	PoS
Snow White	Extract public keys based on the amount of currency owned	PoS

- Transaction processing workers/nodes are randomly selected from a larger group: in the case of Dfinity[HMW18], an unbiasable, unpredictable verifiable random function (VRF) based on the BLS threshold signature scheme[BLS01] with the properties of uniqueness and non-interactivity; in the case of OmniLedger[KKJG$^+$17], the original proposal used a collective Schnorr threshold signature scheme[STV$^+$15, SJK$^+$16, KKJG$^+$17], although it has been updated to collective BLS/BDN signatures[BDN18] and now it uses MOTOR[KK19] instead of ByzCoin[KJG$^+$16] with improvements for open and public settings; in the case of Algorand, secure cryptographic sortition is generated using an elliptic curve-based verifiable random function (ECVRF-ED25519-SHA512-Elligator2[GRPV19]); in the case of Snow White, cryptographic committee reconfiguration is done by extracting public keys from the blockchain based on the amount of currency owned. For a detailed comparison of random beacon protocols, see [SJSW18].

- Regular time intervals (also named epochs or rounds) on which randomly selected workers/nodes process the transactions.
- Faster transaction confirmation and finality.
- High scalability.
- Decoupling Sybil-resistance from the consensus mechanism (PoW/PoS is about membership, not consensus).
- PoW/PoS to protect against Sybil attacks: however, the present paper proposes the use of Zero-Knowledge Proof-of-Identity (i.e., more economically[Dia19] and environmentally efficient[KT18, SKG19]).

3.2 X.509 Public Key Infrastructure

X.509 is an ITU-T standard[Uni18b] defining the format of public key certificates, itself based on the ASN.1 standard[Uni18a]: these certificates underpin most implementations of public key cryptography, including SSL/TLS and smartcards. An X.509v3 certificate has the following structure:

- Certificate
  - Version Number
  - Serial Number
  - Signature Algorithm ID
  - Issuer Name
  - Validity Period:
    - Not Before
    - Not After
  - Subject Name
  - Subject Public Key Info
    - Public Key Algorithm
    - Subject Public Key
  - Issuer Unique Identifier (optional)
  - Subject Unique Identifier (optional)
  - Extensions (optional):
    - Key Usage (optional)
    - Authority Information Access (optional)
    - Certificate Policies (optional)
    - Basic Constraints (optional)
    - CRL Distribution Points (optional)
    - Subject Alternative Name (optional)
    - Extended Key Usage (optional)
    - Subject Key Identifier (optional)
    - Authority Key Identifier (optional)
- Certificate Signature Algorithm
- Certificate Signature

Certificates are signed creating a certificate chain: the root certificate of an organization is a self-signed certificate that signs intermediate certificates that themselves are used to sign end-entities certificates. To obtain a signed certificate, the entity creates a key pair and signs a Certificate Signing Request (CSR) with the private key: the CSR contains the applicant's public key that is used to verify the signature of the CSR and a unique Distinguished Name within the organization. Then, one of the intermediate certificate authorities issues a certificate binding a public key to the requested Distinguished Name and that also contains information identifying the certificate authority that vouches for this binding.

The certificate validation chain algorithm checks the validity of an end-entity

certificate following the next steps:

1. The certificates are correct regarding the ASN.1 grammar of X.509 certificates.
2. The certificates are within their validity periods (i.e., non-expired).
3. If access to a Certificate Revocation List is granted, the algorithm checks that none of the certificates is included (i.e., the certificate has not been revoked).
4. The certificate chain is traversed checking that:
   (a) The issuer matches the subject of the next certificate in the chain.
   (b) The signature is valid with the public key of the next certificate in the chain.
5. The last certificate is a valid self-signed certificated trusted by the end-entity checker.

Additionally, the algorithm could also check complex application policies (i.e., the certificate can be used for web server authentication and/or web client authentication).

3.3 Electronic Passports

Data Group	**Data Elements**
Data Group 1	Document Types
	Issuing State or Organizaton
	Name (of Holder)
	Document Number
	Check Digit - Doc Number
	Nationality
	Date of Birth
	Check Digit - DOB
	Sex
	Date of Expiry or Valid Until Date
	Check Digit DOE/VUD
	Optional Data
	Check Digit - Optional Data Field
	Composite Check Digit
Data Group 11	Personal Number
Data Group 15	User's Public Key

Table 2: Data Groups from Electronic Passports

Modern electronic passports feature NFC chips[ICA15b] that contain all their printed information in digital form, using a proprietary format set by International Civil Aviation Organization[ICA15a] and not X.509 certificates 3.2 like the ones used in national identity cards: the relevant fields are contained within its Data Group 1 2 (i.e., the same information available within the Machine Readable Zone), and the Document Security Object contains a hash of all the Data Groups signed by a Document Signing Certificate issued every three months (also stored on the passports), itself signed by a Country Signing Certificate Authority (all the certificates are available online[ICA18]). Additionally, the data within the NFC chips are cryptographically protected and it's necessary to derive the cryptographic keys by combining the passport number, date of birth and expiry date (i.e., BAC authentication).

Finally, note that the electronic identity cards of some countries can also work as ePassports (e.g., Spanish Identity Card -Documento Nacional de Identidad-).

3.4 Verifiable Computation

A public verifiable computation scheme allows a computationally limited client to outsource to a worker the evaluation of a function $F(u, w)$ on inputs u and w: other alternative uses of these schemes allow a verifier V to efficiently check computations performed by an untrusted prover P. More formally, the following three algorithms are needed:

Definition 1. (Public Verifiable Computation). A public verifiable computation scheme VC consists of three polynomial-time algorithms (Keygen, Compute, Verify) defined as follows:

- $(EK_F, VK_F) \leftarrow \text{Keygen}\left(F, 1^\lambda\right)$: the key generation algorithm takes the function F to be computed and security parameter λ; it outputs a public evaluation key EK_F and a public verification key VK_F.
- $(y, \pi_y) \leftarrow \text{Compute}\left(EK_F, u, w\right)$: the prover runs the deterministic worker algorithm taking the public evaluation key EK_F, an input u supplied by the verifier and an input w supplied by the prover. It outputs $y \leftarrow F(u, w)$ and a proof π_y of y's correctness (as well as of prover's knowledge of w).
- $\{0, 1\} \leftarrow \text{Verify}\left(VK_F, u, w, y, \pi_y\right)$: the deterministic verification algorithm outputs 1 if $F(u, w) = y$, and 0 otherwise.

A public verification computation scheme VC must comply with the following properties of correctness, security, and efficiency:

- Correctness: for any function F and any inputs $\underline{u, w}$ to F, if we run $(EK_F, VK_F) \leftarrow \text{Keygen}\left(F, 1^\lambda\right)$ and $(y, \pi_y) \leftarrow \text{Compute}\left(EK_F, u, w\right)$ then we always get $\text{Verify}\left(VK_F, u, w, y, \pi_y\right) = 1$.
- Efficiency: Keygen is a one-time setup operation amortised over many calculations and $Verify$ is computationally cheaper than evaluating F.

- Security: for any function F and any probabilistic polynomial-time adversary A, we require that

$$\Pr\left[(\hat{u}, \hat{w}, \hat{y}, \hat{\pi_y}) \leftarrow A\left(EK_F, VK_F\right) : F\left(\hat{u}, \hat{w}\right) \neq \hat{y}\right] \leq \text{negl}\left(\lambda\right)$$

  and

$$1 = \text{Verify}\left(VK_F, \hat{u}, \hat{w}, \hat{y}, \hat{\pi_y}\right) \leq \text{negl}\left(\lambda\right)$$

  where $\text{negl}\left(\lambda\right)$ denotes a negligible function of inputs λ.

Additionally, we require the public verification computation scheme VC to be succinct and zero-knowledge:

- Succinctness: the generated proofs π_y are of constant size, that is, irrespective of the size of the function F and inputs u and w.
- Zero-knowledge: the verifier learns nothing about the prover's input w beyond the output of the computation.

Practical implementations are Pinocchio[PGHR13] and Geppeto[CFH+14], or Buffet[WSH+14] and Pequin[Pro16](a simplified version of Pepper[SMBW12]).

3.4.1 Verifiable Validation of X.509 Certificates as Anonymous Credentials

The algorithm for certificate chain validation chain in section 3.2 can be implemented with the public verifiable computation scheme of section 3.4 using zk-SNARKS to obtain a verifiable computation protocol so that a certificate holder is able to prove that he holds a valid X.509 certificate chain with a unique Distinguished Name, without actually sending the public key to the verifier and selectively disclosing the contents of the certificate: in other words, we re-use existing certificate chains and PKI infrastructure without requiring any modifications, turning X.509 certificates into anonymous credentials. A previous work already demonstrated the technical and practical viability of this approach[DLFKP16]: the only handicap was that the proof generation could take a long time (e.g., more than 10 minutes) and large keys (e.g., 1 Gbyte)..

Recent research advances have improved[BGG17] the initial setup of the zk-SNARK protocol used to generate the Common-Reference String (CRS) with an MPC protocol, such that it's secure even if all participants are malicious (except one). And faster proving times could be obtained by efficiently composing the non-interactive proving of algebraic and arithmetic statements[AGM18] since QAP-based zk-SNARKs are only efficient for arithmetic representations and not algebraic statements, but at the cost of increasing the proof size.

In this paper, a practical implementation was completed to check a certificate chain with an additional validation policy and written as C code for Pequin[Pro16], then compiled into a public evaluation and verification keys: unfortunately, it isn't scalable to millions of users and/or the large circuits/constraints required to cover all the typologies of national identity cards/ePassports, thus

an implementation based on TEE and mutual attestation is the preferred implementation 4.2. The only zero-knowledge proof system that could be scalable enough[WZC+18] works on a computer cluster, thus it doesn't fit the setting of a single user authenticating on his own device, and a libsnark backend can't handle more than 4 million gates requiring more than an hour of computation.

3.5 Cryptographic Accumulators

Firstly devised by Benaloh and de Mare[BdM94], a cryptographic accumulator [DHS15] is a compact binding set of elements supporting proofs of membership and more space-efficient than storing all of the elements of the set; given an accumulator, an element, and a membership witness, the element's presence in the accumulated set can be verified. Generally speaking, an accumulator consists of four polynomial-time algorithms:

- $Generate\left(1^k\right)$: given the security parameter k, it instantiates the initial value of the empty accumulator.
- $Add\,(a, y) \rightarrow (a', w)$: adds the element y to the current state of the accumulator a producing the updated accumulator value a' and the membership witness w for y.
- $WitnessAdd\,(w, y) \rightarrow w'$: on the basis of the current state of a witness w and the newly added value y, it returns an updated witness w'.
- $Verify\,(a, y, w) \rightarrow \{true, false\}$: verifies the membership of y using its witness w on the current state of accumulator a.

The following are interesting security properties of accumulators:

- Dynamic accumulators[CL02]: accumulators supporting the removal of elements from the accumulator by means of a deletion algorithm $Removal()$ and a witness update algorithm $WitnessRemoval\,()$.
- Universality[LLX07]: accumulators supporting non-membership proofs, $NonWitnessAdd\,()$, $NonWitnessRemoval\,()$ and $NonVerify\,()$.
- Strong accumulators[CHKO12]: deterministic and publicly executable, meaning that it does not rely on a trusted accumulator manager.
- Public checkable accumulators, the correctness of every operation can be publicly verified.

Recent constructions of cryptographic accumulators specifically tailored for blockchains are: a dynamic, universal, strong and publicly checkable accumulator [FVY14]; an asynchronous accumulator[RY15] with low frequency update and old-accumulator compatibility (i.e., up-to-date witnesses can be verified even against an outdated accumulator); a constant-sized, fair, public-state, additive, universal accumulator[PSRK18], and an accumulator optimised for batch and aggregation operations[BBF18].

3.6 Remote Attestation

In the terminology of Intel SGX, remote attestation is used to prove that an enclave has been established without alterations of any kind: in other words, remote parties can verify that an application is running inside an SGX enclave. Concretely, remote attestation is used to verify three properties: the identity of the application, that it has not been tampered with, and that it is running securely within an SGX enclave. Remote attestation is carried out in several stages: requesting a remote attestation from the challenger; performing a local attestation of the enclave; converting said local attestation to a remote attestation; returning the remote attestation to the challenger, and the challenger verifying the remote attestation to the Intel Attestation Service.

A detailed technical description is outside of the scope of this paper: detailed descriptions can be found in the standard technical documentation[CD16, AGJS13, M18]. Recent attacks[VBMW$^+$18] can be used to extract the secret attestation keys used to verify the identity of an SGX enclave, and microcode updates must be installed[Cor18] to prevent their exploitation: that is, it's essential to check that parties to a remote attestation are using a safe and updated version. However, our protocols are inherently resistant to deniability attacks[GPA18] because they are based on mutual attestation.

As it would be shown in the next section 4.2, remote attestation can be used as a more efficient substitute of verifiable computation.

4 Authentication Protocols

In this section, we describe authentication protocols for Sybil-resistant, anonymous authentication using Zero-Knowledge protocols 4.1 and remote attestation 4.2.

4.1 Authentication Protocols using Zero-Knowledge

The use of zero-knowledge protocols guarantee the public-verifibility of the correctness of the Sybil-resistant, anonymous identities committed to the permissionless blockchain.

4.1.1 Security Goals

The following security goals must be met for the system to be considered secure:

1. The registered miner's key to the blockchain *opens, but no one can shut; he can shut, but no one can open* (*Isaiah 22:22*, [Isa00]). For the security of the system to be considered equivalent to the currently available permissionless blockchains, anyone holding a valid public certificate should be able to register a pseudo-anonymous identity on the blockchain but no one should be able to remove it (i.e., uncensorable free entry is guaranteed).

2. Protection against malicious issuers: some certification authorities may turn against some citizens and try to cancel access to the permissionless blockchain or stole their funds.

   (a) Mandatory passphrase. An issuer may counterfeit a certificate with the same unique identifiers, thus possessing a valid certificate isn't secure enough and a passphrase is deemed mandatory.

   (b) Non-bruteforceable. Operations must be computationally costly on the client side to prevent brute-forcing.

   (c) No OCSP checking. Prevention against malicious blacklisting.

3. Privacy: miner's real identity can't be learned by anyone.

4. Unique pseudonyms: from each identity card/ePassport, only one unique identifier can be generated.

5. Publicly verifiable: anyone should be able to verify the validity of the miner's public key and its pseudonym.

4.1.2 Zero-Knowledge Protocols (X.509)

Anonymous miner registration of a new public key on a permissionless blockchain. This protocol generates a unique pseudonym for each miner, and attaches a verifiable proof that its new public key to be stored on-chain is signed with a valid public certificate included on a recognised certification authorities list, and that the new public key is linked to the blockchain-specific pseudonym that is in turn uniquely linked to the citizen's public key certificate.

Miners holding a public key certificate must execute the following steps:

1. Create a deterministic public/secret key pair based on a secret passphrase (no need for verifiable computation):

$$pk, sk = \text{Det\_KeyPairGen}\left(KDF\left(passphrase, hash(publicCert)\right)\right)$$

   The generation algorithm must be determistic because the smartcard may be unable to store them and/or the miner may loose them (i.e., as in deterministic wallets). KDF is a password-based key derivation function (e.g., PBKDF2).

2. Obtain a signature of the previously generated public key pk with the miner's public key certificate (no need for verifiable computation, this operation could be executed on a smartcard):

$$sign_{PK} = \text{PKCS\_Sign}\left(secretKey_{publicCert}, pk\right)$$

3. Check the validity of the certificate chain of the miner's public key certificate as extracted from the smartcard:

(a) Load the public key of the root certificate.

(b) Hash and verify all intermediates, based on their certificate templates, and the public key of their parent certificate starting from the root certificate and following with the verified public key from the previous intermediate certificate template.

(c) Hash and verify the miner's public key certificate using the last verified public key returned from the previous step.

(d) Check the time validity of the miner's public key certificate.

(e) Check that the miner's public key certificate is contained on a list of trusted certification authorities.

4. Obtain the unique identifier from the miner's public key:

$$uniqueID = getID(publicCert)$$

Note that the unique identifier is usually contained on Serial Number of the certificate, or the Subject Alternative Name extension under different OIDs, depending on the country.

5. Generate a deterministic pseudonym using the blockchain identifier:

$$\begin{aligned} signatureSecret &= \text{PKCS\_Sign}\,(secretKey_{publicCert}, \\ &\quad \text{"PREFIXED\_COMMON\_STRING"}) \end{aligned}$$

$$\begin{aligned} pseudonym &= Hash\,(signatureSecret||BlockchainIdentifier||uniqueID) \\ &\quad ||\text{"REG"} \end{aligned}$$

PKCS_Sign is the deterministic PKCS#1.5 signing algorithm executed on a prefixed string to obtain a unique, non-predictable secret based on the certificate's owner. The obtained signature is appended to the blockchain identifer and the unique identifier, and then hashed to derive a unique pseudonym. Finally, the string "REG" is appended to differentiate this pseudonym from the one generated during a remove protocol and prevent replay attacks for removal reusing the generated zero-knowledge proof.

6. Verify the signature $sign_{PK}$ on the miner's public key certificate pk:

$$\text{PKCS\_Verify}\,(publicCert, sign_{PK})$$

7. As the *signatureSecret* is calculated offline by the smartcard, it's also necessary to verify it using the miner's public key certificate *publicCert*:

$$\text{PKCS\_Verify}\,(publicCert, signatureSecret)$$

8. Generate the zero-knowledge proof π (e.g., zk-SNARK, zk-STARK or zk-SNARG) of the miner's public key certificate pk, the generated pseudonym and, signature $sign_{PK}$ such that all the previous conditions 3-7 hold.

9. Anonymously contact the permissionless blockchain:

   (a) optionally, check the miner's real identity on a cryptographic accumulator:

      i. establish a shared secret running a Diffie-Hellman key exchange between the prospective miner and the permissionless blockchain

      ii. send attributes of the miner's real identity encrypted with the shared secret

      iii. execute the non-membership proof $NonWitnessAdd(w, y)$ on the cryptographic accumulator

   (b) register the generated pseudonym, the new public key pk, the signature $sign_{PK}$ and π: note that they don't reveal the miner's real identity ($publicCert$, $uniqueID$ and $signatureSecret$ are all keep as a secret).

The registering node of the permissionless blockchain verifies π before adding the new public key, the associated pseudonym, the signature $sign_{PK}$ and the succinct proof π: note that the miner is unable to register multiple pseudonyms, and he can only use one running node that would be signing messages with the generated secret key sk. Other nodes would be able to efficiently verify π to confirm that the public key pk is a signed by someone from an allowed certificate authority, and that the pseudonym is the miner's unique alias for the blockchain.

Taking offline registrations from a permissionless blockchain. This protocol takes offline a pseudonym and its associated public key pk and signature $sign_{PK}$ from a permissionless blockchain. Miners must execute the following steps to take offline an identity from a permissionless blockchain:

1. Generate a zero-knowledge proof π (e.g., zk-SNARK, zk-STARK or zk-SNARG) of the steps 3-7 of the previous protocol to prove secret knowledge of sk and that he's able to re-generate the pseudonym, but this time appending the string "OFF" to the pseudonym.

2. Anonymously contact the permissionless blockchain to take offline the generated pseudonym and all its associated data (including the cryptographic accumulator), attaching π.

The registering node of the permissionless blockchain verifies π before taking offline the pseudonym without learning the real identity of the miner (publicCert, uniqueID and signatureSecret remain secret).

4.1.3 Zero-Knowledge Protocols (ePassports)

Analogous to the zero-knowledge protocols for X.509 4.1.2, but now considering the specific details of ePassports 3.3, which usually contain a unique keypair with the public key on Data Group 15 and the private key hidden within the chip: the Active Authentication protocol can be used to sign random challenges

that can be verified with the corresponding public key. Some ePassports don't feature Active Authentication, nonetheless a modified version of the following protocols could still be executed (see subsection 4.1.5).

Anonymous miner registration of a new public key on a permissionless blockchain. This protocol generates a unique pseudonym for each miner, and attaches a verifiable proof that its new public key to be stored on-chain is signed with a valid public certificate included on the list of Country Signing Certificate Authorities, and that the new public key is linked to the blockchain-specific pseudonym that is in turn uniquely linked to the public key certificate of the passport holder.

Miners holding a public key certificate must execute the following steps:

1. Create a deterministic public/secret key pair based on a secret passphrase (no need for verifiable computation):

$$pk, sk = \text{Det\_KeyPairGen}\left(KDF\left(passphrase, hash(publicCert)\right)\right)$$

   The *publicCert* is taken from the Data Group 15. KDF is a password-based key derivation function (e.g., PBKDF2).

2. Obtain a signature of the previously generated public key pk with the miner's public key certificate (no need for verifiable computation, this operation is executed within the ePassport's chip using the Active Authentication protocol):

$$sign_{PK} = \text{Sign}\left(secretKey_{publicCert}, pk\right)$$

3. Check the validity of the Data Security Object of the miner's ePassport:

   (a) Load the public key of the Country Signing Certificate from a trusted source [ICA18] and the Document Signing Certificate from the ePassport.

   (b) Hash all the Data Groups and check their equivalence to the Data Security Object.

   (c) Verify the signature of the Data Security Object using the Document Signing Certificate.

   (d) Verify the signature of the Document Signing Certificate using the Country Signing Certificate.

   (e) Check the time validity of the certificates.

4. Obtain the unique identifier of the ePassport:

$$uniqueID = getID(DataGroups)$$

   Note that the unique identifier is usually contained on the Data Element "Document Number" of the Data Group 1: as it's legally valid for the same person to own multiple passports with different Document Numbers, some countries include a unique "Personal Number" on the Data Group 11.

5. Generate a deterministic pseudonym using the blockchain identifier:

$$\begin{aligned} signatureSecret &= \text{Sign}\,(secretKey_{publicCert}, \\ &\quad \text{"PREFIXED\_COMMON\_STRING"}) \end{aligned}$$

$$\begin{aligned} pseudonym &= Hash\,(signatureSecret||BlockchainIdentifier||uniqueID) \\ &\quad ||\text{"REG"} \end{aligned}$$

Sign is the Active Authentication protocol executed within the ePassport's chip, a deterministic signing algorithm executed on a prefixed string to obtain a unique, non-predictable secret based on the certificate's owner. The obtained signature is appended to the blockchain identifer and the unique identifier, and then hashed to derive a unique pseudonym. Finally, the string "REG" is appended to differentiate this pseudonym from the one generated during a remove protocol and prevent replay attacks for removal reusing the generated zero-knowledge proof (e.g., zk-SNARK, zk-STARK or zk-SNARG).

6. Verify the signature $sign_{PK}$ on the miner's public key certificate pk:

$$\text{PKCS\_Verify}\,(publicCert, sign_{PK})$$

The *publicCert* is taken from the Data Group 15.

7. As the *signatureSecret* is calculated offline by the ePassport's chip, it's also necessary to verify it using the miner's public key certificate *publicCert*:

$$\text{PKCS\_Verify}\,(publicCert, signatureSecret)$$

The *publicCert* is taken from the Data Group 15.

8. Generate the zero-knowledge proof π (e.g., zk-SNARK, zk-STARK or zk-SNARG) of the miner's public key certificate pk, the generated pseudonym, and signature $sign_{PK}$ such that all the previous conditions 3-7 hold.

9. Anonymously contact the permissionless blockchain

   (a) optionally, check the miner's real identity on a cryptographic accumulator:

      i. establish a shared secret running a Diffie-Hellman key exchange between the prospective miner and the permissionless blockchain
      ii. send attributes of the miner's real identity encrypted with the shared secret
      iii. execute the non-membership proof $NonWitnessAdd\,(w, y)$ on the cryptographic accumulator

   (b) register the generated pseudonym, the new public key pk, the signature $sign_{PK}$ and π: note that they don't reveal the miner's real identity (*publicCert*, $uniqueID$ and *signatureSecret* are all keep as a secret).

The registering node of the permissionless blockchain verifies π before adding the new public key, the associated pseudonym, the signature $sign_{PK}$ and the succinct proof π: note that the miner is unable to register multiple pseudonyms, and he can only use one running node that would be signing messages with the generated secret key sk. Other nodes would be able to efficiently verify π to confirm that the public key pk is a signed by someone from an allowed certificate authority and that the pseudonym is the miner's unique alias for the blockchain.

Taking offline registrations from a permissionless blockchain. This protocol takes offline a pseudonym and its associated public key pk and signature $sign_{PK}$ from a permissionless blockchain. Miners must execute the following steps to take offline an identity from a permissionless blockchain:

1. Generate a zero-knowledge proof π (e.g., zk-SNARK, zk-STARK or zk-SNARG) of the steps 3-7 of the previous protocol to prove secret knowledge of sk and that he's able to re-generate the pseudonym, but this time appending the string "OFF" to the pseudonym.
2. Anonymously contact the permissionless blockchain to take offline the generated pseudonym and all its associated data (including the cryptographic accumulator), attaching π.

The registering node of the permissionless blockchain verifies π before taking offline the pseudonym without learning the real identity of the miner (publicCert, uniqueID and signatureSecret remain secret).

4.1.4 Mapping to goals

The previous protocols achieve the security goals:

1. The registered miner's key to the blockchain *opens, but no one can shut; he can shut, but no one can open.* Only someone in possession of a valid public certificate can create a unique miner identity on the open blockchain and destroy it. Please note that the signing and verification of steps 2, 5, 6 and 7 are only needed if it's required to check that the miner is the real owner of the smartcard/ePassport.
2. Protection against malicious issuers: the passphrase is mandatory, there's no OCSP checking and the protocol is non-bruteforceable because it requires the generation of a proof π for every passphrase that is going to be tried (>60 secs per π).
3. Privacy: miner's real identity can't be learned by anyone because publicCert and uniqueID are keep secret.
4. Unique pseudonyms: from each identity card/ePassport, only one unique identifier can be generated because there's only one uniqueID per citizen.
5. Publicly verifiable: using the proof π, anyone is able to validate the miner's public key and its pseudonym.

Additionally, cryptographic accumulators could be added to the protocols in order to prevent multiple registrations whenever an expired certificate is renovated.

4.1.5 Absence of Active Authentication

Signing using the secret key of the Active Authentication protocol provides an extra layer of security: it guarantess that the remote party executing the protocol owns a physical copy of the ePassport (i.e., it hasn't stolen a copy of the public certificates from others). However, some ePassports don't feature Active Authentication, requiring a simplified version of the previous protocols:

- Steps 2,6 and 7 are removed.
- Step 5 doesn't calculate the signature.
- The zero-knowledgeπ is extended to Step 1, with a password-based key derivation function using less steps.

4.2 Detailed Authentication Protocols using Mutual Attestation

The use of remote attestation protocols guarantee the efficiency and scalability of the full authentication solution (i.e., it can easily scale to billions of users). By design, the architecture has detached the encrypted DB from the mining nodes to maintain the implementation as blockchain-agnostic as possible: some mining nodes may include an encrypted DB, but it's not necessary that all mining nodes include it.

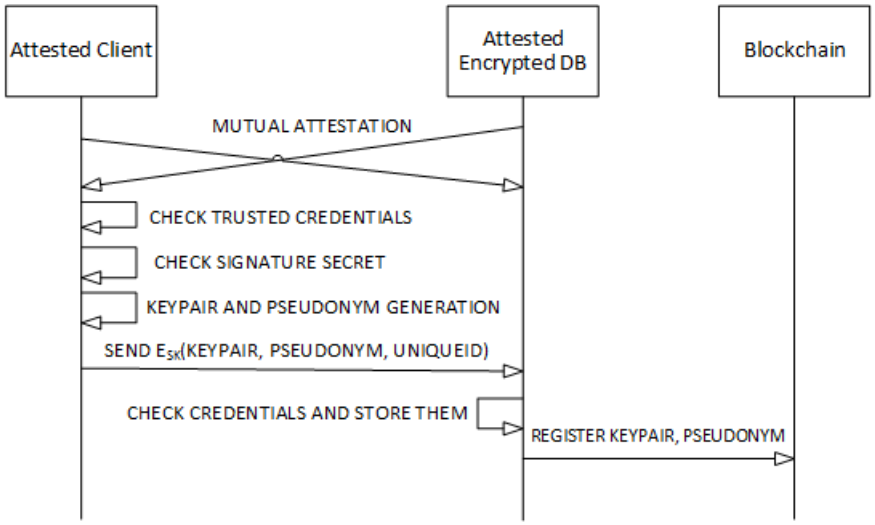


Figure 4.1: Simplified overview of mutual attestation.

4.2.1 Security Goals

The following security goals must be met for the system to be considered secure:

1. The registered miner's key to the blockchain *opens, but no one can shut; he can shut, but no one can open* (*Isaiah 22:22*, [Isa00]). For the security of the system to be considered equivalent to the currently available permissionless blockchains, anyone holding a valid public certificate should be able to register a pseudo-anonymous identity on the blockchain but no one should be able to remove it (i.e., uncensorable free entry is guaranteed).

2. Protection against malicious issuers: some certification authorities may turn against some citizens and try to cancel access to the permissionless blockchain or stole their funds.

   (a) Mandatory passphrase. An issuer may counterfeit a certificate with the same unique identifiers, thus possessing a valid certificate isn't secure enough and a passphrase is deemed mandatory.

   (b) Non-bruteforceable. Operations must be computationally costly on the client side to prevent brute-forcing.

   (c) No OCSP checking. Prevent against malicious blacklisting.

3. Privacy: miner's real identity can't be learned by anyone.

4. Unique pseudonyms: from each identity card/ePassport, only one unique identifier can be generated.

4.2.2 Mutual Attestation for X.509 Certificates

Anonymous miner registration of a new public key on a permissionless blockchain. This protocol generates a unique pseudonym for each miner, with a new public key linked to the blockchain-specific pseudonym that is in turn uniquely linked to the citizen's public key certificate: the mutual attestation between the parties guarantees the correctness of the execution of both parties.

The following are the steps to the protocol:

1. The client locally generates a signature secret using its secret key:

$$\begin{aligned} signatureSecret \quad = \quad & \text{PKCS\_Sign}\,(secretKey_{publicCert}, \\ & \text{"PREFIXED\_COMMON\_STRING"}) \end{aligned}$$

2. Mutual attestation between the authenticating client and the blockchain: the attestation is anonymous thanks to the use of unlinkable signatures (Enhanced Privacy ID -EPID-), and both parties obtain a temporary secret key to encrypt their communications.

3. Client's attested code checks the validity of the certificate chain of the miner's public key certificate as extracted from the smartcard:

(a) Load the public key of the root certificate.

(b) Hash and verify all intermediates, based on their certificate templates, and the public key of their parent certificate starting from the root certificate and following with the verified public key from the previous intermediate certificate template.

(c) Hash and verify the miner's public key certificate using the last verified public key returned from the previous step.

(d) Check the time validity of the miner's public key certificate.

(e) Check that the miner's public key certificate is contained on a list of trusted certification authorities.

4. If the previous step concluded satisfactorily, then the client's attested code verifies the $signatureSecret$ using the miner's public key certificate $publicCert$ because the $signatureSecret$ is calculated offline by the smartcard:

$$\text{PKCS\_Verify}\left(publicCert, signatureSecret\right)$$

5. If the previous step concluded satisfactorily, then the client's attested code creates a deterministic public/secret key pair based on a secret passphrase:

$$pk, sk = \text{Det\_KeyPairGen}\left(KDF\left(passphrase, hash(publicCert)\right)\right)$$

The generation algorithm must be deterministic because the smartcard may be unable to store them and/or the miner may lose them (i.e., as in deterministic wallets). KDF is a password-based key derivation function (e.g., PBKDF2).

6. The client's attested code generates a deterministic pseudonym using the blockchain identifier:

$$\begin{aligned} pseudonym \quad &= \quad Hash\left(signatureSecret||BlockchainIdentifier||uniqueID\right) \\ &\quad\quad ||\text{"REG"} \end{aligned}$$

and it obtains the unique identifier from the miner's public key:

$$uniqueID = getID(publicCert)$$

Note that the unique identifier is usually contained on Serial Number of the certificate, or the Subject Alternative Name extension under different OIDs, depending on the country.

7. Anonymously contact the attested encrypted database of the permissionless blockchain to register the generated pseudonym and the new public key pk: the uniqueID is also included using the temporary encrypted key, but it won't be revealed to the host computer of the blockchain node because it will only be decrypted within the attested enclave.

8. The blockchain's attested code checks within its encrypted database that the uniqueID has never been included: then, it proceeds to store the encrypted uniqueID (i.e., this time with a database secret key that only resides within the enclaves), the generated pseudonym and the new public key pk.

9. Then, the encrypted database's attested code contacts the permissionless blockchain to register the generated pseudonym and its new public key pk.

Taking offline registrations from a permissionless blockchain. This protocol takes offline a pseudonym and its associated public key pk from a permissionless blockchain. To take offline an identity from a permissionless blockchain, miners must re-run the previous protocol to prove that the client is able to re-generate the pseudonym with the same certificate, but this time appending the string "OFF" to the pseudonym.

The registering encrypted database of the permissionless blockchain verifies that the encrypted uniqueID is included in the database before taking offline the pseudonym from the permissionless blockchain without it learning the real identity of the miner.

4.2.3 Mutual Attestation for ePassports

Analogous to the zero-knowledge protocols for X.509 4.2.2, but now considering the specific details of ePassports 3.3, which usually contain a unique keypair with the public key on Data Group 15 and the private key hidden within the chip: the Active Authentication protocol can be used to sign random challenges that can be verified with the corresponding public key. Some ePassports don't feature Active Authentication, nonetheless a modified version of the following protocols could still be executed (see subsection 4.2.6).

Anonymous miner registration of a new public key on a permissionless blockchain. This protocol generates a unique pseudonym for each miner, with a new public key linked to the blockchain-specific pseudonym that is in turn uniquely linked to the citizen's ePassport: the mutual attestation between the parties guarantees the correctness of the execution of both parties.

The following are the steps to the protocol:

1. The client locally generates a signature secret using its secret key:

$$\begin{aligned} signatureSecret \quad = \quad & \text{Sign}\,(secretKey_{publicCert}, \\ & \text{"PREFIXED\_COMMON\_STRING"}) \end{aligned}$$

2. Mutual attestation between the authenticating client and the blockchain: the attestation is anonymous thanks to the use of unlinkable signatures (Enhanced Privacy ID -EPID-), and both parties obtain a temporary secret key to encrypt their communications.

3. Client's attested code checks the validity of the Data Security Object of the miner's ePassport:

(a) Load the public key of the Country Signing Certificate from a trusted source[ICA18] and the Document Signing Certificate from the ePassport.

(b) Hash all the Data Groups and check their equivalence to the Data Security Object.

(c) Verify the signature of the Data Security Object using the Document Signing Certificate.

(d) Verify the signature of the Document Signing Certificate using the Country Signing Certificate.

(e) Check the time validity of the certificates.

4. If the previous step concluded satisfactorily, then the client's attested code verifies the *signatureSecret* using the miner's public key certificate *publicCert* because the *signatureSecret* is calculated offline by the ePassport's chip:

$$\text{PKCS\_Verify}\left(publicCert, signatureSecret\right)$$

The publicCert is taken from the Data Group 15.

5. If the previous step concluded satisfactorily, then the client's attested code creates a deterministic public/secret key pair based on a secret passphrase:

$$pk, sk = \text{Det\_KeyPairGen}\left(KDF\left(passphrase, hash(publicCert)\right)\right)$$

The generation algorithm must be deterministic because the ePassport is unable to store them and/or the miner may lose them (i.e., as in deterministic wallets). KDF is a password-based key derivation function (e.g., PBKDF2).

6. The client's attested code generates a deterministic pseudonym using the blockchain identifier:

$$\begin{aligned} pseudonym \quad &= \quad Hash\left(signatureSecret||BlockchainIdentifier||uniqueID\right) \\ &\quad\quad ||\text{"REG"} \end{aligned}$$

and it obtains the unique identifier from the ePassport:

$$uniqueID = getID(DataGroups)$$

Note that the unique identifier is usually contained on the Data Element "Document Number" of the Data Group 1: as it's legally valid for the same person to own multiple passports with different Document Numbers, some countries include a unique "Personal Number" on the Data Group 11. Sign is the Active Authentication protocol executed within the ePassport's chip, a deterministic signing algorithm executed on a prefixed string to obtain a unique, non-predictable secret based on the certificate's owner.

7. Anonymously contact the attested encrypted database of the permissionless blockchain to register the generated pseudonym and the new public key pk: the uniqueID is also included using the temporary encrypted key, but it won't be revealed to the host computer of the blockchain node because it will only be decrypted within the attested enclave.

8. The blockchain's attested code checks within its encrypted database that the uniqueID has never been included: then, it proceeds to store the encrypted uniqueID (i.e., this time with a database secret key that only resides within the enclaves), the generated pseudonym and the new public key pk.

9. Then, the encrypted database's attested code contacts the permissionless blockchain to register the generated pseudonym and its new public key pk.

Taking offline registrations from a permissionless blockchain. This protocol takes offline a pseudonym and its associated public key pk from a permissionless blockchain. To take offline an identity from a permissionless blockchain, miners must re-run the previous protocol to prove that the client is able to re-generate the pseudonym with the same certificate, but this time appending the string "OFF" to the pseudonym.

The registering encrypted database of the permissionless blockchain verifies that the encrypted uniqueID is included in the database before taking offline the pseudonym from the permissionless blockchain without it learning the real identity of the miner.

4.2.4 Mapping to goals

The previous protocols achieve the security goals:

1. The registered miner's key to the blockchain *opens, but no one can shut; he can shut, but no one can open*[Isa00]. Only someone in possession of a valid public certificate can create a unique miner identity on the open blockchain and destroy it. The signing and verification operations of steps 1 and 4 are only needed if it's required to check that the miner is the real owner of the smartcard/ePassport.

2. Protection against malicious issuers: the passphrase is mandatory, there's no OCSP checking and the protocol is non-bruteforceable because it can be rate-limited.

3. Privacy: miner's real identity can't be learned by anyone because publicCert and uniqueID are keep secret.

4. Unique pseudonyms: from each identity card/ePassport, only one unique identifier can be generated because there's only one uniqueID per citizen.

The proposed solution depends on the security of Intel SGX (enclave and remote attestation protocols): in order to limit the impact of side-channels attacks on

Intel SGX, mining nodes featuring the role of the Attested Encrypted DB will be restricted to trustworthy nodes.

4.2.5 Performance Evaluation

# VMs	Mean Time/Req.	#Req./Sec	Time/Connections	Total time
1 VM	416 ms	4.76	210 ms	21 secs
4 VM	112 ms	16.5	59 ms	5.9 secs

A load testing scenario featuring an Intel Xeon E3-1240 3.5 GHz and running 1 or 4 virtual machines was performed (with 5 users executing 100 requests per user). Operations like reading and/or signing from the smartcard were not included in the performance evaluation. The implementation will be open-sourced.

4.2.6 Absence of Active Authentication

Signing using the secret key of the Active Authentication protocol provides an extra layer of security: it guarantess that the remote party executing the protocol owns a physical copy of the ePassport (i.e., it hasn't stolen a copy of the public certificates from others). However, some ePassports don't feature Active Authentication, requiring a simplified version of the previous protocols by removing steps 1 and 4.

4.2.7 Removing Single-Points of Failure

One of the shortcomings of relying on Intel's Attestation Service (IAS) is that it becomes a single-point of failure: in practice, Intel would learn who is performing the attestation. For a public, permissionless blockchain it would be preferable to remove this trusted third party: to solve this problem, OPERA[CZL19] provides the first open, privacy-preserving attestation service to substitute Intel's Attestation Service.

4.2.8 Substituting EPID for DCAP

Since Intel is deprecating EPID (Enhanced Privacy ID) in favour of DCAP (Data Center Attestation Primitives [SJBZ18, SFF20]), a new re-implementation of the previously described mutual attestation protocol 4.2.3 has been carried out using Occlum[STC$^+$20]: additionally, one of the benefits of DCAP is that it removes Intel as a trusted third party as it doesn't use Intel's Attestation Service.

Remote biometric authentication using smartphones has also been improved with remote attestation and DCAP, as described below 4.3.3.

4.3 Worldwide Coverage and Distribution

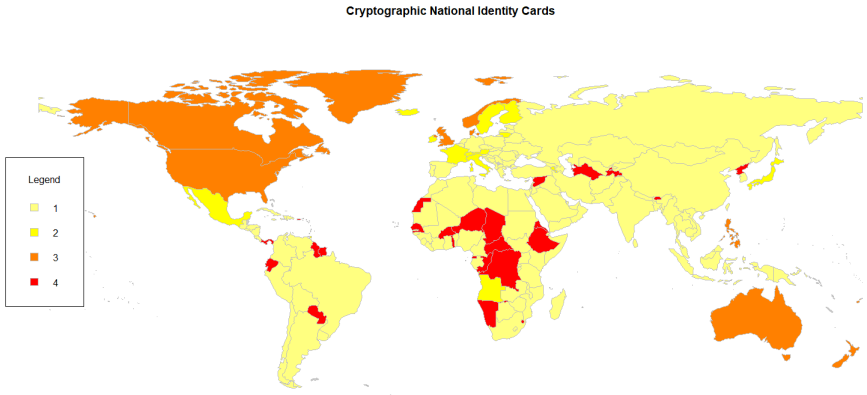


Figure 4.2: Legend: (1) National identity card is a mandatory smartcard; (2) National identity card is a voluntary smartcard; (3) No national identity card, but cryptographic identification is possible using an ePassport, driving license and/or health card; (4) Non-digital identity card.

Fortunately, there is a unique cryptographic identifier for most people in the world: figure 4.2 shows a worldmap of the distribution of national identity cards. For some countries, there is no national identity card -code 3-, but some other unique cryptographic identifier is available (e.g., ePassport[Nit09] and/or biometric passports as in figure 4.3, social security card, driving license and/or health card). Transforming these unique identifiers into anonymous credentials enables the unique identification of individuals in a permissionless blockchain without revealing their true identities, making them indistinguishable: that is, authentication is not only anonymous but permissionless since there is no need to be pre-invited. Please note the enormous cost savings resulting from this approach compared to other anonymous credential[GGM13, SABBD18, CDD17] proposals that would require re-issuing new credentials: for example, consider that the UK's national identity scheme was estimated at £5.4bn[Pro08].

In some cases, an individual could obtain multiple cryptographic identifiers (e.g., multiple nationalities), but their number would still be limited and certainly less than the number of mining nodes that could be spawned on PoW permissionless blockchains. Additionally, the true identities provided by national identity cards could be used for other purposes, such as non-anonymous accounts identified by their legal identities.

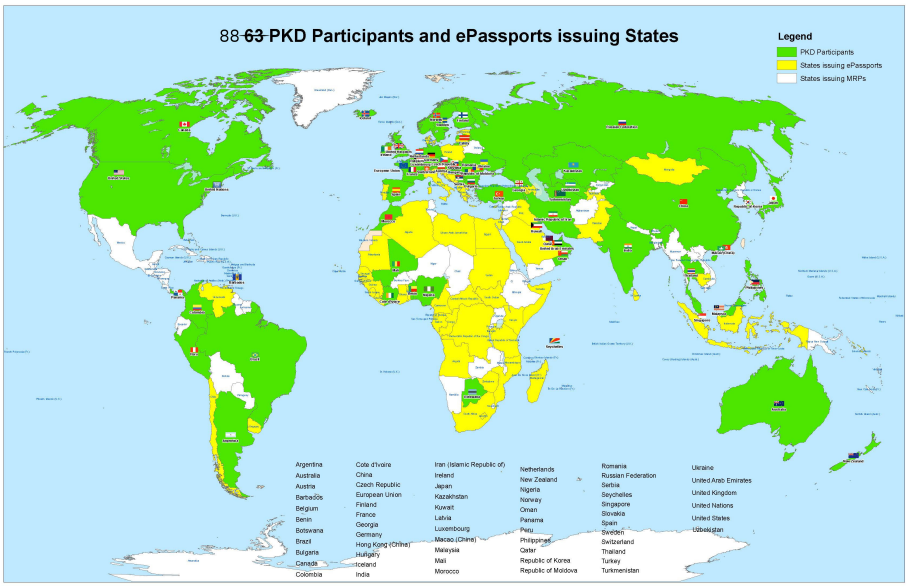
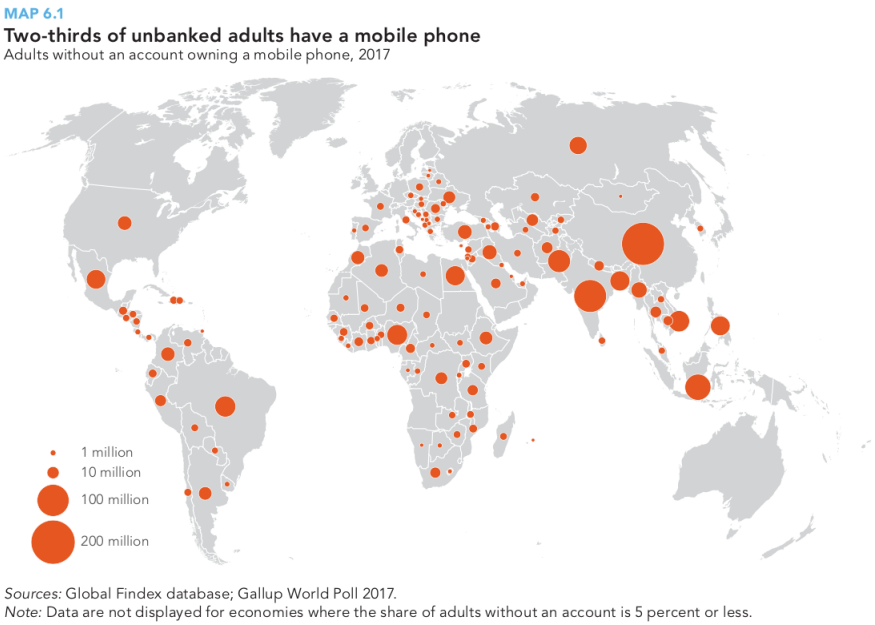


Figure 4.3: Availability of biometric passports. Source (ICAO, 2019)



4.3.1 eSIM's Public Key Infrastructure

Latest specifications of SIM cards determine that SIM's identity and data can be downloaded and remotely provisioned to devices[GSM18]: instead of the traditional SIM card, there is an embedded SIM (i.e., eUICC[All19]) that can store multiple SIM profiles containing the operator and subscriber data that would be stored on a traditional SIM card (e.g., IMSI, ICCID, ...).

A novel public key infrastructure has been created in order to protect the distribution of these new eSIM profiles[Ass17]: every certified eSIM is signed by its certified manufacturer, with a certificate that is itself signed by the GSMA root certificate issuer[GSM19]. Network operators must also get certified and obtain certificates for their Subscription Management roles.

The eSIM's PKI provides an alternative identification system for users where national identity cards and/or ePassports are difficult to obtain, as they must be unique and non-anonymous (4.13 and 4.1.5[Ass17]), but only when the mobile operator's KYC processes can be considered trustworthy.

4.3.2 Combining with Non-Zero-Knowledge Authentication

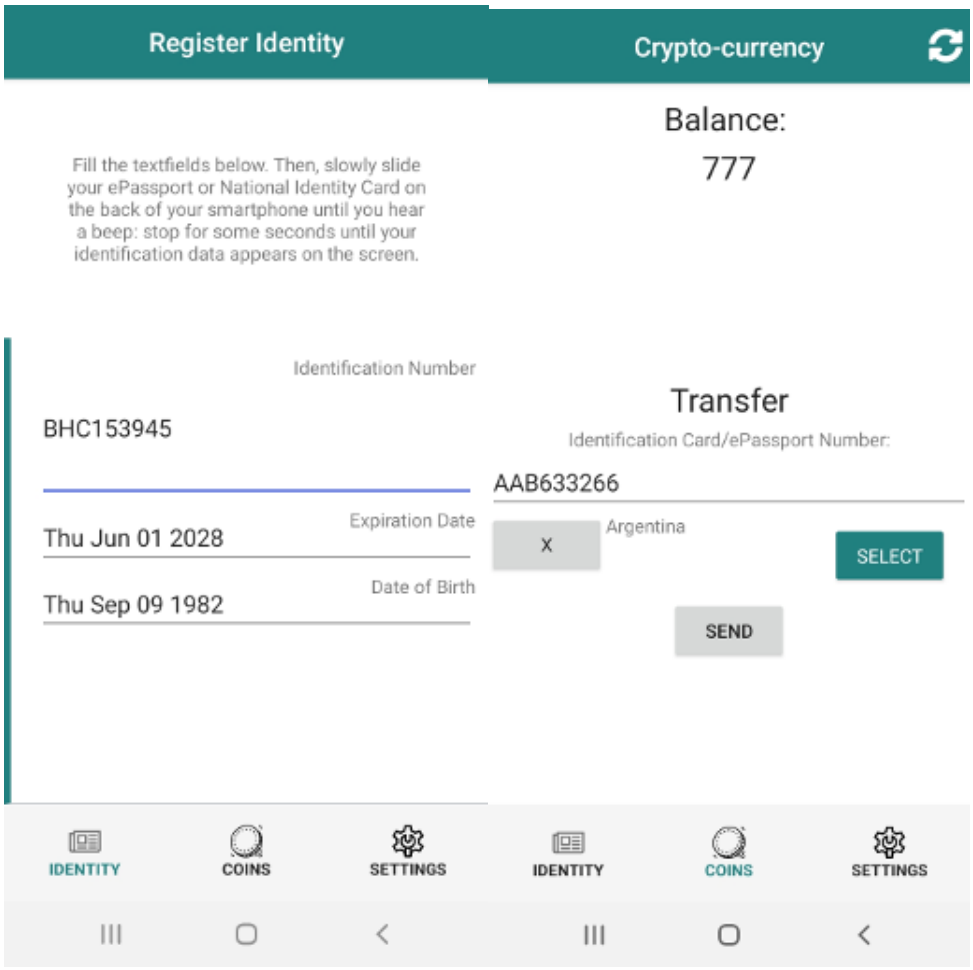


Figure 4.4: Mobile App using Non-Zero-Knowledge Authentication

The use of zero-knowledge techniques to authenticate miners doesn't preclude non-zero-knowledge authentication on the same blockchain: figure 4.4 shows a mobile app[Lim19] using BAC authentication to read a National Identity Card and/or ePassport (left), and then transferring to another ePassport (right). All national identifiers are publicly addressable.

4.3.3 Remote attestation for smartphones

Remote biometric authentication using smartphones (Android/iOS) has been improved with remote attestation [SDH+23] based on Intel SGX Data Center Attestation Primitives:

- much faster and less memory consumption than solutions based on pure zero-knowledge proofs, thus even low-end smartphones are supported: note that pure zero-knowledge solutions may require gigabytes of RAM to verify the full certificate chain, thus excluding low-end smartphones (i.e., inclusion is the primary goal of this authentication solution)
- support for all the complete ciphersuite used in biometric passports/electronic identity cards: the signatures schemes used in blockchains for zero-knowledge proofs (e.g., BLS, Secp256k1) are different from the ones used in government IDs (Brainpool, ED448, ...), thus zero-knowledge implementations do not even exist yet. Worse still, ICAO standards [ICA15a, ICA15b] leave open the use of new elliptic curves described by their parameters, difficulting the implementation of pure zero-knowledge solutions.
- better updatability: new techniques for anti-money laundering and/or sanction screening can be easily implemented by upgrading the server-side enclaves, while pure zero-knowledge solutions are constrained to previous data structures

Note that the use of remote attestation doesn't exclude the possibility of using zero-knowledge proofs: both techniques can be combined together, but relegating zero-knowledge proofs to less demanding cases of selective disclosure of credentials.

4.4 Circumventing the Impossibility of Full Decentralization

Most blockchains using PKIs are consortium blockchains, thus it has become widespread that they always are permissioned and centralised. However, the term "permissionless" literally means "without requiring permission" (i.e., to access, to join, ...), thus a blockchain with a PKI could be permissionless if it accepts any self-signed certificate (i.e., a behaviour conceptually equivalent to Bitcoin), or any certificate from any government in the world as described in the previous subsection 4.3. In the same way, a blockchain using PKIs doesn't imply that its control has become centralised, it means that it accepts identities from said PKIs as described in this paper: actually, decentralization in the blockchain context strictly means that the network and the mining are distributed in a large number of nodes, thus unrelated to authentication.

A recent publication[KLK+19a] proves that it's impossible for blockchains to be fully decentralised without real identity management (e.g., PoW, PoS and DPoS) because they cannot have a positive Sybil cost, defined as the additional cost that a player should pay to run multiple time nodes compared to the total

cost of when those nodes are run by different players. To reflect the level of decentralization, they introduce the following definition:

Definition 2. ((m, ϵ, δ)-Decentralization)[KLK$^+$19a]. For $1 \leq m$, $0 \leq \epsilon$ and $0 \leq \delta \leq 100$, a system is (m, ϵ, δ)-decentralized if it satisfies the following:

1. The size of the set of players running nodes in the consensus protocol, P, is not less than m (i.e., $|P| \geq m$).
2. Define EP_{p_i} as the effective power of player p_i as $\sum_{n_i \in N_{p_i}} \alpha_{n_i}$ where N is the set of all nodes in the consensus protocol and α_{p_i} is the resource power of player p_i. The ratio between the effective power of the richest player, EP_{max}, and the $\delta - th$ percentile player, EP_δ, is less than or equal to $1 + \epsilon$ (i.e., $(EP_{max}/EP_{delta}) \leq 1 + \epsilon$).

Ideally, the number m should be as high as possible (i.e., too many players do not combine into one node); and for the most resourceful and the δ-th percentile player running nodes, the gap between their effective power is small. Therefore, full decentralization is represented by $(m, 0, 0)$ for sufficiently large m.

Definition 3. (Sufficient Conditions for Fully Decentralized Systems)[KLK$^+$19a]. The four following conditions are sufficient to reach (m, ϵ, δ)-decentralization with probability 1.

1. (Give Rewards (GR-m)). Nodes with any resource power earn rewards.
2. (Non-delegation (ND-m)). It is not more profitable for too many players to delegate their resource power to fewer participants than to directly run their own nodes.
3. (No Sybil nodes (NS-δ)). It is not more profitable for a participant with above the δ-th percentile effective power to run multiple nodes than to run one node.
4. (Even Distribution (ED-(ϵ, δ))). The ratio between the resource power of the richest and the δ-th percentile nodes converges in probability to a value less than $1+\epsilon$.

Theorem 4. *For any initial state, a system satisfying GR-m, ND-m, NS-δ, and ED-(ϵ, δ) converges in probability to (m, ϵ, δ)-decentralization. [KLK$^+$19a]*

As it should be obvious by now, a blockchain using zk-PoI with strong identities from trusted public certificates (e.g., national identity cards and/or ePassports 4.3) as described in this paper is the perfect candidate to achieve a fully decentralized blockchain.

Theorem 5. *A blockchain using zk-PoI with strong identities from trusted public certificates (e.g., national identity cards and/or ePassports 4.3) reaches (m, ϵ, δ)-decentralization with probability 1.*

Sketch of Proof. A blockchain using zk-PoI with strong identities from trusted public certificates effectively limits the number of mining nodes to one per individual (ND-m), independently of how resourceful they are (NS-δ, ED-(ϵ, δ)), while keeping membership open to everyone (i.e., achieves a large number of participants (GR-m)). The presence of strong identities allows positive Sybil costs, thus the fulfillment of the sufficient conditions for fully decentralized systems[KLK+19a]. □

Preventing delegation (ND-m) is the most difficult condition to meet:

- market-enforced: richest participants could buy rights-of-use of others' identities, but the market value of said identities (e.g., the Net Present Value of future profits obtained from the exploitation of said identities by their real owners) should wipe away almost all the profits from these exchanges.
- strictly-enforced: miners' software could frequently check for the presence of the physical trusted public certificate (e.g., national identity cards and/or ePassports) and/or require them when transferring funds out of their accounts.

A posterior revision of the paper[KLK+19b] introduces new definitions that try to emphasize that Trusted Third Parties (i.e., Certificate Authorities) shouldn't be used in decentralized blockchains: as described in this paper4.3, using 400 CAs of national identity cards and ePassports from over the world is still being decentralized according to the original definition of decentralization, and certainly much more decentralized than Bitcoin/Ethereum that concentrate >50% of their hashrate in 4-3 entities[GBE+18]. It's very important not to mix concepts and misattribute qualities to different concepts:

- permissionless doesn't imply without identification
- a decentralized consensus protocol doesn't imply that it can't use identification from TTPs: recent consensus protocols decouple Sybil-resistance from the consensus mechanism3.1

Thus, permissionless, decentralized and identification(-less) are different qualities that shouldn't be intermixed. The results of this paper hold in the *trusted* decentralized setting, while the results of [KLK+19b] hold in the *trustless* decentralised setting.

4.5 Resistance against Dark DAOs

Dark DAOs[DKMJ18] appear as a consequence of permissionless blockchains where users can create their own multiple identities and there's no attributability of the actions.

1. When using real-world identities, it's possible to establish the identity of the parties running the Dark DAO that are committing frauds (attributability) or at least, their pseudonyms: therefore, it's possible to punish them.

2. To prevent Dark DAOs buying real-world identities, a smart contract can be setup that pays a reward for denouncing the promoters of the fraud: the whistleblowers would be paid a multiple of what they would get from the defrauders, thus making denunciation the preferred option. Then defrauders would be banned as in step 1.

4.6 Resistance against Collusion and other Attacks

In this sub-section, we consider different avenues for attack and provide detailed defense mechanisms:

1. Corrupt root certificate authorities
2. Attacks against consensus protocols
3. Resistance against collusion
4. On achieving collusion-freeness

4.6.1 Corrupt Root Certificate Authorities

Corrupt countries may be tempted to create fake identities or frequently renovate existing ones: these countries can be easily banned out by removing them from the list of valid authorities (i.e., root X.509 certificate and/or Country Signing Certificate). Bounties in cryptocurrency could be offered for whistleblowing any corrupt attack against the long-term existence of the blockchain.

4.6.2 Attacks against Consensus Protocols

Modern consensus protocols based on the cryptographically secure random choice of the leader (e.g., [HMW18, KKJG$^+$17]) detect cheating by monitoring changes to the chain quality. The following table gathers cheating events for different consensus algorithms that could be detected and punished:

Protocol	Cheater detection
Dfinity[HMW18]	Equivocation: multiple blocks for same round with same rank.
	Equivocation: multiple blocks with the highest priority.
	All the blocks must be timely published.
	All the notarizations must be timely published within one round.
OmniLedger[KKJG$^+$17, KK19]	Core validators can detect rogue validators.
	Withholders can be detected after multiple consecutive rounds.
	5>= failed RandHound views from a rogue validator.

4.6.3 Resistance against Collusion

Consensus protocols already provide collusion-tolerance by design: an adversary controlling a high number of nodes, or equivalently all said nodes colluding for the same attack, must confront the difficulties introduced by shard re-assignment at the beginning of every new epoch. For the case of OmniLedger[KKJG+17], the security of the validator assignment mechanism can be modeled as a random sampling problem using the binomial distribution,

$$P\left[X \leq \left\lfloor \frac{n}{3} \right\rfloor\right] = \sum_{k=0}^{n} \left(\begin{array}{c} n \\ k \end{array} \right) m^k \left(1-m\right)^{n-k}$$

assuming that each shard has less than $\left\lfloor \frac{n}{3} \right\rfloor$ malicious validators. Then, the failure rate of an epoch is the union bound of the failures rates of individual shards, each one calculated as the cumulative distribution over the shard size n, with X being the random variable that represents the number of times we pick a malicious node. An upper bound of the epoch failure event, X_E, is calculated as:

$$P\left[X_E\right] \leq \sum_{k=0}^{l} \frac{1}{4^k} \cdot n \cdot P\left[X_S\right]$$

where l is the number of consecutive views the adversary controls, n is the number of shards and $P\left[X_S\right]$ is the failure rate of one individual shard. Finally, for $l \to \infty$, we obtain

$$P\left[X_E\right] \leq \frac{4}{3} \cdot n \cdot P\left[X_S\right]$$

4.6.4 On Achieving Collusion-Freeness

Start noticing that collusion-freeness is not about preventing malicious behaviour, only preventing that malicious players act as independently of each other as possible. Following a previous seminal work[LMas05], collusion-freeness can only be obtained under very stringent conditions: (a) the game must be finite; (b) the game must be publicly observable; and (c) the use of private channels at the beginning of the game is essential, but forbidden during the execution of the protocol. Although blockchains are publicly observable, they are also an infinite game where parties can freely communicate between them using private channels at any time: therefore, collusion-freeness is impossible in the sense of [LMas05].

Fortunately, there is a way to get around this impossibility result: forbid malicious/Byzantine behaviours requiring the use of mutual attestation for all the nodes, thus precluding any deviation from the original protocol.

Conjecture 6. *If mutual attestation is required for all the nodes, any infinite, partial-information blockchain game with publicly observable actions has a collusion-free protocol.*

As mutual attestation is already required for zk-PoI 4.2, we would only be extending its use for the rest of the blockchain protocol.

5 Incentive Compatible and Strictly Dominant Cryptocurrencies

The success of cryptocurrencies is better explained by their incentive mechanisms rather than their consensus algorithms: a cryptocurrency with poor incentives (e.g., a cryptocurrency not awarding coins to miners) will not achieve any success; conversely, better incentives and much more inefficient consensus algorithm could still find some success.

Much research has been focused on conceiving better consensus algorithms for decentralised systems and cryptocurrencies[PS16, DPS16, KKJG+17, GHM+17, KJG+16, HMW18]: unfortunately, obtaining consensus mechanisms with better incentives and economic properties is an area that is lacking much research, and the combination of all the game-theoretic results contained in this section fills this gap for the sake of achieving a *focal point* (i.e., Schelling point[Sch60]) in the multi-equilibria market of cryptocurrencies. Thus, a selective advantage is introduced by design over all the other cryptocurrencies, in explicit violation of the neutral model of evolution[EAK+17] in order to obtain an incentive compatible and strictly dominant cryptocurrency.

5.1 Incentive-Compatible Cryptocurrency

Shard-based consensus protocols have been recently introduced in order to increase the scalability and transaction throughput of public permissionless blockchains: however, the study of the strategic behaviour of rational miners within shard-based blockchains is very recent. Unlike Bitcoin, that grants all rewards to the most recent miner, block rewards and transactions fees must be proportionally shared between all the members of the sharding committee[KJG+16], and this includes incentives to remain live during all the lifecycle of the consensus protocol. Even so, existing sharding proposals[KKJG+17, ZMR18] remain silent on how miners will be rewarded to enforce their good behaviour: as it's evident, if all miners are equally rewarded without detailed consideration of their efforts, rational players will *free-ride* on the efforts of others.

One significant difference introduced in this paper with respect to other shard-based consensus protocols is the use of Zero-Knowledge Proof-of-Identity as the Sybil-resistance mechanism: as we will see in the following sections, it's a significant novelty because solving Proof-of-Work puzzles is the most computationally expensive activity of consensus protocols, thus it's no longer dominated by computational costs. This makes the necessity for an incentive-compatible protocol even more acute: the preferred rational miner's strategy is to execute the Proof-of-Work of the initial phase of the protocol for each epoch and to refrain from the transaction verification and consensus of subsequent phases of the protocol, but still selfishly claim the rewards as if they had participated. The substitution of costly PoW for cheap Zero-Knowledge Proof-of-Identity only increases the attractiveness of this rational strategy, that can only be counteracted by using an incentive-compatible protocol.

5.1.1 A Nash Equilibrium for a Cryptocurrency on a Shard-Based Blockchain

This section is based on a stylised version of a recent game-theoretic model[MJMF18], taking into consideration that there is no cost associated with committee formation to enter each shard since we are using Zero-Knowledge Proof-of-Identity, and not costly Proof-of-Work: instead, a penalty p is imposed to defective and/or cheating miners. The following is a list of symbols:

Symbol	Definition
k	Number of shards
N	Number of miners
x_i^j	Vector of received transactions by miner i in shard j
y^j	Vector of transactions submitted by shard j to blockchain
c	Minimum number of miners in each committee
τ	Required number of miners in shard for consensus
r	Benefit for each transaction
b_i	Benefit of miner i after adding the block
c_i^t	Total cost of computation for miner i
c^o	Total optional costs in each epoch
c^v	Cost of transaction verification
c^f	Fixed costs in optional cost
p	Penalty cost
BR	Block Reward
l_j	Number of cooperative miners in each shard
L	Total number of cooperative miners in all shards
$C_j^{l_j}$	Set of all cooperative miners in shard j
$D_j^{n-l_j}$	Set of all defective miners in shard j
C^L	Set of all cooperative miners
D^{N-L}	Set of all defective miners
s^r	Signed receipt of a transaction

Let $\mathbb{G}$ denote the shard-based blockchain game, defined as a triplet (P, S, U) where $P = \{P_i\}_{i=1}^N$ is the set of players, $S = \{C, D\}$ is the set of strategies (Cooperate C, or Defect D) and U is the set of payoff values. Each miner receives a reward if and only it has already cooperated with other miners within the shard, the payoff of cooperative miners in set C^{l_j} is

$$u_i(C) = \frac{BR}{kl_j} + \frac{r\left|y^j\right|}{l_j} - \left(c^f + \left|x_i^j\right| c^v\right) \tag{5.1}$$

We assume that the block reward BR is uniformly distributed among shards and each cooperative miner can receive a share of it. A miner might be cooperative but all other miners may agree on a vector of transactions y^j that is different from his own vector of transactions x_i^j (i.e., $\left|x_i^j\right| \neq \left|y^j\right|$): nonetheless, transaction rewards are uniformly distributed among all cooperative miners in each shard, proportional to all the transactions submitted to the blockchain by each shard.

The defective miners' payoff can be calculated as

$$u_i^D = -p^m$$

because the defective miners will have to pay a penalty and they will not receive any benefit (and it doesn't incur in any mandatory cost such as solving PoW puzzles because we use cheap Zero-Knowledge Proof-of-Identity).

There exists a cooperative Nash equilibrium profile in game $\mathbb{G}$ under the following conditions:

Theorem 7. *Let $C_j^{l_j}$ and $D_j^{m-l_j}$ denote the sets of l_j cooperating miners and $n-l_j$ defecting miners inside each shard j with n miners, respectively. $\left(C^L, D^{N-L}\right)$ represents a Nash equilibrium profile in each epoch of game $\mathbb{G}$, if the following conditions are satisfied:*

1. *In all shards j, $l_j \geq \tau$.*
2. *If for a given miner P_i in shard j, with $x_i^j = y^j$, then the number of transactions $\left|x_i^j\right|$ must be greater than*

$$\theta_c^1 = \frac{c^f - \frac{BR}{kl_j} + p}{r/l_j - c^v}$$

3. *If for a given miner P_i in shard j, with $x_i^j \neq y^j$, then the number of transactions $\left|x_i^j\right|$must be smaller than*

$$\theta_c^2 = \frac{\frac{BR}{kl_j} + \frac{r\left|y^j\right|}{l_j} - c^f - p}{c^v}$$

Proof. The first condition $l_j \geq \tau$ (i.e., the number of cooperative miners must be greater than τ) must hold so that cooperative miners will receive benefits for transactions and block rewards.

Let l_j^* be the largest set of cooperative miners in each shard, where no miner in $D_j^{n-l_j}$ can join $C_j^{l_j}$ to increase its payoff: if miner P_i^j is among the set of cooperative miners where $x_i^j = y^j$, then it would not unilaterally deviate from cooperation if:

$$\frac{BR}{kl_j} + \frac{r\left|x_i^j\right|}{l_j} - \left(c^f + \left|x_i^j\right| c^v\right) \geq -p$$

which shows that $x_i^j \geq \theta_c^1$, whereas in the second condition,

$$\theta_c^1 = \frac{c^f - \frac{BR}{kl_j} - p}{r/l_j - c^v}$$

But if P_i^j is among the cooperators whose vector of transactions is different from the output of the shard, $x_i^j \neq y^j$, then it would not deviate from cooperation if:

$$\frac{BR}{kl_j} + \frac{r\left|y^j\right|}{l_j} - \left(c^f + \left|x_i^j\right| c^v\right) \geq -p$$

which shows that $x_i^j < \theta_c^2$, whereas in the third condition,

$$\theta_c^2 = \frac{\frac{BR}{kl_j} + \frac{r\left|y^j\right|}{l_j} - c^f - p}{c^v}$$

Then if l_j^* represents the largest set of cooperative miners in each shard, then $\left(C^L, D^{N-L}\right)$ would be the unique cooperative Nash equilibrium of the game $\mathbb{G}$. □

As can be understood from the proof, cooperative miners have less incentive to cooperate when: 1) the number of participants N increases; 2) the optional costs of computation increase (c^f is in the numerator or c^v in denominator of θ_C); 3) or in general, when the number of transactions is not large enough compared to a fixed threshold.

5.1.2 Incentive-Compatible Cryptocurrency on a Shard-Based Coordinated Blockchain

In order to increase the incentives to cooperate rather than defect, an incentive-compatible protocol enforcing cooperation based on the previously presented Nash equilibrium is introduced here 1: all miners should disclose their list of transactions to a coordinator, who then announces to each miner whether their cooperation would be in their interests based on being within the maximum subset of miners with a similar list of transactions (i.e., $x_i^j = y^j$), and then enforces their cooperation by checking their compliance and rewarding them properly.

The protocol proceeds as follows: for the first function (i.e., *ShardTransactionsAssignment*), each miner receives a list of transactions x_i^j to verify based on the epochRandomness and his pseudonymous identity and public key obtained by the Zero-Knowledge Proof-of-Identity.

For the second function (i.e., *NodeSelection*), all miners calculate a hash $H\left(x_i^j\right)$ over their transaction list and send it to the coordinator. The coordinator finds the subset with the maximum number of miners with a common transaction list, thus calculating θ_c^1, θ_c^2, l_j and $C_j^{l_j}$: in each epoch, the coordinator publicly defines the list of cooperative miners $C_j^{l_j}$ and defective miners $D_j^{n-l_j}$ using on theorem 7.

For the third function (i.e., *ShardParticipation*), all the transactions of each miner are verified and a signed consensus is reached.

function ShardTransactionsAssignment {
  $Shard \leftarrow GetShard(epochRandomness, pseudonym, PK)$
  $x_i \leftarrow ShardTransactions(Shard)$
}
function NodeSelection {
  P_i send $H\left(x_i^j\right)$ to Coordinator

  if (PresentNode() == Coordinator) {

    Receive all $H\left(x_i^j\right)$

    $l_j \leftarrow$Max number of miners with common txs. from list of $H\left(x_i^j\right)$
    if ($l_j < \tau$)
      return "All Defective"
    else {
      $C_j^{l_j}$= list of $l_j$miners
      Calculate θ_c^1 and θ_c^2 from theorem 7
      return θ_c^1, θ_c^2 and $C_j^{l_j}$
    }
  }
}
function ShardParticipation {
  if ($P_i \in C_j^{l_j}$ and $\left|x_i^j\right| \leq \theta_c^1$)
    return Defect
  else if ($P_i \notin C_j^{l_j}$ and $\left|x_i^j\right| \geq \theta_c^2$)
    return Defect
  Verify transactions
  y^j=set of verified transactions by remaining cooperative P_i
  Consensus on verified transactions
  Sign BFT agreement result
  return signature, agreed block's header
}
function VerificationAndRewards {
  Verify cooperation of $P_i \in C^L$ for each shard
  Send rewards to cooperative P_i using equation (5.1)
}

Algorithm 1: Incentive-Compatible Protocol on a Coordinated Shard-Based Blockchain

For the fourth function (i.e., *VerificationAndRewards*), the rewards are shared between the cooperative miners and denied to those miners in $C_j^{l_j}$ that didn't cooperate.

5.1.3 Improved Incentive-Compatible Cryptocurrency on a Shard-Based Blockchain

Although the role of a coordinator is essential to BFT protocols[KJG+16], its expanded functionality in the previous incentive-compatible protocol 1 is problematic: it introduces latency and network costs due to the new obligations to report to the coordinator; moreover, it creates new opportunities for malicious miners which may report false $H\left(x_i^j\right)$ or not follow coordinator's instruction to cooperate or defect. The next incentive-compatible protocol significantly improves over the state of the art: the role of the coordinator is minimised, strengthing the protocol by removing the previous vulnerabilities and making it resistant to malicious miners.

Information propagation[DW18] is an essential part of any blockchain, and gossiping transactions to neighbouring miners is one of its key features. In the new incentive-protocol protocol, we require that any broadcasted/gossiped/propagated transaction gets acknowledged with a signed receipt to its sender: then, senders must attach these receipts to the consensus leaders and verification nodes in order to ease detection of defective and/or cheating miners. Miners who were gossiped transactions but didn't participate are considered defective, and not rewarded. In other words, the signed receipts serve as snitches that denounce non-cooperative miners thus preventing that any reward gets assigned to them: at the same time, all miners are incentivised to participate in the denunciation in order to gain the rewards of non-cooperative miners and other *free-riders*.

In order to save bandwidth, note that it's not obligatory to send the full list of all signed transaction receipts to consensus leaders and/or verification nodes: only a random subset per each miner should be enough to catch defective miners.

Additionally, the absence of signed receipts could be used to detect the need of a change of a consensus leader (i.e., "view-change") in BFT protocols[KJG+16, KKJG+17].

5.2 On Strictly Dominant Cryptocurrencies

A cryptocurrency using Zero-Knowledge Proof-of-Identity as the Sybil-resistance mechanism strictly dominates PoW/PoS cryptocurrencies: a miner having to choose between mining different cryptocurrencies, one with no costs associated with its Sybil-resistance mechanism and distributing equally the rewards, and the others using costly PoW/PoS and thus featuring mining concentration, will always choose the first one. That is, mining equally distributed cryptocurrencies using Zero-Knowledge Proof-of-Identity is a dominant strategy; in other words, the strategy of mining Bitcoin and other similar cryptocurrencies is strictly dominated by the hereby described cryptocurrency. *Ceteris paribus*, this cryptocurrency will have better network effects, thus better long-term valuation.

```
function ShardTransactionsAssignment {
   Shard ← GetShard(epochRandomness, pseudonym, PK)
   x_i ← ShardTransactions(Shard)
}
function GossipTransaction {
   GossipTransaction()
   s^r = AcknowledgeTransmission()
   Store s^r
}
function ReceiveTransaction {
   tx=ReceiveTransaction()
   ReplyTransaction(sign(hash(tx)))
}
function ShardParticipation {
   Verify transactions
   Collect lists of s^r for every P_i
   y^j=set of verified transactions by remaining cooperative P_i
   Consensus on verified transactions
   Sign BFT agreement result
   return signature, agreed block's header
}
function VerificationAndRewards {
   Verify cooperation of P_i ∈ C^L using lists of s^r
   Send rewards to cooperative P_i using equation (5.1)
}
```

Algorithm 2: Improved Incentive-Compatible Protocol on a Shard-Based Blockchain

5.2.1 Strictly Dominant Cryptocurrencies and a Nash Equilibrium

The intuition behind the preference to mine fully decentralised cryptocurrencies with the lowest expenditure (i.e., lowest CAPEX/OPEX implies higher profitability), thus the search for better hash functions[CLC17, ACP+16, BDK15, BCGS16], is formally proved here and then applied to the specific case of the proposed cryptocurrency.

Definition 8. (Power-Law Fee-Concentrated (PLFC) cryptocurrency). A cryptocurrency whose distribution of mining and/or transaction fees follows a power-law (i.e., a few entities earn most of the fees/rewards), usually due to the high costs of its Sybil-resistance mechanism.

Example 9. Proof-of-Work cryptocurrencies are Power-Law Fee-Concentrated: 90% of the mining power is concentrated in 16 miners in Bitcoin and 11 in Ethereum[GBE+18].

Proof-of-Stake cryptocurrencies are Power-Law Fee-Concentrated: miners earn fees proportional to the amount of money at stake, and wealth is Pareto-

concentrated[Par14].

Definition 10. (Uniformly-Distributed Capital-Efficient (UDCE) cryptocurrency). A cryptocurrency whose distribution of mining and/or transaction fees is uniformly distributed among all the transaction processing nodes, and doesn't require significant investments from the participating miners.

Example 11. The proposed cryptocurrency using Zero-Knowledge Proof-of-Identity is a Uniformly-Distributed Capital-Efficient cryptocurrency.

Definition 12. (Game of Rational Mining of Cryptocurrencies). A rational miner ranks the cryptocurrencies according to their expected mining difficulty, and chooses to mine those with lowest expected difficulty.

Example 13. Awesome Miner[Min18a], MinerGate[Min18b], MultiMiner[Mul18a], MultiPoolMiner[Mul18b], Smart-Miner[SM19, CBGL19] and NiceHash Miner[Nic18] are practical implementations of the Game of Rational Mining of Cryptocurrencies (although also considering their prices in addition to their difficulties). Specific calculators for mining profitability[Wha19, Rub19, Coi19, Cry19a, Cry19b, CC19] could also be used for the similar purposes. Additionally, other papers[BLT19] provide models regarding optimal hash rate allocation.

Let u_i be the payoff or utility function for each miner, expressing his payoff in terms of the decisions or strategies s_i of all the miners,

$$u_i\left(s_1, s_2, \ldots, s_n\right) = u_i\left(s_i, s_{-i}\right)$$

whese s_{-i} are set of the strategies of the rest of miners,

$$s_{-i} = \left(s_1, s_2, \ldots, s_{i-1}, s_{i+1}, \ldots, s_n\right)$$

Definition 14. A strategy s_1 *strictly dominates a strategy* s_2 *for miner* i if and only if, for any s_{-i} that miner i's opponents might use,

$$u_i\left(s_1, s_{-i}\right) > u_i\left(s_2, s_{-i}\right)$$

That is, no matter what the other miners do, playing s_1 is strictly better than playing s_2 for miner i. Conversely, we say that the strategy s_2 *is strictly dominated by* s_1: a rational miner i would never play a strictly dominated strategy.

Definition 15. A strategy s_i^* is a *strictly dominant strategy for miner* i if and only if, for any profile of opponent strategies s_{-i} and any other strategy $s_i^{'}$ that miner i could choose,

$$u_i\left(s_i^*, s_{-i}\right) > u_i\left(s_i^{'}, s_{-i}\right)$$

We now demonstrate that mining *UDCE crypto-cryptocurrencies* 10 is a strictly dominant strategy with regard to *PLFC cryptocurrencies 8* in the *Game of Rational Mining of Cryptocurrencies* 12 by showing that every miner's expected profitability is higher in UDCE cryptocurrencies.

Theorem 16. *UDCE cryptocurrencies yield a strictly higher miner's expected profitability compared to PLFC cryptocurrencies in the Game of Rational Mining of Cryptocurrencies.*

Proof. Let N be the number of miners and R the average daily minted reward per day: UDCE cryptocurrencies award an average of N/R units of cryptocurrency to every participant miner. For every miner on the long tail of the power distribution, the amount earned with UDCE cryptocurrencies is obviously higher than with PLFC cryptocurrencies. For the few miners that dominate PLFC cryptocurrencies, their profitability is lower because they have to account for the energy[KT18, SKG19] and equipment costs in the case of PoW cryptocurrencies or the opportunity cost of staking capital in volatile PoS cryptocurrencies[Dia19], meanwhile in UDCE cryptocurrencies their cost of mining is so negligible compared to PLFC cryptocurrencies that the balance of profitability is always tipped in their favour. □

Definition 17. The process to solve games called *Iterated Deletion of Strictly Dominated Strategies (IDSDS)* is defined by the next steps:

1. For each player, eliminate all strictly dominated strategies.
2. If any strategy was deleted during Step 1, repeat Step 1. Otherwise, stop.

If the process eliminates all but one unique strategy profile s^*, we say it is the *outcome of iterated deletion of strictly dominated strategies* or a *dominant strategy equilibrium.*

Definition 18. A strategy profile $s^* = (s_1^*, s_2^*, \ldots, s_n^*)$ is a *Pure-Strategy Nash Equilibrium* (PSNE) if, for every player i and any other strategy $s_i^{'}$ that player i could choose,

$$u_i\left(s_i^*, s_{-i}^*\right) \geq u_i\left(s_i^{'}, s_{-i}^*\right)$$

Definition 19. A strategy profile $s^* = (s_1^*, s_2^*, \ldots, s_n^*)$ is a *Strict Nash Equilibrium* (SNE) if, for every player i and any other strategy $s_i^{'}$ that player i could choose,

$$u_i\left(s_i^*, s_{-i}^*\right) > u_i\left(s_i^{'}, s_{-i}^*\right)$$

Additionally, if a game is solvable by Iterated Deletion of Strictly Dominated Strategies, the outcome is a Nash equilibrium.

Theorem 20. *A UDCE cryptocurrency dominating PLFC cryptocurrencies is a Nash equilibrium.*

Proof. Mining a UDCE cryptocurrency is a strictly dominant strategy with regard to other miners mining PLFC cryptocurrencies because PLFC cryptocurrencies are strictly dominated by UCDE cryptocurrencies by Theorem 16, thus a rational miner will always prefer to miner the latter.

Thus, by the application of Iterated Deletion of Strictly Dominated Strategies (IDSDS) to the Game of Rational Mining of Cryptocurrencies 12, each miner will eliminate mining PLFC cryptocurrencies in favor of mining an UDCE cryptocurrency, leaving this as the unique outcome: therefore, mining a UDCE cryptocurrency is a *dominant strategy equilibrium* by Definition 17 and a *Nash equilibrium* by Definition 18 or by Definition 19. □

Claim 21. (Uniqueness of Technical Solution). The proposed technical solution using Zero-Knowledge Proof of Identity from trusted public certificates (i.e., national identity cards and/or ePassports) is the only practical and unique solution for a UCDE cryptocurrency.

Proof. As demonstrated in the paper describing "The Sybil Attack"[Dou02], Sybil attacks are always possible unless a trusted identification agency certifies identities.

As National Identity Cards and ePassports are the only globally available source of trusted cryptographic identity (3.5B for National Identity Cards and 1B for ePassports), the only way to bootstrap a UCDE cryptocurrency is by using the proposed Zero-Knowledge Proof-of-Identity from trusted public certificates (National Identity Cards and/or ePassports). □

5.2.2 Strictly Dominant Cryptocurrencies and Evolutionary Stable Strategies

Another interesting viewpoint to consider in the analysis of the cryptocurrency market is the one offered by behavioural ecology and its Evolutionary Stable Strategies 22: each cryptocurrency can be considered a unique individual in a population, genetically programmed to play a pre-defined strategy. New cryptocurrencies are introduced into the population as individuals with different mutations that define their technical features (e.g., forking the code of a cryptocurrency to change the hashing algorithm, or a zk-PoI cryptocurrency). An Evolutionary Stable Strategy 22 is a strategy that cannot be invaded by any alternative strategy, that is, it can resist to the invasion of a mutant and it's impenetrable to them: once it's introduced and becomes dominant in a population, natural selection is sufficient to prevent invasions from new mutant strategies.

Definition 22. The pure strategy s^* is an Evolutionary Stable Strategy[Joh73] if there exists $\epsilon_0 > 0$ such that:

$$(1-\epsilon)\left(u\left(s^*, s^*\right)\right) + \epsilon\left(u\left(s^*, s^{'}\right)\right) > (1-\epsilon)\left(u\left(s^{'}, s^*\right)\right) + \epsilon\left(u\left(s^{'}, s^{'}\right)\right)$$

for all possible deviations $s^{'}$and for all mutation sizes $\epsilon < \epsilon_0$. There are two conditions for a strategy s^* to be an Evolutionary Stable Strategy: for all $s^* \neq s^{'}$ either

1. $u\left(s^*, s^*\right) > u\left(s^{'}, s^*\right)$, that is, it's a Strict Nash Equilibrium 19, **or**,

2. if $u(s^*, s^*) = u\left(s^{'}, s^*\right)$ then $u\left(s^*, s^{'}\right) > u\left(s^{'}, s^{'}\right)$

Theorem 23. *Mining a UDCE cryptocurrency is an Evolutionary Stable Strategy.*

Proof. Since mining a UCDE cryptocurrency is a strictly-dominant strategy and a Strict Nash Equilibrium 20, then it is an Evolutionary Stable Strategy because it fulfills its first condition 22.

Additionally, mining a UCDE cryptocurrency based on the global network of National Identity Cards and ePassports is an Evolutionary Stable Strategy over national variants/mutants due to Claim 21. □

Thus, the Game of Rational Mining of Cryptocurrencies 12 is a "survival of the fittest" ecology, where the cheapest cryptocurrency to mine offering the higher profits rises above the others.

5.2.3 Obviating the Price of Crypto-Anarchy

The most cost efficient Sybil-resistant mechanism is the one provided by a trusted PKI infrastructure[Dou02] and a centralised social planner would prefer the use of National Identity Cards and/or ePassports in order to minimise costs: instead, permissionless blockchains are paying very high costs by using PoW/PoS as Sybil-resistant mechanisms. In this paper, Zero-Knowledge Proof-of-Identity is introduced as a compromise solution between both approaches, thus obtaining a very efficient Sybil-resistant mechanism with the best of both worlds.

In order to measure how the efficiency of a Sybil-resistant mechanism degrades due to the selfish behaviour of its agents (i.e., a fixed amount of block reward to be distributed among a growing and unbounded number of miners paying high energy costs, as in Bitcoin), we compare the ratio between the worst Nash equilibrium and the optimal centralised solution, a concept known as Price of Anarchy in game theory because it bounds and quantifies the costs of the selfish behavior of the agents.

Definition 24. *The Price of Anarchy*[KP99]. Consider a game $G = (N, S, u)$ defined as a set of players N, strategy sets S_i for each player and utilities $u_i : S \to \mathbb{R}$ (where $S = S_1 x \ldots x S_n$ are also called the set of outcomes). Define a measure of efficiency of each outcome that we want to minimise, $Cost : S \to \mathbb{R}$, and let $Equil \subseteq S$ be the set of strategies in Nash equilibria. The *Price of Anarchy* is given by the following ratio:

$$\text{Price of Anarchy} = \frac{\max_{s \in Equil} Cost(s)}{\min_{s \in S} Cost(s)}$$

The competition game between several blockchains and their cryptocurrencies can be reformulated[ARMM+18] as a congestion game[Ros73, DM96] (hereby included for completeness), more amenable to the formulations commonly used for analyzing the Price of Anarchy (the necessity for the following definitions is already intuited in the Discussion of [AH19]): as the number of miners increases,

it also exponentially decreases the chance that a given miner wins the block reward by being the first to solve the mining puzzle (i.e., the system becomes increasingly congested); it has been proved that free entry is solely responsible for determining the resource usage[MGT18], and that the difficulty is not an instrument that can regulate it.

Miners, mining servers and crypto-currencies Denote by $\mathcal{N} \coloneqq \{1, 2, \ldots, N\}$ the finite set of miners that alter the utilities of other miners if any of them change strategies and let $\mathcal{K} \coloneqq \{1, 2, \ldots, K\}$ be the set of cryptocurrencies, each associated to exactly one puzzle that miners are trying to solve. Let $\mathcal{M} \coloneqq \{1, 2, \ldots, M\}$denote the set of Edge computing Service Providers (ESPs), or mining servers used to offload the costly computational processing.

Strategies Let $\mathcal{S}_i \subset \mathcal{K}x\mathcal{M}$ denote the set of ordered pairs (cryptocurrency, ESP) corresponding to ESPs that miner i can rely on to solve the puzzles of a given cryptocurrency. A strategy for miner i is denoted by $s_i \in \mathcal{S}_i$ corresponding to the cryptocurrency (puzzle) which a miner intends to solve using a given infrastructure. A strategy vector $s \coloneqq (s_i)_{i \in \mathcal{N}}$ produces a load vector $l \coloneqq (l_{k,m})_{k,m}$, where $l_{k,m}$ denotes the number of users mining blockchain k at ESP m.

Rewards, costs, and utilities Let η_k be the load of miners across all ESPs towards cryptocurrency k. Then,

$$\eta_k \coloneqq \sum_{m' \in \mathcal{M}} l_{k,m'} \mu_{k,m'}$$

For a given load vector l, the time to solve the puzzle of the k^{th} cryptocurrency is exponentially distributed with expectation $1/\eta_k$. Let q_k be the probability that puzzle k is solved by time T,

$$q_k = 1 - \exp\left(-T\eta_k\right)$$

The probability that a given miner using ESP m is the first to solve puzzle k is

$$p_{k,m} = 1_{l_{k,m>0}} q_k \mu_{k,m} / \eta_k$$

where 1_c equals 1 if condition c holds and 0 otherwise. For simplification, subscript m can be dropped and we consider a single ESP. Then, the probability that a miner is the first to solve the puzzle is

$$p_k\left(l_k\right) = \left(1 - \exp\left(-T\mu_k l_k\right)\right) / l_k$$

Let $U_{k,m}\left(l\right)$ denote the utility to a miner who tries to find the solution of the current puzzle associated to cryptocurrency k using ESP m and $\gamma_{k,m}$ denote the cost of mining blockchain k at ESP m:

$$U_{k,m}\left(l\right) = \begin{cases} p_{k,m} - \gamma_{k,m} & \text{if } p_{k,m} > \gamma_{k,m}, \\ 0 & \text{otherwise} \end{cases}$$

and the utility of a tagged miner to mine a cryptocurrency k when there are l_k miners associated with the same cryptocurrency is

$$U_k\left(l_k\right) = p_k - \gamma_k, \text{ if } p_k - \gamma_k \geq 0$$

Theorem 25. *[ARMM$^+$18]If for all i and j, $S_i = S_j$ and s_i does not depend on i, then the Nash equilibrium is given by the solution of the following optimization problem,*

$$\begin{aligned} argmin_s \Phi\left(s\right) \quad &:= \quad \sum_{k\in\mathcal{K}} \sum_{l=1}^{l_k} p_k\left(l\right) - \gamma_k \\ \text{subject to:} \qquad & \sum_{k\in\mathcal{K}} l_k \leq N, l_k \geq 0 \end{aligned}$$

Definition 26. Let $NashCongestedEquil \subseteq S$ be the set of strategies given as solution of the optimization problem of Theorem 25, then the *Price of Crypto-Anarchy* is given by the following ratio:

$$\text{Price of Crypto-Anarchy} = \frac{\max_{s\in NashCongestedEquil} Cost\left(s\right)}{Cost\left(\text{zk-PoI}\right)}$$

In practice, the real-world costs of the Zero-Knowledge Proof of Identity can be considered almost zero because it's subsidised by governments and thus exogenous to any blockchain system. Quite the opposite, the energy costs of PoW cryptocurrencies are notoriously high[KT18, SKG19]: it is estimated that mining Bitcoin, Ethereum, Litecoin and Monero consumed an average of 17, 7, 7 and 14 MJ to generate one US$, respectively; and that Bitcoin causes around 22 megatons in CO2 emissions annually[SKG19].

The trivial extension to Proof-of-Stake is left as an exercise to the reader, although it's not as affordable as it may be seen: as of March 2019, an average of 40% of the cryptocurrency supply is staked at a total of $4Bn between all PoS blockchains[Dia19]. Actually, Proof-of-Stake is not strictly better than Proof-of-Work as the distribution of the market shares between both technologies has been shown to be indistinguishable (Appendix 3, [EAK$^+$17]).

5.2.4 Pareto Dominance on Currency Circulation

For completeness, a stylised version of a model of competing currencies[FVS16] is introduced here to prove that UDCE cryptocurrencies also dominate PLFC cryptocurrencies on their circulation (i.e., trading, speculating) due to their stronger network effects, and not only mining as previously proved. The key observation here is that by definition 8, the returns of mining PLFC cryptocurrencies is concentrated on a very limited number of miners and the newly minted cryptocurrency could be held for long periods of times: otherwise, if they didn't expect that the held cryptocurrencies would appreciate in time, they would be mining another set of cryptocurrencies with better expectations. In direct contrast, the distribution of mining and/or transaction fees of UDCE

cryptocurrencies is uniformly distributed by definition 10: therefore, the returns of the holding strategy after minting them would be lower and their subsequent circulation much less restricted.

Suppose an economy divided into periods, each period divided into two subperiods: in the first subperiod, a perishable good demanded by everyone is produced and consumed in a Centralised Market; in the second subperiod, buyers who only consume are randomly matched with sellers who only produce with probability $\sigma \in (0,1)$ in a Decentralised Market. Let $\beta \in (0,1)$ denote the discount factor, $\phi_t^i \in \mathbb{R}_+$ denote the value of a unit of currency $i \in \{1,\ldots,N\}$ in terms of the CM food and $\phi_t = \left(\phi_t^1,\ldots,\phi_t^N\right) \in \mathbb{R}_+^N$ denote the vector of real prices.

Definition 27. (Buyers). In a $[0,1]$-continuum of buyers, $x_t^b \in \mathbb{R}$ denotes the buyer's net consumption of the CM good and $q_t \in \mathbb{R}_+$denotes the consumption of the DM good. The utility function of the buyer's preferences is given by

$$U^b\left(x_t^b, q_t\right) = x_t^b + u\left(q_t\right)$$

with $u : \mathbb{R}_+ \to \mathbb{R}$ continuously differentiable, increasing and strictly concave with $u'(0) = \infty$ and $u(0) = 0$.

Let $W^b\left(M_{t-1}^b, t\right)$ denote the value function for a buyer who starts period t holding a portfolio $M_{t-1}^b \in \mathbb{R}_+^N$ of cryptocurrencies in the CM and let $V^b\left(M_t^b, t\right)$ denote the value function in the DM: the dynamic programming equation is

$$W^b\left(M_{t-1}^b, t\right) = \max_{\left(x_t^b, M_t^b\right) \in \mathbb{R}\times\mathbb{R}_+^N} \left[x_t^b + V^b\left(M_t^b, t\right)\right]$$

subject to the budget constraint

$$\phi_t \cdot M_t^b + x_t^b = \phi_t \cdot M_{t-1}^b.$$

The value for a buyer holding a portfolio M_t^b in the DM is

$$\begin{aligned} V^b\left(M_t^b, t\right) &= \sigma\left[u\left(q\left(M_t^b, t\right)\right) + \beta W^b\left(M_t^b - d\left(M_t^b, t\right), t+1\right)\right] \\ &\quad + (1-\sigma)\,\beta W^b\left(M_t^b, t+1\right) \end{aligned}$$

and let $q^* \in \mathbb{R}$ denote the quantity satisfying $u'(q^*) = w'(q^*)$ so that q^* gives the surplus-maximizing quantity that determines the efficient level of production in the DM. The solution to the bargaining problem is given by

$$q\left(M_t^b, t\right) = \begin{cases} m^{-1}\left(\beta \times \phi_{t+1} \cdot M_t^b\right) & \text{if } \phi_{t+1} \cdot M_t^b < \beta^{-1}\left[\theta w\left(q^*\right) + (1-\theta)\,u\left(q^*\right)\right] \\ q^* & \text{if } \phi_{t+1} \cdot M_t^b \geq \beta^{-1}\left[\theta w\left(q^*\right) + (1-\theta)\,u\left(q^*\right)\right] \end{cases}$$

and

$$\phi_{t+1} \cdot d\left(M_t^b, t\right) = \begin{cases} \phi_{t+1} \cdot M_t^b & \text{if } \phi_{t+1} \cdot M_t^b < \beta^{-1}\left[\theta w\left(q^*\right) \right. \\ & \left. + (1-\theta)\,u\left(q^*\right)\right] \\ \beta^{-1}\left[\theta w\left(q^*\right) + (1-\theta)\,u\left(q^*\right)\right] & \text{if } \phi_{t+1} \cdot M_t^b \geq \beta^{-1}\left[\theta w\left(q^*\right) \right. \\ & \left. + (1-\theta)\,u\left(q^*\right)\right] \end{cases}$$

with the function $m : \mathbb{R}_+ \to \mathbb{R}$ defined as

$$m(q) \equiv \frac{(1-\theta)\, u(q)\, w'(q) + \theta w(q)\, u'(q)}{\theta u'(q) + (1-\theta)\, w'(q)}.$$

The optimal portfolio problem can be defined as

$$\max_{M_t^b \in \mathbb{R}_+^N} \left\{ -\phi_t \cdot M_t^b + \sigma \left[u\left(q\left(M_t^b, t \right) \right) - \beta \times \phi_{t+1} \cdot d\left(M_t^b, t \right) \right] + \beta \times \phi_{t+1} \cdot M_t^b \right\}$$

thus the optimal choice satisfies

$$\phi_t^i = \beta \phi_{t+1}^i L_\theta \left(\phi_{t+1} \cdot M_t^b \right) \tag{5.2}$$

for every type $i \in \{1, \ldots, N\}$ together with the transversality condition

$$\lim_{t \to \infty} \beta^t \times \phi_t \cdot M_t^b = 0 \tag{5.3}$$

where $L_\theta : \mathbb{R}_+ \to \mathbb{R}_+$ is given by

$$L_\theta(A) = \begin{cases} \sigma \frac{u'\left(m^{-1}(\beta A) \right)}{w'(m^{-1}(\beta A))} + 1 - \sigma & \text{if } A < \beta^{-1} \left[\theta w(q^*) + (1-\theta)\, u(q^*) \right] \\ 1 & \text{if } A \geq \beta^{-1} \left[\theta w(q^*) + (1-\theta)\, u(q^*) \right] \end{cases}$$

Definition 28. (Sellers). In a $[0,1]$-continuum of sellers, $x_t^s \in \mathbb{R}$ denotes the seller's net consumption of the CM good and $n_t \in \mathbb{R}_+$denotes the seller's effort level to produce the DM good. The utility function of the seller's preferences is given by

$$U^s(x_t^s, n_t) = x_t^s - w(n_t)$$

with $w : \mathbb{R}_+ \to \mathbb{R}_+$ continuously differentiable, increasing and weakly convex with $w(0) = 0$.

Let $W^s\left(M_{t-1}^s, t \right)$ denote the value function for a seller who enters period t holding a portfolio $M_{t-1}^s \in \mathbb{R}_+^N$ of cryptocurrencies in the CM and let $V^s(M_t^s, t)$ denote the value function in the DM: the dynamic programming equation is

$$W^s\left(M_{t-1}^s \right) = \max_{(x_t^s, M_t^s) \in \mathbb{R} \times \mathbb{R}_+^N} \left[x_t^s + V^s(M_t^s, t) \right]$$

subject to the budget constraint

$$\phi_t \cdot M_t^s + x_t^s = \phi_t \cdot M_{t-1}^s.$$

The value for a seller holding a portfolio M_t^s in the DM is

$$\begin{aligned} V^s(M_t^s, t) &= \sigma \left[-w\left(q\left(M_t^b, t \right) \right) + \beta W^s\left(M_t^s + d\left(M_t^b, t \right), t+1 \right) \right] \\ &\quad + (1-\sigma)\, \beta W^s(M_t^s, t+1) \end{aligned}$$

Definition 29. (Miners). In a $[0,1]$-continuum of miners of each type-$i \in \{1,\ldots,N\}$ token, $x_t^i \in \mathbb{R}_+$ denotes the miner's consumption of the CM good and $\triangle_t^i \in \mathbb{R}_+$denotes the production of the type-i token. The utility function of the miner's preferences is given by

$$U^e\left(x_t^i, \Delta_t\right) = x_t^i - c\left(\Delta_t^i\right)$$

with the cost function $c : \mathbb{R}_+ \to \mathbb{R}_+$ continuously differentiable, strictly increasing and weakly convex with $c(0) = 0$.

Let $M_t^i \in \mathbb{R}_+$ denote the per-capita supply of cryptocurrency i in period t and $\Delta_t^i \in \mathbb{R}$ denote the miner i's net circulation of newly minted tokens in period t. To describe the miner's problem to determine the money supply in the economy, we start assuming that all miners solve the same decision problem, thus the law of motion of type-i tokens at all date $t \geq 0$ is given by

$$M_t^i = \Delta_t^i + M_{t-1}^i \tag{5.4}$$

where $M_{-1}^i \in \mathbb{R}_+$ denotes the initial stock. The budget constraint is

$$x_j^i = \phi_t^i \Delta_t^i,$$

and given that the miner takes prices $\{\phi_t\}_{t=0}^{\infty}$ as given, the profit maximization of the cryptocurrency emission problem is solved by

$$\Delta_t^{*,i} \in \arg\max_{\Delta \in \mathbb{R}_+} \left[\phi_t^i \Delta - c(\Delta)\right] \tag{5.5}$$

Definition 30. (Equilibrium). A perfect-foresight monetary equilibrium is an array $\left\{M_t, M_t^b, \Delta_t^*, \phi_t\right\}_{t=0}^{\infty}$ satisfying 5.2, 5.3, 5.5 and 5.4 for each $i \in \{1,\ldots,N\}$ at all dates $t \geq 0$ and satisfying the following market-clearing condition

$$M_t = M_t^b + M_t^s \tag{5.6}$$

Suppose that each miner j starts with $M^i > 0$ units of currency $i \in \{1,\ldots,N\}$: let δ denote the fraction $1-\delta$ of randomly selected miners in each location j at each date $t \geq 0$ who doesn't offer their tokens to sellers because they are holding them in expectation of their appreciation (i.e., PLFC cryptocurrencies), so these tokens don't circulate to other j positions whenever sellers are relocated.

Conversely, an equilibrium with the property that miners don't restrict the circulation of recently mined tokens (i.e., UDCE cryptocurrencies) is as follows: the optimal portfolio choice implies the first-order condition

$$\frac{u'\left(q\left(M_t, t\right)\right)}{w'\left(q\left(M_t, t\right)\right)} = \frac{1}{\beta \gamma_{t+1}^i}$$

for each currency i, where $\gamma_{t+1} \in \mathbb{R}_+$ represents the common return across all valued currencies between dates t and $t+1$. The demand for real balances in

each location is given by

$$z\left(\gamma_{t+1};1\right)\equiv\frac{1}{\gamma_{t+1}}L_1^{-1}\left(\frac{1}{\beta\gamma_{t+1}}\right)$$

because

$$\beta\gamma_{t+1}\sum_{i=1}^{N}b_t^i<\theta w\left(q^*\right)+\left(1-\theta\right)u\left(q^*\right)$$

and with

$$L_\delta\left(A\right)=\begin{cases}\delta\frac{u'\left(m^{-1}(\beta A)\right)}{w'\left(m^{-1}(\beta A)\right)}+1-\delta & \text{if } A<\beta^{-1}w\left(q^*\right)\\ 1 & \text{if } A\geq\beta^{-1}w\left(q^*\right)\end{cases}$$

Because the market-clearing condition implies

$$\sum_{i=1}^{N}\phi_t^iM^i=z\left(\gamma_{t+1};1\right)$$

the equilibrium sequence $\{\gamma_t\}_{t=0}^{\infty}$ satisfies the law of motion

$$z\left(\gamma_{t+1};1\right)=\gamma_tz\left(\gamma_t;1\right)$$

because

$$M_t^i=M_{t-1}^i=\Delta^i$$

for each i and provided that $\gamma_t\leq t$, and the boundary condition

$$\beta\gamma_tz\left(\gamma_t;1\right)\leq w\left(q^*\right)$$

Suppose $u\left(q\right)=\left(1-\eta\right)^{-1}q^{1-\eta}$, with $0<\eta<1$, and $w\left(q\right)=\left(1+\alpha\right)^{-1}q^{1+\alpha}$ with $\alpha\geq0$. The dynamic system describing the equilibrium evolution of γ_t is

$$\gamma_{t+1}^{\frac{1+\alpha}{\eta+\alpha}-1}=\gamma_t^{\frac{1+\alpha}{\eta+\alpha}}\tag{5.7}$$

Theorem 31. *The allocation associated with the circulation of UDCE cryptocurrencies Pareto dominates the allocation with the associated the circulation of PLFC cryptocurrencies, on a stationary equilibrium with the property that the quantity traded in the Decentralised Market is given by* $\hat{q}\left(1\right)\in\left(\hat{q}\left(\delta\right),q^*\right)$ *satisfying*

$$\frac{u'\left(\hat{q}\left(1\right)\right)}{w'\left(\hat{q}\left(1\right)\right)}=\beta^{-1}\tag{5.8}$$

Proof. The sequence $\gamma_t=1$ for all $t\geq0$ satifies 5.7. Then, the solution to the optimal portfolio problem implies that the DM output must satisfy 5.8. The quantity $\hat{q}$ satisfies

$$\frac{u'\left(\hat{q}\left(1\right)\right)}{w'\left(\hat{q}\left(1\right)\right)}=\delta\frac{u'\left(\hat{q}\left(\delta\right)\right)}{w'\left(\hat{q}\left(\delta\right)\right)}+1-\delta$$

Because $\delta\in\left(0,1\right)$, we have $\hat{q}\left(1\right)>\hat{q}\left(\delta\right)$, that is, the allocation associated with the circulation of UCDE cryptocurrencies -$\hat{q}\left(1\right)$- Pareto dominates the allocation associated with the circulation of PLFC cryptocurrencies -$\hat{q}\left(\delta\right)$-. □

5.2.5 On Network Effects

At the time of the release of this paper, cryptocurrencies have failed to provide an alternative to traditional payment networks due to a combination of high transaction fees, high finalization time and high volatility. The failure to find the favor of merchants is also their biggest weakness: they aren't part of two-sided networks, and thus easily replaceable by any newer cryptocurrency better able to create them. Actually, the first-mover advantage of the most valued cryptocurrencies is lower than expected if any competing cryptocurrency leverages network effects from other different sources (e.g., Zero-Knowledge Proof-of-Identity from trusted PKI certificates).

A simple model is introduced here to analyze the evolution of competing payment networks: consider the two-sided and incompatible payment networks of two cryptocurrencies, BTC and zk-PoI, each with their corresponding groups of merchants m and customers c; let m^t_{BTC}, m^t_{zkPOI} denote the number of merchants at time t and c^t_{BTC}, c^t_{zkPOI} the number of customers. A user joins the payment networks at each time step t, with λ being the probability of being a customer and $1-\lambda$ of being a merchant: each merchant prefers to join BTC or zk-PoI depending on the number of customers in the same network, thus the probabilities to join one of the networks are given by

$$\frac{c^\beta_{BTC}}{c^\beta_{BTC}+c^\beta_{zkPOI}}, \frac{c^\beta_{zkPOI}}{c^\beta_{BTC}+c^\beta_{zkPOI}}$$

and conversely, for customers the probabilities are given by

$$\frac{m^\alpha_{BTC}}{m^\alpha_{BTC}+m^\alpha_{zkPOI}}, \frac{m^\alpha_{zkPOI}}{m^\alpha_{BTC}+m^\alpha_{zkPOI}}.$$

Note that some categories of users would prefer to use the expected number of users and not their current tally: forward-looking merchants that need to invest on equipment to access the payment network are within this group, thus they would prefer to use expected numbers,

$$\frac{E\left(c^\beta_{BTC}\right)}{E\left(c^\beta_{BTC}\right)+E\left(c^\beta_{zkPOI}\right)}, \frac{E\left(c^\beta_{zkPOI}\right)}{E\left(c^\beta_{BTC}\right)+E\left(c^\beta_{zkPOI}\right)}.$$

Each user can only join one payment network, modelling the fact that single-homing is preferred to multi-homing in the real-world, and the particular network is determined by the distribution of users on the other side at each time t. The parameters $\alpha, \beta > 0$ are elasticities of demand with regard to the numbers of users on the other side of the payment network, effectively acting as measures of indirect network effects: α can be empirically estimated by observing joining customers over a small period of time and then calculating

$$\alpha = \frac{\ln\ (m^\alpha_{BTC}/(m^\alpha_{BTC}+m^\alpha_{zkPOI})) - \ln\ (1-(m^\alpha_{BTC}/(m^\alpha_{BTC}+m^\alpha_{zkPOI})))}{\ln\ m_{BTC} - \ln\ m_{zkPOI}}$$

and converselyβ can be empirically estimated by observing joining merchants and then calculating

$$\beta = \frac{\ln\ \left(c_{BTC}^{\beta}/(c_{BTC}^{\beta}+c_{zkPOI}^{\beta})\right) - \ln\ \left(1-\left(c_{BTC}^{\beta}/(c_{BTC}^{\beta}+c_{zkPOI}^{\beta})\right)\right)}{\ln c_{BTC} - \ln c_{zkPOI}}.$$

Theorem 32. *(Dominance of the Zero-Knowledge Proof-of-Identity cryptocurrency). A new cryptocurrency could achieve dominance over previous cryptocurrencies, overcoming first-mover advantages, if the expected number of accepting customers would be much higher and the number of merchants using the previous cryptocurrencies is low.*

Proof. Note that the number of steps needed for a new cryptocurrency, m_{zkPOI}, to overtake the previous one, m_{BTC}, on the number of merchants, $m_{zkPOI} > m_{BTC}$, is given by

$$(m_{BTC}+1)\cdot(1-\lambda)^{-1}$$

It's possible for a new cryptocurrency to overtake a previous one on the number of merchants whenever

$$E(c_{zkPOI}) - E(c_{BTC}) > (m_{BTC}+1)\cdot(1-\lambda)^{-1}$$

and since m_{BTC} is a low number and $E(c_{zkPOI}) \gg E(c_{BTC})$, it's conceivable that the previous condition could hold.

Now let's consider the results of strong network effects on the final market shares of both payment networks by examining the following differential equations,

$$\frac{d\left(m_{BTC}/m_{zkPOI}\right)}{dt} = (1-\lambda)\,\frac{\left(c_{BTC}/c_{zkPOI}\right)^{\beta} - \left(m_{BTC}/m_{zkPOI}\right)}{\left(1+\left(c_{BTC}/c_{zkPOI}\right)^{\beta}\right)m_{zkPOI}}$$

and

$$\frac{d\left(c_{BTC}/c_{zkPOI}\right)}{dt} = \lambda\frac{\left(m_{BTC}/m_{zkPOI}\right)^{\alpha} - \left(c_{BTC}/c_{zkPOI}\right)}{\left(1+\left(m_{BTC}/m_{zkPOI}\right)^{\alpha}\right)c_{zkPOI}}$$

According to the signs of the previous derivatives, when $\alpha\cdot\beta > 1$ and $t\to\infty$, the payment network with even a slight advantage over the other will end acquiring all the merchants and customers, for example

$$\lim_{t\to\infty} m_{zkPOI} = \infty, \lim_{t\to\infty} c_{zkPOI} = \infty$$

$$\lim_{t\to\alpha} m_{BTC} = 0, \lim_{t\to\alpha} c_{BTC} = 0$$

but with $\alpha\cdot\beta < 1$, the number of merchants and customers will equalize

$$m_{zkPOI} = m_{BTC}, c_{zkPOI} = c_{BTC}$$

thus highlighting the importance of network effects. □

5.2.6 Dominance over Cash and other Cryptocurrencies

The dominance of subsection 5.2.1 is based on mining and subsection 5.2.4 extends said dominance to the circulation of currencies: in this subsection, the dominance will be based on the lower costs of a payment network of the cryptocurrency using Zero-Knowledge Proof-of-Identity; therefore, there exists a unique equilibrium in which this payment system dominates.

A recent paper[Pid15] offers a model based on a version of Lagos-Wright[HL05] to explain the substitution of cash by debit cards or any other non-deferred electronic payment system incurring a fixed cost $\Omega(z)\tau$ per each period τ, the cost $\Omega(z)$ being financed by imposing fee ω on each payment where ω should satisfy

$$\Omega(z) = S[\theta\omega + (1-\theta)\omega]$$

and where S denotes the instantaneous measure of electronic payment transactions, z is the state of development of the economy, $\theta \in [0,1]$ is the share of cost allocated to a buyer and $(1-\theta)$ is the share of cost allocated to a seller. In this model, an electronic payment system can achieve dominance over cash using the solution concept of iterative elimination of conditionally dominated strategies whenever the state of development of the economy z is sufficiently high, and there exists a unique equilibrium in the model such that agents choose electronic payment transactions when z is strictly higher than the limiting cut-off function Z_∞ of the sequence of boundaries $Z_0, Z_1, \ldots$ of regions where an agent chooses electronic payment transactions regardless of the choices of other agents. In other words, it's strictly dominant to choose electronic payments in an economy having sufficiently advanced information technology so that Ω is negligible.

Corollary 33. *Since the cost function $\Omega(z)$ of a UCDE cryptocurrency based on Zero-Knowledge Proof-of-Identity is much cheaper than of PoW/PoS cryptocurrencies and other forms of electronic payment because its cards are already distributed (i.e., de facto subsidised by governments), there exists a unique equilibrium in the model [Pid15] such that the agents choose the UCDE cryptocurrency using zk-PoI and it dominates the other forms of payment.*

6 Conclusion

Although all permissionless blockchains critically depend on Proof-of-Work or Proof-of-Stake to prevent Sybil attacks, their high resource consumption corroborates their non-scalability and act as a limiting factor to the general diffusion of blockchains. This paper proposed an alternative approach that not only doesn't waste resources, it could also help in the real-world identity challenges faced by permissionless blockchains: the derivation of anonymous credentials from widely trusted public PKI certificates.

Additionally, we study the better incentives offered by the proposed cryptocurrency based on our anonymous authentication scheme: mining is proved to be incentive-compatible and a strictly dominant strategy over previous cryptocurrencies, thus a Nash equilibrium over previous cryptocurrencies and an

Evolutionary Stable Strategy; furthermore, zk-PoI is proved to be optimal because it implements the social optimum, unlike PoW/PoS cryptocurrencies that are paying the Price of (Crypto-)Anarchy. The circulation of the proposed cryptocurrency is proved to Pareto dominate other cryptocurrencies based on its negligible mining costs and it could also become dominant thanks to stronger network effects; finally, the lower costs of its infrastructure imply the existence of a unique equilibrium where it dominates other forms of payment.

References

[AABM17] Sarah Azouvi, Mustafa Al-Bassam, and Sarah Meiklejohn. Who Am I? Secure Identity Registration on Distributed Ledgers, 2017. https://smeiklej.com/files/cbt17.pdf.

[ABFG13] Giuseppe Ateniese, Ilario Bonacina, Antonio Faonio, and Nicola Galesi. Proofs of Space: When Space is of the Essence, 2013. https://eprint.iacr.org/2013/805.

[AC10] Aymen Abed and Sébastien Canard. One Time Anonymous Certificate: X.509 Supporting Anonymity, 2010. https://link.springer.com/chapter/10.1007/978-3-642-17619-7_23.

[ACP+16] Joël Alwen, Binyi Chen, Krzysztof Pietrzak, Leonid Reyzin, and Stefano Tessaro. Scrypt is Maximally Memory-Hard. Cryptology ePrint Archive, Report 2016/989, 2016. https://eprint.iacr.org/2016/989.

[AG16] Louise Axon and Michael Goldsmith. PB-PKI: a privacy-aware blockchain-based PKI, 2016. https://ora.ox.ac.uk/objects/uuid:8bd33c4f-5614-400e-b958-48e53fe1b342.

[AGJS13] Ittai Anati, Shay Gueron, Simon P Johnson, and Vincent R Scarlata. Innovative Technology for CPU Based Attestation and Sealing, 2013. https://software.intel.com/sites/default/files/article/413939/hasp-2013-innovative-technology-for-attestation-and-sealing.pdf.

[AGM18] Shashank Agrawal, Chaya Ganesh, and Payman Mohassel. Non-Interactive Zero-Knowledge Proofs for Composite Statements. Cryptology ePrint Archive, Report 2018/557, 2018. https://eprint.iacr.org/2018/557.

[AH19] Sarah Azouvi and Alexander Hicks. SoK: Tools for Game Theoretic Models of Security for Cryptocurrencies, 2019. https://arxiv.org/pdf/1905.08595.pdf.

[AKMP08] Christer Andersson, Markulf Kohlweiss, Leonardo A. Martucci, and Andriy Panchenko. A Self-certified and Sybil-Free Framework for Secure Digital Identity Domain Buildup. In *Information*

Security Theory and Practices. Smart Devices, Convergence and Next Generation Networks, Second IFIP WG 11.2 International Workshop, WISTP 2008, Seville, Spain, May 13-16, 2008. Proceedings, pages 64–77, 2008. http://dl.ifip.org/db/conf/wistp/wistp2008/AnderssonKMP08.pdf.

[All19] SIM Alliance. eUICC Technical Releases, 2019. https://simalliance.org/euicc/euicc-technical-releases/.

[Ana19] Hiroaki Anada. Decentralized Multi-authority Anonymous Authentication for Global Identities with Non-interactive Proofs, 2019. https://eprint.iacr.org/2019/701.

[ARMM+18] Eitan Altman, Alexandre Reiffers-Masson, Daniel Sadoc Menasché, Mandar Datar, Swapnil Dhamal, and Corinne Touati. Mining competition in a multi-cryptocurrency ecosystem at the network edge: A congestion game approach. In *SOCCA 2018 - 1st Symposium on Cryptocurrency Analysis*, pages 1–4, Toulouse, France, December 2018. https://hal.inria.fr/hal-01906954/file/camera_socca.pdf.

[Arn18] Arnau. darkID: A proof of concept of an anonymous decentralized identification system based on blockchain, 2018. https://github.com/arnaucode/darkID.

[Ass17] GSMA Association. GSMA eUICC PKI Certificate Policy, 2017. https://www.gsma.com/iot/wp-content/uploads/2017/04/SGP.14_v1.1.pdf.

[BBF+07] G. Bianchi, M. Bonola, V. Falletta, F. S. Proto, and S. Teofili. The SPARTA pseudonym and authorization system, 2007. https://core.ac.uk/download/pdf/82569804.pdf.

[BBF18] Dan Boneh, Benedikt Bünz, and Ben Fisch. Batching Techniques for Accumulators with Applications to IOPs and Stateless Blockchains, 2018. https://eprint.iacr.org/2018/1188.

[BCGS16] Dan Boneh, Henry Corrigan-Gibbs, and Stuart Schechter. Balloon Hashing: A Memory-Hard Function Providing Provable Protection Against Sequential Attacks. Cryptology ePrint Archive, Report 2016/027, 2016. https://eprint.iacr.org/2016/027.

[BCKL07] Mira Belenkiy, Melissa Chase, Markulf Kohlweiss, and Anna Lysyanskaya. Non-Interactive Anonymous Credentials. Cryptology ePrint Archive, Report 2007/384, 2007. https://eprint.iacr.org/2007/384.

[BCKL09] Mira Belenkiy, Melissa Chase, Markulf Kohlweiss, and Anna Lysyanskaya. Compact E-Cash and Simulatable VRFs Revisited. Cryptology ePrint Archive, Report 2009/107, 2009. https://eprint.iacr.org/2009/107.

[BCLP14] Julien Bringer, Hervé Chabanne, Roch Lescuyer, and Alain Patey. Efficient and Strongly Secure Dynamic Domain-Specific Pseudonymous Signatures for ID Documents. Cryptology ePrint Archive, Report 2014/067, 2014. https://eprint.iacr.org/2014/067.

[BDFK12] Jens Bender, Özgür Dagdelen, Marc Fischlin, and Dennis Kügler. Domain-Specific Pseudonymous Signatures for the German Identity Card. Cryptology ePrint Archive, Report 2012/558, 2012. https://eprint.iacr.org/2012/558.

[BDK15] Alex Biryukov, Daniel Dinu, and Dmitry Khovratovich. Fast and Tradeoff-Resilient Memory-Hard Functions for Cryptocurrencies and Password Hashing. Cryptology ePrint Archive, Report 2015/430, 2015. https://eprint.iacr.org/2015/430.

[BdM94] Josh Benaloh and Michael de Mare. One-Way Accumulators: A Decentralized Alternative to Digital Signatures. In Tor Helleseth, editor, *Advances in Cryptology — EUROCRYPT '93*, pages 274–285, Berlin, Heidelberg, 1994. Springer Berlin Heidelberg. https://www.microsoft.com/en-us/research/wp-content/uploads/1993/01/owa.pdf.

[BDN18] Dan Boneh, Manu Drijvers, and Gregory Neven. Compact Multi-Signatures for Smaller Blockchains, 2018. https://eprint.iacr.org/2018/483.

[BG19] Alun John Brenda Goh. China wants to ban bitcoin mining, traders say move not a surprise, 2019. https://www.reuters.com/article/us-china-cryptocurrency/china-says-it-wants-to-eliminate-bitcoin-mining-idUSKCN1RL0C4.

[BGG17] Sean Bowe, Ariel Gabizon, and Matthew D. Green. A multi-party protocol for constructing the public parameters of the Pinocchio zk-SNARK. Cryptology ePrint Archive, Report 2017/602, 2017. https://eprint.iacr.org/2017/602.

[BHK+18] P. Błaskiewicz, Lucjan Hanzlik, Kamil Kluczniak, Ł. Krzywiecki, Mirosław Kutyłowski, M. Słowik, and M. Wszoła. Pseudoanonymous Signatures Schemes, 2018. https://doi.org/10.1007/978-981-13-1483-4.

[BKKJ+17] Maria Borge, Eleftherios Kokoris-Kogias, Philipp Jovanovic, Linus Gasser, Nicolas Gailly, and Bryan Ford. Proof-of-Personhood: Redemocratizing Permissionless Cryptocurrencies, 2017. https://zerobyte.io/publications/2017-BKJGGF-pop.pdf.

[BL12] Foteini Baldimtsi and Anna Lysyanskaya. Anonymous Credentials Light. Cryptology ePrint Archive, Report 2012/298, 2012. https://eprint.iacr.org/2012/298.

[BLS01] Dan Boneh, Ben Lynn, and Hovav Shacham. Short Signatures from the Weil Pairing. pages 514–532. Springer-Verlag, 2001. https://cseweb.ucsd.edu/~hovav/dist/sigs.pdf.

[BLT19] George Bissias, Brian N. Levine, and David Thibodeau. Using Economic Risk to Model Miner Hash Rate Allocation in Cryptocurrencies, 2019. https://people.cs.umass.edu/~gbiss/economic_risk.pdf.

[CBGL19] Panagiotis Chatzigiannis, Foteini Baldimtsi, Igor Griva, and Jiasun Li. Diversification Across Mining Pools: Optimal Mining Strategies under PoW, 2019. https://arxiv.org/pdf/1905.04624.pdf.

[CC19] Crypto-CoinZ. Crypto-CoinZ, 2019. https://www.crypto-coinz.net/crypto-calculator/.

[CD16] Victor Costan and Srinivas Devadas. Intel SGX Explained. Cryptology ePrint Archive, Report 2016/086, 2016. https://eprint.iacr.org/2016/086.

[CDD17] Jan Camenisch, Manu Drijvers, and Maria Dubovitskaya. Practical UC-Secure Delegatable Credentials with Attributes and Their Application to Blockchain. In *Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security, CCS 2017, Dallas, TX, USA, October 30 - November 03, 2017*, pages 683–699, 2017. https://acmccs.github.io/papers/p683-camenischA.pdf.

[CDDH19] Jan Camenisch, Manu Drijvers, Petr Dzurenda, and Jan Hajny. Fast Keyed-Verification Anonymous Credentials on Standard Smart Cards, 2019. https://eprint.iacr.org/2019/460.

[CFH+14] Craig Costello, Cédric Fournet, Jon Howell, Markulf Kohlweiss, Benjamin Kreuter, Michael Naehrig, Bryan Parno, and Samee Zahur. Geppetto: Versatile Verifiable Computation. Cryptology ePrint Archive, Report 2014/976, 2014. https://eprint.iacr.org/2014/976.

[CG10] Jan Camenisch and Thomas Groß. Efficient Attributes for Anonymous Credentials (Extended Version). Cryptology ePrint Archive, Report 2010/496, 2010. https://eprint.iacr.org/2010/496.

[Cha83] David Chaum. Blind Signatures for Untraceable Payments. In David Chaum, Ronald L. Rivest, and Alan T. Sherman, editors, *Advances in Cryptology*, pages 199–203, Boston, MA, 1983. Springer US. http://sceweb.sce.uhcl.edu/yang/teaching/csci5234WebSecurityFall2011/Chaum-blind-signatures.PDF.

[CHKO12] Philippe Camacho, Alejandro Hevia, Marcos Kiwi, and Roberto Opazo. Strong accumulators from collision-resistant hashing. *International Journal of Information Security*, 11(5):349–363, Oct 2012. https://users.dcc.uchile.cl/~pcamacho/papers/strongacc08.pdf.

[CHL05] Jan Camenisch, Susan Hohenberger, and Anna Lysyanskaya. Compact E-Cash. Cryptology ePrint Archive, Report 2005/060, 2005. https://eprint.iacr.org/2005/060.

[CL99] Miguel Castro and Barbara Losko. Practical Byzantine Fault Tolerance, 1999. http://pmg.csail.mit.edu/papers/osdi99.pdf.

[CL01] Jan Camenisch and Anna Lysyanskaya. An Efficient System for Non-transferable Anonymous Credentials with Optional Anonymity Revocation. Cryptology ePrint Archive, Report 2001/019, 2001. https://eprint.iacr.org/2001/019.

[CL02] Jan Camenisch and Anna Lysyanskaya. Dynamic Accumulators and Application to Efficient Revocation of Anonymous Credentials. In Moti Yung, editor, *Advances in Cryptology — CRYPTO 2002*, pages 61–76, Berlin, Heidelberg, 2002. Springer Berlin Heidelberg. https://cs.brown.edu/people/alysyans/papers/camlys02.pdf.

[CL04] Jan Camenisch and Anna Lysyanskaya. Signature Schemes and Anonymous Credentials from Bilinear Maps. In Matt Franklin, editor, *Advances in Cryptology – CRYPTO 2004*, pages 56–72, Berlin, Heidelberg, 2004. Springer Berlin Heidelberg. http://cs.brown.edu/~anna/papers/cl04.pdf.

[CLC17] Fabien Coelho, Arnaud Larroche, and Baptiste Colin. Itsuku: a Memory-Hardened Proof-of-Work Scheme. Cryptology ePrint Archive, Report 2017/1168, 2017. https://eprint.iacr.org/2017/1168.

[Coi19] CoinWarz. CoinWarz, 2019. https://www.coinwarz.com.

[Cor18] Intel Corporation. L1 Terminal Fault / CVE-2018-3615 , CVE-2018-3620,CVE-2018-3646 / INTEL-SA-00161, 2018. https://software.intel.com/security-software-guidance/software-guidance/l1-terminal-fault.

[Cry19a] CryptoCompare. CryptoCompare, 2019. https://www.cryptocompare.com.

[Cry19b] CryptoZone. CryptoZone, 2019. https://crypt0.zone/calculator.

[CZL19] Guoxing Chen, Yinqian Zhang, and Ten-Hwang Lai. OPERA: Open Remote Attestation for Intel's Secure Enclaves, 2019. https://dl.acm.org/citation.cfm?id=3354220.

[Dan18] George Danezis. Combining social network based Sybil detection to secure state of the art Federated Byzantine Agreement systems, 2018. https://github.com/gdanezis/SybilQuorum.

[DC19] China's National Development and Reform Commission. Catalog for Guiding Industry Restructuring, 2019. http://gys.ndrc.gov.cn/cyjgtzzdml20190408.pdf.

[DFKP13] Stefan Dziembowski, Sebastian Faust, Vladimir Kolmogorov, and Krzysztof Pietrzak. Proofs of Space, 2013. https://eprint.iacr.org/2013/796.

[DHS15] David Derler, Christian Hanser, and Daniel Slamanig. Revisiting Cryptographic Accumulators, Additional Properties and Relations to other Primitives, 2015. https://eprint.iacr.org/2015/087.

[Dia19] Diar. Cosmos In Vogue for Proof-of-Stake Blockchain Model, 2019. https://diar.co/volume-3-issue-9/.

[DKMJ18] Philip Daian, Tyler Kell, Ian Miers, and Ari Juels. On-Chain Vote Buying and the Rise of Dark DAOs, 2018. http://hackingdistributed.com/2018/07/02/on-chain-vote-buying/.

[DLFKP16] Antoine Delignat-Lavaud, Cédric Fournet, Markulf Kohlweiss, and Bryan Parno. Cinderella: Turning Shabby X.509 Certificates into Elegant Anonymous Credentials with the Magic of Verifiable Computation. In *Proceedings of the IEEE Symposium on Security and Privacy*. IEEE – Institute of Electrical and Electronics Engineers, May 2016. https://www.microsoft.com/en-us/research/wp-content/uploads/2016/06/cinderella-1.pdf.

[DLS88] Cynthia Dwork, Nancy Lynch, and Larry Stockmeyer. Consensus in the Presence of Partial Synchrony, 1988. https://groups.csail.mit.edu/tds/papers/Lynch/jacm88.pdf.

[DM96] Lloyd S. Shapley Dov Monderer. Potential Games, 1996. https://www.cs.bu.edu/~steng/teaching/Fall2008/potential.pdf.

[Dou02] John (JD) Douceur. The Sybil Attack. In *Proceedings of 1st International Workshop on Peer-to-Peer Systems (IPTPS)*, January 2002. https://www.microsoft.com/en-us/research/wp-content/uploads/2002/01/IPTPS2002.pdf.

[DPS16] Phil Daian, Rafael Pass, and Elaine Shi. Snow White: Provably Secure Proofs of Stake. Cryptology ePrint Archive, Report 2016/919, 2016. https://eprint.iacr.org/2016/919.

[DS83] Danny Dolev and H. Raymond Strong. Authenticated algorithms for byzantine agreement, 1983. http://drona.csa.iisc.ernet.in/~arpita/BroadcastBAReadingGroup/DS83.pdf.

[DW18] Christian Decker and Roger Wattenhofer. Information Propagation in the Bitcoin Network, 2018. http://www.tik.ee.ethz.ch/file/49318d3f56c1d525aabf7fda78b23fc0/P2P2013_041.pdf.

[DYSZ17] Wang Dejia, Guo Yu, Wang Shaofan, and Jiang Zhongzheng. Blockchain based CA (Certificate Authority) management method, device and system, 2017. http://worldwide.espacenet.com/publicationDetails/biblio?DB=EPODOC&II=0&ND=3&adjacent=true&locale=en_EP&FT=D&CC=CN&NR=106384236A&KC=A.

[EAK+17] Abeer ElBahrawy, Laura Alessandretti, Anne Kandler, Romualdo Pastor-Satorras, and Andrea Baronchelli. Evolutionary dynamics of the cryptocurrency market. *arXiv e-prints*, page arXiv:1705.05334, May 2017. https://arxiv.org/abs/1705.05334.

[ES13] Ittay Eyal and Emin Gün Sirer. Majority is not Enough: Bitcoin Mining is Vulnerable. *CoRR*, abs/1311.0243, 2013. https://arxiv.org/abs/1311.0243.

[Eya14] Ittay Eyal. The Miner's Dilemma. *CoRR*, abs/1411.7099, 2014. https://arxiv.org/abs/1411.7099.

[FAT19] FATF. Guidance for a Risk-Based Approach to Virtual Assets and Virtual Asset Service Providers, 2019. https://www.fatf-gafi.org/media/fatf/documents/recommendations/RBA-VA-VASPs.pdf.

[fIS16] BSI Federal Office for Information Security. Technical Guideline TR-03110 - Advanced Security Mechanisms for Machine Readable Travel Documents and eIDAS Token 2.20, 2016. https://www.bsi.bund.de/SharedDocs/Downloads/EN/BSI/Publications/TechGuidelines/TR03110/BSI_TR-03110_Part-2-V2_2.pdf?__blob=publicationFile&v=3.

[FMMO18] Prastudy Fauzi, Sarah Meiklejohn, Rebekah Mercer, and Claudio Orlandi. Quisquis: An Anonymous Cryptocurrency Based on Updatable Public Keys, 2018. https://eprint.iacr.org/2018/990.

[FVS16] Jesus Fernandez-Villaverde and Daniel Sanches. Can Currency Competition Work? NBER Working Papers 22157, National Bureau of Economic Research, Inc, 2016. http://economics.sas.upenn.edu/~jesusfv/currency_competition.pdf.

[FVY14] Conner Fromknecht, Dragos Velicanu, and Sophia Yakoubov. A Decentralized Public Key Infrastructure with Identity Retention, 2014. https://eprint.iacr.org/2014/803.pdf.

[FWB15] Martin Florian, Johannes Walter, and Ingmar Baumgart. Sybil-Resistant Pseudonymization and Pseudonym Change without Trusted Third Parties, 2015. http://telematics.tm.kit.edu/publications/Files/566/wpes25f-florianA.pdf.

[GBE+18] Adem Efe Gencer, Soumya Basu, Ittay Eyal, Robbert van Renesse, and Emin Gün Sirer. Decentralization in Bitcoin and Ethereum Networks. *CoRR*, abs/1801.03998, 2018. https://arxiv.org/abs/1801.03998.

[GGM13] Christina Garman, Matthew Green, and Ian Miers. Decentralized Anonymous Credentials. Cryptology ePrint Archive, Report 2013/622, 2013. https://eprint.iacr.org/2013/622.

[GHM+17] Yossi Gilad, Rotem Hemo, Silvio Micali, Georgios Vlachos, and Nickolai Zeldovich. Algorand: Scaling Byzantine Agreements for Cryptocurrencies. Cryptology ePrint Archive, Report 2017/454, 2017. https://eprint.iacr.org/2017/454.

[GOT18] Chaya Ganesh, Claudio Orlandi, and Daniel Tschudi. Proof-of-stake protocols for privacy-aware blockchains. Cryptology ePrint Archive, Report 2018/1105, 2018. https://eprint.iacr.org/2018/1105.

[GP14] Felix Günther and Bertram Poettering. Linkable Message Tagging: Solving the Key Distribution Problem of Signature Schemes, 2014. https://eprint.iacr.org/2014/014.

[GPA18] Lachlan J. Gunn, Ricardo Vieitez Parra, and N. Asokan. Circumventing Cryptographic Deniability with Remote Attestation, 2018. https://eprint.iacr.org/2018/424.

[GRPV19] Sharon Goldberg, Leonid Reyzin, Dimitrios Papadopoulos, and Jan Vcelak. draft-irtf-cfrg-vrf-04 - Verifiable Random Functions, 2019. https://tools.ietf.org/html/draft-irtf-cfrg-vrf-04.

[GSM18] GSMA. eSIM Whitepaper, 2018. https://www.gsma.com/esim/wp-content/uploads/2018/12/esim-whitepaper.pdf.

[GSM19] GSMA. GSMA Root Certificate Issuer, 2019. https://www.gsma.com/esim/ceritificateissuer/.

[HL05] Pidong Huang and Manjong Lee. A Unified Framework for Monetary Theory and Policy Analysis, 2005. https://www.minneapolisfed.org/research/sr/sr346.pdf.

[HMW18] Timo Hanke, Mahnush Movahedi, and Dominic Williams. DFINITY Technology Overview Series Consensus System, 2018. https://dfinity.org/pdf-viewer/library/dfinity-consensus.pdf.

[HSP16] Thomas Hardjono, Ned Smith, and Alex Pentland. Anonymous Identities for Permissioned Blockchains, 2016. https://petertodd.org/assets/2016-04-21/MIT-ChainAnchor-DRAFT.pdf.

[ICA15a] ICAO. Machine Readable Travel Documents (Doc 9303) - Part 10: Logical Data Structures (LDS) for Storage of Biometrics and Other Data in the Contactless Integrated Circuit (IC), 2015. https://www.icao.int/publications/Documents/9303_p10_cons_en.pdf.

[ICA15b] ICAO. Machine Readable Travel Documents (Doc 9303) - Part 11: Security Mechanisms for MRTDs, 2015. https://www.icao.int/publications/Documents/9303_p11_cons_en.pdf.

[ICA18] ICAO. ICAO Public Key Directory, 2018. https://pkddownloadsg.icao.int/.

[IHD16] Marios Isaakidis, Harry Halpin, and George Danezis. UnlimitID: Privacy-Preserving Federated Identity Management using Algebraic MACs, 2016. http://www0.cs.ucl.ac.uk/staff/G.Danezis/papers/UnlimitID_WPES16.pdf.

[Isa00] Isaiah. The Great Isaiah Scroll, -100. http://dss.collections.imj.org.il/isaiah#22:22.

[JKR18] Stanislaw Jarecki, Hugo Krawczyk, and Jason Resch. Threshold Partially-Oblivious PRFs with Applications to Key Management, 2018. https://eprint.iacr.org/2018/733.

[Joh73] John Maynard Smith and George Price. The Logic of Animal Conflict, 1973. https://doi.org/10.1038/246015a0.

[KHK] Kamil Kluczniak, Lucjan Hanzlik, and Mirosław Kutyłowski. A Formal Concept of Domain Pseudonymous Signatures.

[KHK16] Mirosław Kutyłowski, Lucjan Hanzlik, and Kamil Kluczniak. Pseudonymous Signature on eIDAS Token – Implementation Based Privacy Threats. In Joseph K. Liu and Ron Steinfeld, editors, *Information Security and Privacy*, pages 467–477, Cham, 2016. Springer International Publishing. https://doi.org/10.1007/978-3-319-40367-0_31.

[KHK18] Mirosław Kutyłowski, Lucjan Hanzlik, and Kamil Kluczniak. Towards Practical Security of Pseudonymous Signature on the BSI eIDAS Token. Cryptology ePrint Archive, Report 2018/1148, 2018. https://eprint.iacr.org/2018/1148.

[KITD17] Bogdan Kulynych, Marios Isaakidis, Carmela Troncoso, and George Danezis. ClaimChain: Decentralized Public Key Infrastructure. *CoRR*, abs/1707.06279, 2017. http://arxiv.org/abs/1707.06279.

[KJG+16] Eleftherios Kokoris-Kogias, Philipp Jovanovic, Nicolas Gailly, Ismail Khoffi, Linus Gasser, and Bryan Ford. Enhancing Bitcoin Security and Performance with Strong Consistency via Collective Signing. *CoRR*, abs/1602.06997, 2016. https://arxiv.org/pdf/1602.06997.

[KK19] Eleftherios Kokoris-Kogias. Robust and Scalable Consensus for Sharded Distributed Ledgers, 2019. https://eprint.iacr.org/2019/676.

[KKJG+17] Eleftherios Kokoris-Kogias, Philipp Jovanovic, Linus Gasser, Nicolas Gailly, Ewa Syta, and Bryan Ford. OmniLedger: A Secure, Scale-Out, Decentralized Ledger via Sharding. Cryptology ePrint Archive, Report 2017/406, 2017. https://eprint.iacr.org/2017/406.

[KKKZ18] Thomas Kerber, Markulf Kohlweiss, Aggelos Kiayias, and Vassilis Zikas. Ouroboros Crypsinous: Privacy-Preserving Proof-of-Stake. Cryptology ePrint Archive, Report 2018/1132, 2018. https://eprint.iacr.org/2018/1132.

[KKS+17] Yujin Kwon, Dohyun Kim, Yunmok Son, Eugene Y. Vasserman, and Yongdae Kim. Be Selfish and Avoid Dilemmas: Fork After Withholding (FAW) Attacks on Bitcoin. *CoRR*, abs/1708.09790, 2017. https://arxiv.org/abs/1708.09790.

[KLK+19a] Yujin Kwon, Jian Liu, Minjeong Kim, Dawn Song, and Yongdae Kim. Impossibility of Full Decentralization in Permissionless Blockchains, 2019. https://arxiv.org/pdf/1905.05158v1.pdf.

[KLK+19b] Yujin Kwon, Jian Liu, Minjeong Kim, Dawn Song, and Yongdae Kim. Impossibility of Full Decentralization in Permissionless Blockchains, 2019. https://arxiv.org/pdf/1905.05158v2.pdf.

[KLL+18] Vireshwar Kumar, He Li, Noah Luther, Pranav Asokan, Jung-Min (Jerry) Park, Kaigui Bian, Martin B. H. Weiss, and Taieb Znati. Direct Anonymous Attestation with Efficient Verifier-Local Revocation for Subscription System. Cryptology ePrint Archive, Report 2018/290, 2018. https://eprint.iacr.org/2018/290.

[Klu16] Kamil Kluczniak. Domain-Specific Pseudonymous Signatures Revisited. Cryptology ePrint Archive, Report 2016/070, 2016. https://eprint.iacr.org/2016/070.

[KP99] Elias Koutsoupias and Christos H. Papadimitriou. Worst-case Equilibria. In *STACS 99, 16th Annual Symposium on Theoretical Aspects of Computer Science, Trier, Germany, March 4-6, 1999, Proceedings*, pages 404–413, 1999. http://cgi.di.uoa.gr/~elias/papers/paper-kp99.pdf.

[KT18] Max J. Krause and Thabet Tolaymat. Quantification of energy and carbon costs for mining cryptocurrencies, 2018. https://doi.org/10.1038/s41893-018-0152-7.

[Lim19] Calctopia Limited. Raziel Wallet, 2019. https://calctopia.com/app.

[LLX07] Jiangtao Li, Ninghui Li, and Rui Xue. Universal Accumulators with Efficient Nonmembership Proofs. In *Applied Cryptography and Network Security*, pages 253–269, Berlin, Heidelberg, 2007. Springer Berlin Heidelberg. https://www.cs.purdue.edu/homes/ninghui/papers/accumulator_acns07.pdf.

[LMas05] Matt Lepinksi, Silvio Micali, and abhi shelat. Collusion-Free Protocols, 2005. https://shelat.ccis.neu.edu/dl/CollusionFreeSTOC.pdf.

[LSP24] Jacob D. Leshno, Elaine Shi, and Rafael Pass. On the Viability of Open-Source Financial Rails: Economic Security of Permissionless Consensus, 2024. https://arxiv.org/abs/2409.08951.

[LWW04] Joseph K. Liu, Victor K. Wei, and Duncan S. Wong. Linkable Spontaneous Anonymous Group Signature for Ad Hoc Groups, 2004. https://eprint.iacr.org/2004/027.

[M18] Josh M. Code Sample: Intel® Software Guard Extensions Remote Attestation End-to-End Example, 2018. https://software.intel.com/en-us/articles/code-sample-intel-software-guard-extensions-remote-attestation-end-to-end-example.

[MGT18] June Ma, Joshua S. Gans, and Rabee Tourky. Market Structure in Bitcoin Mining, 2018. https://papers.ssrn.com/sol3/papers.cfm?abstract_id=3103104.

[Min18a] Awesome Miner. Awesome Miner's Profit Switching, 2018. https://www.awesomeminer.com/help/profitswitching.aspx.

[Min18b] MinerGate. MinerGate's Smart Mining, 2018. https://minergate.com/about.

[MJMF18] Mohammad Hossein Manshaei, Murtuza Jadliwala, Anindya Maiti, and Mahdi Fooladgar. A Game-Theoretic Analysis of Shard-Based Permissionless Blockchains. *ArXiv e-prints*, page arXiv:1809.07307, September 2018. https://arxiv.org/abs/1809.07307.

[MJW+14] John Maheswaran, Daniel Jackowitz, David Isaac Wolinsky, Lining Wang, and Bryan Ford. Crypto-Book: Bootstrapping Privacy Preserving Online Identities from Social Networks. *ArXiv e-prints*, page arXiv:1406.4053, June 2014. https://arxiv.org/abs/1406.4053.

[MKAP08] Leonardo A. Martucci, Markulf Kohlweiss, Christer Andersson, and Andryi Panchenko. Self-Certified Sybil-Free Pseudonyms. 2008. https://leo.kau.se/pdf/LAMartucci_SelfCertified_SybilFree_Pseudonyms.pdf.

[MSCS12] Gabriel Maganis, Elaine Shi, Hao Chen, and Dawn Song. Opaak: Using Mobile Phones to Limit Anonymous Identities Online, 2012. https://people.eecs.berkeley.edu/~dawnsong/papers/2012%20Opaak%20Using%20Mobile%20Phones%20to%20Limit%20Anonymous%20Identities%20Online.pdf.

[Mul18a] MultiMiner. MultiMiner's Features, 2018. http://www.multiminerapp.com/#features.

[Mul18b] MultiPoolMiner.io. MultiPoolMiner, 2018. https://multipoolminer.io/.

[Nic18] NiceHash. NiceHash Miner, 2018. https://miner.nicehash.com/.

[Nic19] Jack Nicas. Does Facebook Really Know How Many Fake Accounts It Has? *The New York Times*, 2019. https://www.nytimes.com/2019/01/30/technology/facebook-fake-accounts.html.

[Nit09] Rishab Nithyanand. A Survey on the Evolution of Cryptographic Protocols in ePassports. Cryptology ePrint Archive, Report 2009/200, 2009. https://eprint.iacr.org/2009/200.

[OP19] Olamide Omolola and Paul Plessing. Revisiting Privacy-aware Blockchain Public Key Infrastructure, 2019. https://eprint.iacr.org/2019/527.

[otCCAC18] Chinese Office of the Central Cyberspace Affairs Commission. Proposed Regulations for Identity on Blockchains, 2018. http://www.cac.gov.cn/2018-10/19/c_1123585598.htm.

[Par14] Vilfredo Pareto. *Manual of Political Economy: A Variorum Translation and Critical Edition*. Oxford University Press UK, 2014. https://global.oup.com/academic/product/manual-of-political-economy-9780199607952.

[Par18] European Parliament. Directive (EU) 2018/843 on the prevention of the use of the financial system for the purposes of money laundering or terrorist financing, 2018. https://eur-lex.europa.eu/legal-content/EN/TXT/?uri=CELEX:32018L0843.

[PGHR13] Bryan Parno, Craig Gentry, Jon Howell, and Mariana Raykova. Pinocchio: Nearly Practical Verifiable Computation. Cryptology ePrint Archive, Report 2013/279, 2013. https://eprint.iacr.org/2013/279.

[Pid15] Pidong Huang and Manjong Lee. Coordination on Use of Non-deferred Electronic Payment Instrument, 2015. https://pidonghuang.weebly.com/uploads/9/7/3/4/9734342/discrete_5.pdf.

[Pro08] LSE's Identity Project. Analysing the Home Office's May 2008 Identity Cards Cost Report, 2008. http://www.lse.ac.uk/management/research/identityproject/s37Response4.pdf.

[Pro16] Pepper Project. Pequin: An end-to-end toolchain for verifiable computation, SNARKs, and probabilistic proofs. GitHub, 2016. https://github.com/pepper-project/pequin.

[PS16] Rafael Pass and Elaine Shi. The Sleepy Model of Consensus. Cryptology ePrint Archive, Report 2016/918, 2016. https://eprint.iacr.org/2016/918.

[PS18] Rafael Pass and Elaine Shi. Rethinking Large-Scale Consensus. Cryptology ePrint Archive, Report 2018/302, 2018. https://eprint.iacr.org/2018/302.

[PSRK18] Christos Patsonakis, Katerina Samari, Mema Roussopoulos, and Aggelos Kiayias. Towards a Smart Contract-based, Decentralized, Public-Key Infrastructure. Cryptology ePrint Archive, Report 2018/853, 2018. https://eprint.iacr.org/2018/853.

[Res18] IBM Research. AnonCreds: Anonymous credentials protocol implementation in Python, 2018. https://github.com/hyperledger/indy-anoncreds.

[Reu19] Reuters. China imposes blockchain rules to enable 'orderly development', 2019. https://uk.reuters.com/article/us-china-blockchain/china-imposes-blockchain-rules-to-enable-orderly-development-idUKKCN1P41FX.

[Ros73] Robert W. Rosenthal. A class of games possessing pure-strategy Nash equilibria, 1973. https://montoya.econ.ubc.ca/Econ522/Congestion_games_-_Rosenthal.pdf.

[RPKC07] J. Ryou, S. Park, J. Kim, and B. Choi. Anonymous PKI Framework for Privacy-Guaranteed e-Services. In *2007 International Conference on Convergence Information Technology - ICCIT '07(ICCIT)*, pages 687–690, 2007. https://www.computer.org/portal/web/csdl/doi/10.1109/ICCIT.2007.180.

[Rub19] Aleksei Rubin. 2CryptoCalc, 2019. https://2cryptocalc.com/.

[RY15] Leonid Reyzin and Sophia Yakoubov. Efficient Asynchronous Accumulators for Distributed PKI, 2015. https://eprint.iacr.org/2015/718.

[SABBD18] Alberto Sonnino, Mustafa Al-Bassam, Shehar Bano, and George Danezis. Coconut: Threshold Issuance Selective Disclosure Credentials with Applications to Distributed Ledgers. *CoRR*, abs/1802.07344, 2018. https://arxiv.org/abs/1802.07344.

[Sch60] Thomas C. Schelling. The Strategy of Conflict. 132(3418):28–29, 1960.

[SD19] Alberto Sonnino and George Danezis. SybilQuorum: Open Distributed Ledgers Through Trust Networks, 2019. https://arxiv.org/pdf/1906.12237.

[SDH+23] Fabian Schwarz, Khue Do, Gunnar Heide, Lucjan Hanzlik, and Christian Rossow. FeIDo: Recoverable FIDO2 Tokens Using Electronic IDs, 2023. https://publications.cispa.de/ndownloader/files/43248810.

[SFF20] Muhammad Sardar, Rasha Faqeh, , and Christof Fetze. Formal Foundations for Intel SGX Data Center Attestation Primitives, 2020. https://link.springer.com/chapter/10.1007/978-3-030-63406-3_16.

[SJBZ18] Vinnie Scarlata, Simon Johnson, James Beaney, and Piotr Zmijewski. Formal Foundations for Intel SGX Data Center Attestation Primitives, 2018. https://www.intel.com/content/dam/develop/external/us/en/documents/intel-sgx-support-for-third-party-attestation-801017.pdf.

[SJK+16] Ewa Syta, Philipp Jovanovic, Eleftherios Kokoris Kogias, Nicolas Gailly, Linus Gasser, Ismail Khoffi, Michael J. Fischer, and Bryan Ford. Scalable Bias-Resistant Distributed Randomness. Cryptology ePrint Archive, Report 2016/1067, 2016. https://eprint.iacr.org/2016/1067.

[SJSW18] Philipp Schindler, Aljosha Judmayer, Nicholas Stifter, and Edgar Weippl. HydRand: Practical Continuous Distributed Randomness. Cryptology ePrint Archive, Report 2018/319, 2018. https://eprint.iacr.org/2018/319.

[SKG19] Christian Stoll, Lena Klaaßen, and Ulrich Gallersdörfer. The Carbon Footprint of Bitcoin, 2019. https://doi.org/10.1016/j.joule.2019.05.012.

[SKT18] Alexander Spiegelman, Idit Keidar, and Moshe Tennenholtz. Game of Coins. *CoRR*, abs/1805.08979, 2018. https://arxiv.org/abs/1805.08979.

[SM19] Smart-Miner. Smart-Miner, 2019. http://smart-miner.com/.

[SMBW12] Srinath Setty, Richard Mcpherson, Andrew J. Blumberg, and Michael Walfish. Making argument systems for outsourced computation practical (sometimes. In *In NDSS*, 2012. https://www.pepper-project.org/pepper-ndss12.pdf.

[SPW+14] Ewa Syta, Benjamin Peterson, David Isaac Wolinsky, Michael Fischer, and Bryan Ford. Deniable Anonymous Group Authentication, 2014. https://cpsc.yale.edu/sites/default/files/files/TR1486.pdf.

[STC+20] Youren Shen, Hongliang Tian, Yu Chen, Kang Chen, Runji Wang, Yi Xu, Yubin Xia, and Shoumeng Yan. Occlum: Secure and Efficient Multitasking Inside a Single Enclave of Intel SGX, 2020. https://arxiv.org/abs/2001.07450.

[STV+15] Ewa Syta, Iulia Tamas, Dylan Visher, David Isaac Wolinsky, and Bryan Ford. Keeping Authorities "Honest or Bust" with Decentralized Witness Cosigning. *CoRR*, abs/1503.08768, 2015. https://arxiv.org/pdf/1503.08768.pdf.

[TLW13] J. Tsai, N. Lo, and T. Wu. Novel Anonymous Authentication Scheme Using Smart Cards. *IEEE Transactions on Industrial Informatics*, 9(4):2004–2013, Nov 2013. https://ieeexplore.ieee.org/document/6365817/.

[Uni18a] International Telecommunications Union. Abstract Syntax Notation One (ASN.1) Recommendations, 2018. https://www.itu.int/ITU-T/studygroups/com17/languages/.

[Uni18b] International Telecommunications Union. Recommendations X.509, 2018. RecommendationX.509.

[VBMW+18] Jo Van Bulck, Marina Minkin, Ofir Weisse, Daniel Genkin, Baris Kasikci, Frank Piessens, Mark Silberstein, Thomas F. Wenisch, Yuval Yarom, and Raoul Strackx. Foreshadow: Extracting the keys to the Intel SGX kingdom with transient out-of-order execution. In *Proceedings of the 27th USENIX Security Symposium*. USENIX Association, August 2018. https://foreshadowattack.eu/foreshadow.pdf.

[WAPaas18] Liang Wang, Gilad Asharov, Rafael Pass, and Thomas Ristenpart amd abhi shelat. Blind Certificate Authorities. 2018. https://eprint.iacr.org/2018/1022.

[Wha19] WhatToMine. WhatToMine, 2019. https://whattomine.com/.

[Wik20a] Wikipedia. Anti-mask law, 2020. https://en.wikipedia.org/wiki/Anti-mask_law.

[Wik20b] Wikipedia. Freedom of Assembly, 2020. https://en.wikipedia.org/wiki/Freedom_of_assembly.

[Wik20c] Wikipedia. Freedom of Association, 2020. https://en.wikipedia.org/wiki/Freedom_of_association.

[Woo15] Gavin Wood. Proof-of-Authority Private Chains, 2015. https://github.com/ethereum/guide/blob/master/poa.md.

[WPasR16] Liang Wang, Rafael Pass, abhi shelat, and Thomas Ristenpart. Secure Channel Injection and Anonymous Proofs of Account Ownership, 2016. https://eprint.iacr.org/2016/925.

[WSH+14] Riad S. Wahby, Srinath Setty, Max Howald, Zuocheng Ren, Andrew J. Blumberg, and Michael Walfish. Efficient RAM and control flow in verifiable outsourced computation. Cryptology ePrint Archive, Report 2014/674, 2014. https://eprint.iacr.org/2014/674.

[WZC+18] Howard Wu, Wenting Zheng, Alessandro Chiesa, Raluca Ada Popa, and Ion Stoica. DIZK: A Distributed Zero Knowledge Proof System, 2018. https://eprint.iacr.org/2018/691.

[YAXY17] Rupeng Yang, Man Ho Au, Qiuliang Xu, and Zuoxia Yu. Decentralized Blacklistable Anonymous Credentials with Reputation. Cryptology ePrint Archive, Report 2017/389, 2017. https://eprint.iacr.org/2017/389.

[YL] Scott Yilek and Shyong (Tony) K. Lam. Traceable Anonymous Pseudonyms with One TTP. https://pdfs.semanticscholar.org/107a/3e1beaa19c1d83b3cbed637e651ac5871260.pdf.

[ZMR18] Mahdi Zamani, Mahnush Movahedi, and Mariana Raykova. RapidChain: A Fast Blockchain Protocol via Full Sharding. Cryptology ePrint Archive, Report 2018/460, 2018. https://eprint.iacr.org/2018/460.