encointer - An Ecological, Egalitarian and Private Cryptocurrency and Self-Sovereign Identity System

Alain Brenzikofer
alain@encointer.org

***Abstract*—*encointer* proposes a new blockchain based cryptocurrency with an ecological consensus mechanism using trusted execution environments and an egalitarian money supply policy. Money issuance is done through a universal basic income subject to a proof-of-personhood. Individuals attending randomized pseudonym key signing events obtain such proofs. *encointer* also features private transactions and scalable, trustless off-chain smart contracts.**

***Index Terms*—cryptocurrency, macroeconomics, identity management, location awareness, privacy, energy efficiency**

I. Motivation

A. Economics

With the appearance of Bitcoin [1] in 2008, a big socio-economic experiment took off. The nature of money itself was widely debated. Bitcoin adopts a hard-coded nominally inflatonary monetary policy saturating at a fixed supply. Rapid adoption made Bitcoin a real deflationary currency, which it will remain if successful. Early adopters made a fortune. Because of its deflationary nature, bitcoin favors accumulation of capital for the few. Wealth increases without work.

The monetary policy followed by central banks issuing national fiat money on the other hand often follows the goal of price stability, aiming at a moderate inflation goal in the order of 1-2%. Issuance of money is appointed to banks who give credit to companies who employ workers who consume goods and thereby make companies profitable and raise the GDP. A process that allegedly benefits everyone. However, the observation that an increase in money supply doesn't benefit everyone equally is referred to as the Cantillon-Effect [22]. Thomas Piketty shows [23] that gains on capital historically exceed economic growth, another factor that questions the trickle-down theory.

encointer aims at turning this logic upside-down and lets all individuals issue money subject to common rules. In order to mint *encointer* , people need to attend to pseudonym key signing parties (meetups) that happen at regular intervals at high sun all over the world within small randomized groups of people [2]. The *encointer* issuance therefore represents a form of *universal basic income* (UBI) for every person attending such meetups.

These *encointer* meetups are at the same time the basis of a self-sovereign identity claim called proof-of-personhood (PoP) [3], proving a one-to-one relationship between a person and her digital identity. One person can only maintain one individuality claim because ceremonies are designed to make it impossible to attend two meetups physically as they happen in different places concurrently.

B. Unpermissioned Consensus

Bitcoin and many other cryptocurrencies use an energy-hungry consensus mechanism called proof of work (PoW). While PoW has been the key idea that made Bitcoin possible in the first place, it is not ecologically sustainable. Moreover, it failed its goal of decentralization as mining has become centralized by a single company in a single country.

Peercoin [4] introduced the first proof-of-stake (PoS) cryptocurrency in 2012. Until today, PoS is not academically respected as a sound consensus mechanism [5]. While PoW makes a compromise on energy efficiency, PoS makes the compromise of benefitting the rent-seeking wealthy.

Accepting that there's always a compromise to make, *encointer* introduces dPoET; a permissionless version of proof-of-elapsed-time (PoET) [6], relying on trusted execution environments (TEE). PoET requires trust in vendor attestation services. Currently, there are few TEE vendors on the market (i.e. Intel SGX [7], ARM trust zone [9] used by AMD, Qualcomm and others) but there are also open-source hardware initiatives that might one day diversify the attestation trust.

C. Private Smart Contracts

In 2015 Ethereum [16] was introduced, bringing turing-complete smart contracts to the blockchain. Ethereum now serves as a platform for many decentralized applications (DApps) and has become the major ecosystem for ICOs. While enabling publicly verifyable smart contracts, there is no way to process private data on Ethereum in trustless manner. Public unpermissioned validation of smart contracts is only possible with a minimum of public inputs. Support for zk-SNARKS has been added to the etheremun virtual machine with the metropolis hard-fork, yet this only enables to verify zero knowledge proofs that have been generated off-chain. It is therefore possible to hide the payload of a smart contract call, but as you have to call the contract by means of a public transaction, your pseudonym is leaking. Quorum [17], an Ethereum fork, approaches privacy by delegating smart contract validation to a small group of permissioned validators using a BFT consensus and allowing Zcash-style shielded transactions to hide your pseudonym when calling a private smart contract. Such setups do not allow GDPR-compliant DApps [18], [19]. The reasoning is the following: In order for the DApp to comply with GDPR, the Dapp has to be

run by a single operator having users opting into his privacy terms. The operator would then have to run all private contract validators by himself. Such centralization would render the use of blockchain meaningless.

encointer enables private, decentralized DApps. As the internal state of a TEE is not leaking, they offer a way to run smart contracts with private, encrypted inputs while still offering verifyability. This has been demonstrated by Hyperledger Sawtooth Private Data Objects (PDO) [20]. PDOs allow to take state and execution of smart contracts off chain, thereby also improving scalability as compared to Ethereum.

D. Transaction Privacy

Bitcoin transactions are pseudonymous but not anonymous. It has been shown that identities of transacting parties can be rgithubevealed [24]. Aiming at transaction privacy, Monero was introduced in 2014, employing the CryptoNote protocol [11]. Receiving funds in Monero means scanning every block for transactions to oneself. This task can only be taken out by full nodes as delegating it would leak private information.

Zcash was introduced in 2016 empoying the Zerocash protocol [13] using zk-SNARKS to hide sender, receiver and value from third parties. Generating SNARKS to send funds is a computationally heavy process, limiting its usability for mobile and IoT devices.

For both Monero and ZCash, privacy comes at the price of large transaction size, letting the blockchain grow quickly. Beeing equipped with private smart contracts, *encointer* only stores tx hashes onchain.

E. Scalability

Because of its block size limit, Bitcoin can only reach about 4-7 transactions per second onchain. In order to tackle Bitcoin's scalability issues, Lightning Network payment channels [14] were introduced in 2015 and demonstrated in 2017. Scalability is achieved by bilaterally treating transactions off-chain with the option to settle the last balance at any time on-chain. Teechan [15] was introduced in 2017, implementing payment channels in TEEs.

encointer takes transactions and smart contract execution off-chain altogether. Only one hash per transaction must be stored onchain. The latest state is shared among validators but only the hash of that state needs to be onchain. This improves scalability by an order of magnitude and delays the need for second layer solutions.

F. Governance

Decentralized blockchain governance has in the past been tried by various means. In the case of Bitcoin, a balance of power between miners and coin holders decides about the future of the protocol, which lead to multiple chain forks in the past, hurting the ecosystem and dividing development teams.

PoS blockchains delegate governance to their whales. Who has more coin has more say. This poses a conflict of interest i.e. in the case of deciding the future nominal inflation.

As *encointer* has an anti-sybil attack measure in place (PoP), a one-person-one-vote (1p1v) scheme could be implemented. However, there is a conflict of interest as well because low-wealth individuals are expected to vote in favor of higher inflation as they directly benefit in the short term due to their role as money issuers in *encointer* .

encointer suggests a novel approach to blockchain governance, delegating blockchain governance to a swiss cooperative holding the *encointer* trademark. Cooperative membership is open to anyone staking the equivalent of *tbd* CHF. The cooperative suggests protocol updates in advance, including changes of nominal inflation rate, tx tax burn rate, block size limit. Suggestions by the cooperative can be blocked by a referendum vote requiring a majority of 2/3 of stake and 2/3 majority of 1p1v voters. Balloting happens onchain anonymously. The 2/3 majority threshold for referendums allows the cooperative to react quickly to changing circumstances but still provide decentralization, given large opposition.

If the *encointer* cooperative should fail to suggest necessary changes, the community may suggest changes as well. They also require a majority of 2/3 of stake and 2/3 majority of 1p1v voters.

II. Monetary Policy

Unlike Bitcoin, *encointer* doesn't have a hard-capped supply. The more people are joining the ecosystem, the more money is issued. The value of one *encointer* is therefore tied to the willingness of people to spend time to attend key signing parties. As median wages vary greatly between nations but *encointer* is a global unit of account, it is expected and intended that ceremonies are more economically attractive to attend in low-wage countries. Also, the possibility of obtaining a digital identity might be especially beneficial in developing countries where people might own a mobile phone but not a state-issued ID.

The total coin supply $M0$ after K parties with N_i total participants for party i can be expressed as

$$M0 = \sum_{i=0}^{K} N_i - D \tag{1}$$

D being the total of burned ceremony deposits.

Exponential community growth causes exponential growth of money supply. If community participation should one day saturate, money issuance will be constant and inflation rate will therefore decrease over time as shown in figure 1.

With such a policy in place, no early adoper should expect to get rich by hoarding *encointer* as adoption drives inflation.

A. Money Issuance

Every PoP-ceremony participant will receive one *encointer* per attended ceremony. To avoid discontinuities, issuance happens evenly during the time between two subsequent ceremonies.

B. Local Currencies

Besides a global money issuance, *encointer* allows to issue local currencies. The location of the first registered meetup of a region defines that region's *seed*. All nearby meetups will

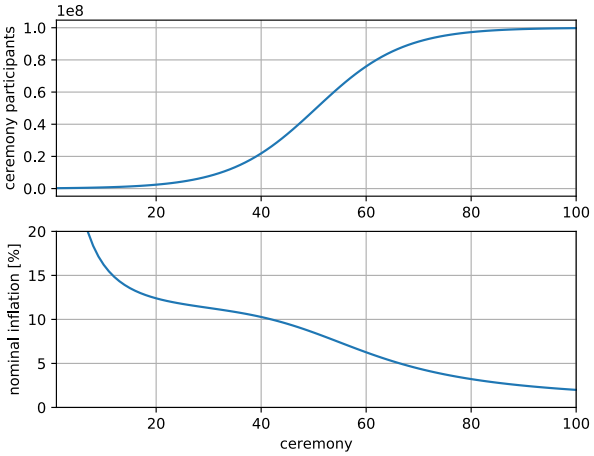

Fig. 1. Nominal inflation rate per ceremony for a fictional participation curve following an S-curve.

be linked to that *seed*. *encointer* creates a new currency for every *seed* and provides a decentralized exchange for all these local currencies.

If a meetup brings together people who minted different seeds during the last ceremony, that meetup will allow to mint the majority seed.

III. PoP Ceremonies

Key signing ceremonies will be scheduled every 41 days. The interval of 41 days is an arbitrary design choice. In order for many people to be able to participate, it should be at different weekdays every time and it shouldn't happen too often - but often enough that missing one ceremony doesn't hurt too much. All meetups will happen at high sun at the same date all over the world. This is crucial because we want nobody to be able to attent two meetups for the same ceremony, because this would allow a single person to maintain more than one PoP identity (a Sybil attack).

A. Preparation

At least 24h before the first meetup of a ceremony, participant a creates a registration transaction for ceremony i containing

$K^{pub}_{a,i}$	one-time public key
$S_{s,j}$	a ring signature proving ceremony $j < i$ was attended with group s last time. The ring consisting of all participants of ceremony j at location $L_{s,j}$.
$L'_{a,i}$	his/her anticipated approximate position as a geohash [21]
$r_{a,i}$	acceptable distance range to meetingpoint in [km]
$n^{min}_{a,i}$	minimum number of counterparties
$d_{a,i}$	minimum deposit for counterparties
$S(tx)$	a deposit in *encointer* to be redeemed after attendance

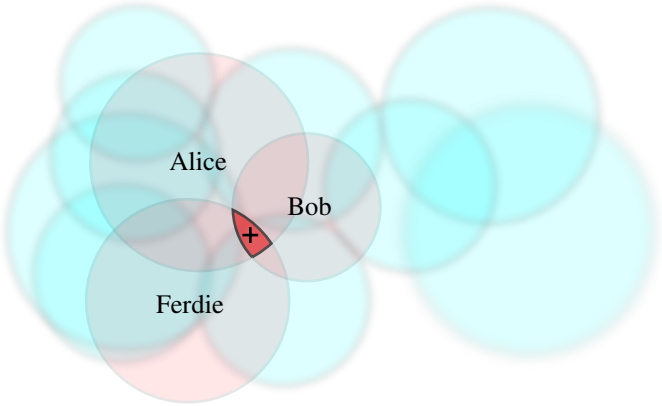

Fig. 2. All participants with their acceptable range. Highlighted in red is one scheduled meetup among Alice, Bob and Ferdie with a suggested location in the intersection of their ranges

The system then tries to find randomized matching counterparties as shown in Figure 2. A meetup can take place if all of the following conditions are met:

- at least 3 attendees with overlapping acceptable range, at most 12.
- at most 2/3 of all participants have met at a meetup for the last ceremony already

The lower limit to meetup size of 3 participants shall provide some safety (see III-E5). The upper limit of 12 attendees shall make sure that meetups can be taken out within short time and to allow each participant to remember who she/he has already signed keys with.

24h before the meetup, the system defines the location anchors $\hat{L}_{s,i}$ for all groups s. It will maximize the distance between any two location anchors subject to individual acceptable ranges. The group is then free to agree on any meeting place $L_{s,i}$ within a range and a tolerance range $r_{s,i}$ by setting up a private chat provided by the *encointer* app. Meeting time $T_{s,i}$ is set precisely to high sun for $\hat{L}_{s,i}$ longitude.

No meetups with more than 12 people will be scheduled to make sure every attendee can remember who he/she already signed with.

B. Meetup Procedure

Key signing must begin within $\Delta t_{s,i}$ after $T_{s,i}$. No latecomers may be accepted.

$$\Delta t_{s,i} = \frac{min\left(dist\left(L_{s,i}, Lk, i\right)\right)}{v_{max}} \forall k \in S \tag{2}$$

Where $v_{max} = 300km/h$ is an *encointer* parameter set to make it impractical to attend two adjacent ceremonies, even when using a helicopter.

Attendees attest the physical presence of all counterparties with their own reputation.

Meetup participants use the *encointer* app to scan QR codes shown on each participants screen. The QR code initiates a bluetooth pairing and a sequence of challenge response involving multiple physical layers follows. Participants are

guided through a process involving stacking phones on top of each other to allow acoustical coupling as well as timing challenges over NFC. Sensor fusion allows to estimate the probability that every two participants devices are indeed in the same place. The following sensors and actors are involved:

- location services (GPS, cellular, WLAN)
- screen (actor)
- camera (sensor)
- loudspeaker (actor)
- microphone (sensor)
- vibration motor (actor)
- accelerometer (sensor)
- magnetic field (sensor)
- gyroscope (sensor)

C. Ceremony Scalability

The most densely inhabitated city is Manila with $43'000/km^2$ [29]. With a limit of 10 people per meetup and a participation of 100% , 4300 meetups would take place per km^2. Each meetup would be allowed an area of $232m^2$ and the tolerance range would be $r_{s,i} \approx 15m$. This is impractical as such a small range might be inaccessible to the public. $\delta_T s, i$ would be as low as $180ms$, which is achievable by automated witnessing over i.e. bluetooth. To avoid trespassing attempts, a group s may file a relocation request with $\hat{L}'_{s,i}$ signed by at least 2/3 of members in s.

D. Proof-of-Personhood

Attending one meetup supplies the individual with a non-bijective proof-of-personhood. It doesn't prove that one individual maintains exactly one id. More than half of all ceremonies must be attended in order to get a valid, bijective PoP claim and be eligible for coin issuance.

E. Attacks and Mitigation

1) Illegit Videoconference: People may try to meet virtually instead of physically.

2) Surrogates: An adversary might pay other people to attend ceremonies on behalf of identities controlled by the adversary. The effect is similar to people renting out their identity. This can't be prevented by means of a pseudonym party.

3) Social Engineering: Attendees might talk others into signing more than one pseudonym. Bribery could happen too.

4) Systematic No-Show: a meeting might become invalid if too many participants don't show up. No shows are punished by burning the deposit.

5) Threats to Personal Safety: As ceremony members need to meet in person, all risks involved with human encounters apply. These risks are reduced by randomizing participants and by the minimal group size of 3 persons. Participants are advised to choose public places for ceremonies. Threats by non-participants who want to hurt the *encointer* ecosystem by attacking participants are mitigated if group s keeps their exact meeting point private.

6) mutual-signing off-shore bot communities: The above rules allow many unconnected communities to grow independently all over the world which accelerates adoption. This opens an attack vector of an adversary creating virtual communities in the middle of the pacific where no real person will likely ever show up. These bot communities could perform meetups in isolation. Countermeasures include live chat Turing tests with participants of other meetups or additional participants at another longitude (and therefore ceremony meeting time) joining the ceremony virtually by means of videoconference.

IV. dPoET Consensus

The major technical innovation of Bitcoin is its use of PoW to avoid double spends in a decentralized digital cash scheme. The higher the cost of a double spend attack, the more secure the Bitcoin blockchain. The security of Bitcoin relies on mining power. Miners invest in infrastructure and energy as long as this remains profitable. Mining power therefore depends on the real value of miner income being proportional to Bitcoin exchange rate.

PoS blockchain security on the other hand relies on game theory. Gaining control over the majority of stake to attack a PoS chain would destroy the real value of that stake, hurting the attacker most. PoS blockchain security therefore is proportional to its total market capitalization. A yet unsolved issue with PoS is the *nothing-at-stake* dilemma [5] which may cause uncontrollable forks.

PoET has similar security properties like PoW as it is just an energy-efficient replacement for PoW to elect the validator who may write the next block to the blockchain. Each validator TEE samples a uniform random number $u \in [0..1)$ and waits for an exponentially distributed wait time defined by:

$$t = -\frac{ln(1-u)}{\lambda} \tag{3}$$

λ has to be adjusted to validator population size to reach a defined average block time, similar to Bitcoin's difficulty adjustment.

Everyone with a supported TEE hardware can join the validator set and gets equal probability of being selected to generate the next block. Therefore, the 51% (double spend) attack applies to PoET as well. The difference is just that it's not 51% of PoW mining power but 51% of TEE devices in the network.

encointer makes such an attack very expensive by demanding that every validator must be linked to a unique person. Every person maintaining her PoP may run at most one validator.

Each *encointer* validator node waits a random time before generating a new block according to PoET. It then assembles transactions into a block including a PoET and broadcasts the block to the network.

In bitcoin, all nodes believe in the blockchain accumulating the most PoW since genesis. In *encointer* , nodes believe in the blockchain accumulating the highest PoET difficulty $\sum_i \frac{1}{\lambda_i}$.

We call this consensus algorithm *democratic proof of elapsed time* (dPoET)

A. Settlement Finality

Like PoW, dPoET only delivers probabilistic transaction finality.

V. Trusted Execution Environment Security

TEEs aim to provide the necessary guarantees for secure remote computation. They should provide integrity and confidentiality guarantees when executing software on a computer maintained by an untrusted party. The most recent TEEs rely on software attestation, a process that guarantees the user that she's communicating with a known piece of code running inside a secure container on a genuine trusted hardware by means of a manufacturer signature.

As criticized in [7], manufacturers seem to follow a security by obscurity principle not disclosing design internals necessary for a proper security review. Their *in dubio contra reum* analysis of Intel SGX shows vulnerabilities to cache timing and sidechannel attacks. *Foreshadow* [8] falsified confidentiality as well as integrity claims for SGX but the attack is mitigated for now. ARM TrustZone on the other hand is only an IP core and design details are left to the manufacturer, equally reluctant to disclose details.

Since at least the post-Snowden era, one also has to be concerned about manufacturers being forced by their state to introduce deliberate backdoors. Even if open-source TEEs like Keystone [27] might soon deliver devices, one would still have to trust the manufacturer not to tamper with the design.

While all this is disturbing, it should be put in perspective. Information security is a never-ending race. All blockchain solutions are software running by large part on Intel CPUs. While hardware wallets may give us some comfort concerning our funds private keys, there's no guarantee on confidentiality when considering sidechannel attacks.

The *encointer* cooperative will follow developments closely and maintain an up to date list of accepted TEE manufacturers' attestation keys. Even if the author is not satisfied with today's TEEs security guarantees, he considers the ecological downsides of PoW to be much more severe for society as a whole and therefore proposes to make heavy use of TEEs for *encointer* .

VI. Proportional Transaction Fees

Bitcoin transactions may pay a fee to the miner including the transaction in his block. Because Bitcoin has a hard cap on block size, a market develops for fees and miners choose the best-paying transactions to fill a block. This effect prevents micropayments effectively undermining adoption in developing countries. While IOTA [25] and Nano [26] have zero transaction fees they must use a small PoW as a measure against spam transactions. Even though small, PoW limits the possibilities to use mobile or IoT devices to send transactions.

encointer suggests yet another way: Each transaction must show a PoET of 0.5 seconds as an anti-spam measure. Nowadays, even mobile devices feature a TEE and are therefore

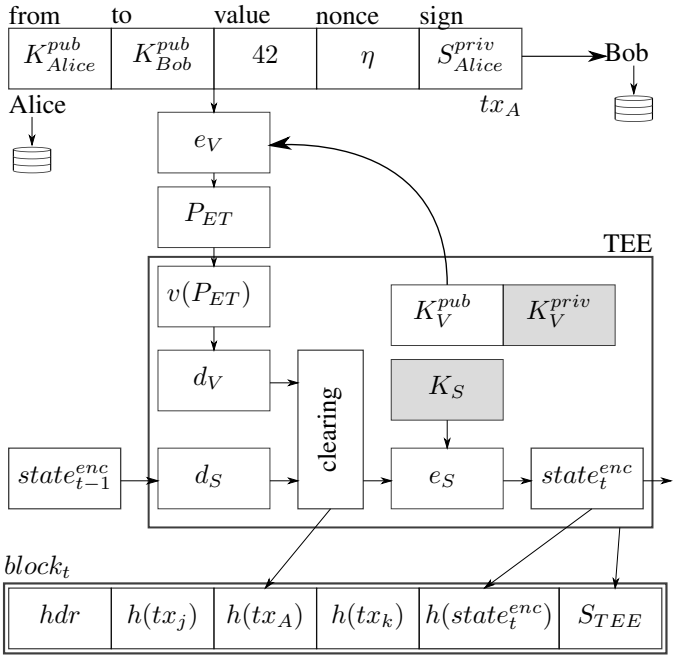

Fig. 3. simplified private transaction scheme

able to provide a PoET. Requiring a TEE to send transactions comes with a nice side benefit: The TEE can be used as a hardware wallet for private keys, effectively mitigating the weakest link in blockchain security by solving the safekeeping of private keys for hot wallets.

Still, validator nodes need an incentive for *encointer* to benefit from a large decentralized network of validators. In the spirit of the Tobin Tax [10] it seems beneficial to the *encointer* ecosystem to charge a proportional transaction fee in the order of 0.1% to incentivize validator nodes to process transactions. In the presence of indirect block size limitations like TEE memory size, a proportional fee may still inhibit micropayments as validators might select high-value transactions with priority. That is left to be solved when we get there.

encointer targets a balance of power among multiple TEE vendors as it is unlikely that different vendors collude or show the same vulnerabilities. Such a balance of power must be incentivized in order to take place. One possible incentive would be to burn a fraction of tx fees that is proportional to the network share of the validor's TEE vendor. Minority TEEs would earn more fees than majority TEEs.

VII. Private Transactions

Fig 3 shows how transactions are processed in encrypted form. Alice creates a transaction to Bob and encrypts it with the shared verifier key K^{pub}_V. She then appends a PoET P_{ET} and sends the tx to the network. If her device has no TEE, she can delegate the PoET to any third party, whereby she only shares the encrypted tx with the PoET provider.

The validator retrieves the most recent $state_{t-1}$ (i.e. from IPFS [30]), assembles transactions from its mempool and sends everything to its TEE. The TEE is able to decrypt

the data to process the clearing. It then writes hashes of a transaction to the new block together with the hash of the encrypted new $state_t$ and a signature, proving which validator created the block. The new block only contains hashes, so no private information is leaking.

In order for Bob to know he received funds, Alice sends him the plaintext tx_A over a private messaging channel. Bob can then hash it and scan the blockchain for $h(tx_A)$. Both Alice and Bob then must make sure to store tx_A securely and redundantly as they can't retrieve their wallet balance by scanning the blockchain.

A. Enclave Provisioning

The shared symmetric state encryption key K_S and asymmetric transaction submission key K_V^{priv} is known to all registered validators. Whenever the validator set changes, a new key K_V^{priv} has to be established among all validators' enclaves who can then derive K_S. A distributed key generation scheme like [32] or [**?**] or shall be applied for that purpose. The shared key is expected to change frequently, thereby improving forward-secrecy in the case of side-channel attacks.

Registering a new enclave for chain validation includes the following steps:

1) initialize *encointer* enclave
2) perform remote attestation with a trusted attestation service (registered on the *encointer* chain)
3) commit attestation service quote to *encointer* chain
4) all existing validators see that the validator set changed. They validate the quote and take out a new distributed key generation.

VIII. Private Off-Chain Smart Contracts

The concept of private transactions explained above can be extended to allow private calls of generic stateful smart contracts.

encointer separates transaction validation consensus from smart contract execution. Unlike Ethereum, *encointer* doesn't need to execute a smart contract on every validator machine. Thanks to TEEs, it would be sufficient to run smart contracts on one TEE only, as correct execution is guaranteed. For resilience, it is of course desirable to have more than one contract TEE.

encointer Dapps provide their own TEE infrastructure in order to provide their services. This way, they're responsible for termination properties of their contracts without any impact on the *encointer* main chain. Metered execution like *gas* is not necessary, opening up a new range of possibilites.

Fig 4 shows how *encointer* transactions allow for an optional payload Λ which gets included in a call table $calls_t$. *encointer* hereby provides a sequential ordering of smart contract calls and all the information needed to process the call is included in $calls_t$, as shown in Fig. 5.

It is up to the Dapp to provide TEEs with their contract key K_C. Dapp users should make sure the contract code is open source and K_C is a manufacturer-attested TEE output. Dapps can be stateful and their hashed state can be anchored

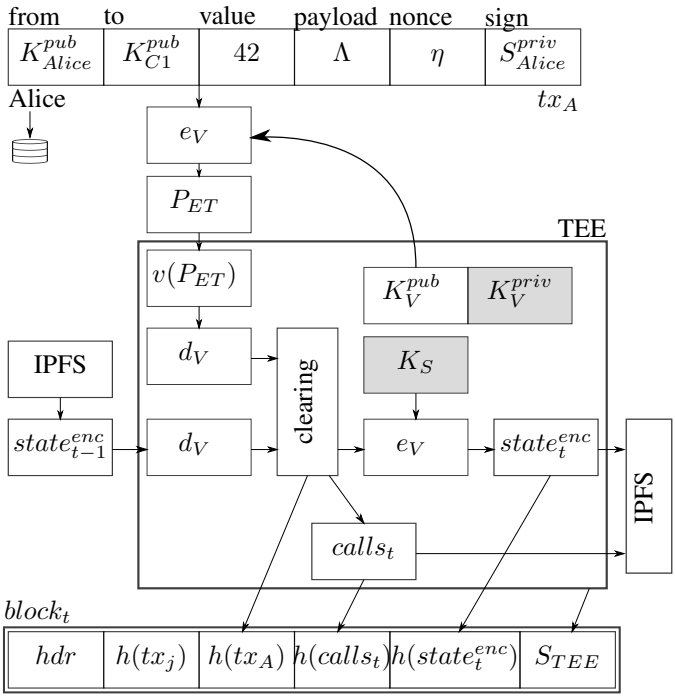

Fig. 4. simplified smart contract call scheme

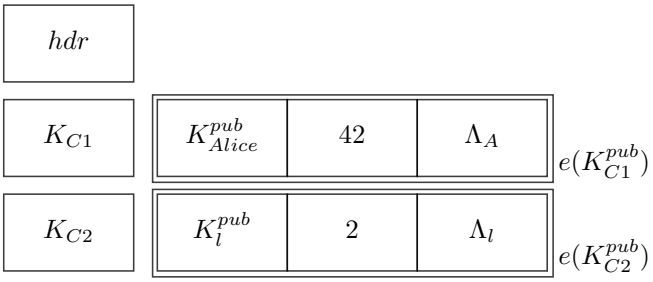

Fig. 5. calls table structure

in *encointer* chain by including a K_C signed transaction $tx_{C,t}$ to itself. $tx_{C,t}$ should be published in cleartext so anyone can verify execution.

Related concepts were presented by [33], [20]

IX. *encointer* Cooperative

encointer cooperative is a not-for-profit cooperative under swiss law. Its purpose is to govern the *encointer* ecosystem. It fulfills the following tasks

- community bootstrapping
- protocol updates
- define tx fee
- maintain list of accepted TEE attestation service keys

All changes are subject to a referendum by the community.

A. Community Bootstrapping

In order to establish liquidity and value for the *encointer* token quickly, *encointer* will run a one-way exchange allowing users to exchange their *encointer* for some established cryptocurrency *tbd*.A minimum ceremony participation of 2 out of

the last 3 will be required. The exchange will be funded with a *tbd* amount per ceremony by the *encointer* cooperative, funded by donations. The amount you get per *encointer* depends on how many *encointer* will be traded.

To bootstrap a community in a new region, a minimum of 9 participants is needed because of ceremony rules defined in III-A.

X. Conclusion

A novel cryptocurrency system has been introduced in conceptual detail. Main contributions are

- A novel global egalitarian approach to monetary policy allowing for a universal basic income (UBI).
- A new definition of trustless pseudonym key signing parties for proof-of-personhood (PoP), incentivized by *encointer* tokens.
- A novel unpermissioned consensus algorithm dPoET, combining PoET with PoP to achieve decentralization of power by ecological means
- Private token transfers with microtransaction-friendly fees and low storage footprint.
- Scalable trustless private off-chain smart contracts

References

[1] Satoshi Nakamoto. Bitcoin: A peer-to-peer electronic cash system, http://bitcoin.org/bitcoin.pdf, 2008
[2] Bryan Ford. Pseudonym Parties: An Offline Foundation for Online Accountability, 2008
[3] Maria Borge et al. Proof-of-Personhood: Redemocratizing Permissionless Cryptocurrencies
[4] Sunny King, Scott Nadal. PPCoin: Peer-to-Peer Crypto-Currency with Proof-of-Stake, 2012
[5] Andrew Poelstra. "Distributed Consensus from Proof of Stake is Impossible"
[6] Hyperledger. PoET 1.0 Specification. https://sawtooth.hyperledger.org/docs/core/releases/1.0/architecture/poet.html
[7] V. Costan S. Devadas. Intel SGX Explained. Tech. rep., Cryptology ePrint Archive, 2016.
[8] Jo Van Bulck et.al. Foreshadow: Extracting the Keys to the Intel SGX Kingdom with Transient Out-of-Order Execution, 2018
[9] Introducing ARM TrustZone. https://developer.arm.com/technologies/trustzone
[10] J. Tobin. The New Economics One Decade Older. The Eliot Janeway Lectures on Historical Economics in Honour of Joseph Schumpeter, Princeton University Press, 1972
[11] Nicolas van Saberhagen, CryptoNote v 2.0, https://cryptonote.org/whitepaper.pdf, 2014
[12] B. Bnz, J. Bootle, D. Boneh, A. Poelstra, P. Wuille, Greg Maxwell. Bulletproofs: Short Proofs for Confidential Transactions and More, IEEE S&P 2018
[13] Eli Ben-Sasson, Alessandro Chiesa, Christina Garman, Matthew Green, Ian Miers, Eran Tromer, Madars Virza, Zerocash: Decentralized Anonymous Payments from Bitcoin, proceedings of the IEEE Symposium on Security & Privacy (Oakland) 2014, 459-474, IEEE, 2014
[14] Poon, Joseph. The Bitcoin Lightning Network: Scalable Off-Chain Instant Payments, 2015
[15] J. Lind, I. Eyal, P. Pietzuch, E. Gn Sirer. Teechan: Payment Channels Using Trusted Execution Environments
[16] Vitalik Buterin, Gavin Wood, Joseph Lubin. Ethereum Whitepaper, https://github.com/ethereum/wiki/wiki/White-Paper, 2014
[17] JP Morgan Chase. Quorum. https://github.com/jpmorganchase/quorum
[18] Regulation (EU) 2016/679 of the European Parliament and of the Council of 27 April 2016 on the protection of natural persons with regard to the processing of personal data and on the free movement of such data, and repealing Directive 95/46/EC (General Data Protection Regulation). Official Journal of the European Union, Vol. L119 (4 May 2016), pp. 1-88
[19] European Union Blockchain Observatory and Forum, Blockchain and the GDPR
[20] Hyperledger Sawtooth Private Data Objects. https://github.com/hyperledger-labs/private-data-objects
[21] Gustavo Niemeyer. Geohash. https://geohash.org
[22] Richard Cantiollon. Essai sur la Nature du Commerce en Gnral, 1755
[23] Thomas Piketty. Capital in the Twenty-First Century, 2013
[24] Fergal Reid. An Analysis of Anonymity in the Bitcoin System, Security and Privacy in Social Networks, 2012
[25] Serguey Popov. The Tangle, http://iotatoken.com/IOTA_Whitepaper.pdf, 2016
[26] Colin LeMahieu. Nano: A Feeless Distributed Cryptocurrency Network, 2016
[27] Keystone Project, https://keystone-enclave.github.io/
[28] https://openpowerfoundation.org/
[29] https://en.wikipedia.org/wiki/List_of_cities_by_population_density, sampled Nov. 2018
[30] Juan Benet. IPFS - Content Addressed, Versioned, P2P File System. 2014
[31] Yuh-Min Tseng, A Robust Multi-Party Key Agreement Protocol Resistant to Malicious Participants, 2005
[32] R. Gennaro, S. Jarecki, H. Krawczyk, and T. Rabin, Secure distributed key generation for discrete-log based cryptosystems, International Conference on the Theory and Applications of Cryptographic Tech- niques. Springer, 1999, pp. 295310.
[33] Raymond Cheng et al., Ekiden: A Platform for Confidentiality-Preserving, Trustworthy, and Performant Smart Contract Execution, arXiv:1804.05141, 2018