Private Attacks in Longest Chain Proof-of-stake Protocols with Single Secret Leader Elections

Private Attacks in Longest Chain Proof-of-stake Protocols with Single Secret Leader Elections

Sarah Azouvi
Protocol Labs

Daniele Cappelletti
Politecnico di Torino

ABSTRACT

Single Secret Leader Elections have recently been proposed as an improved leader election mechanism for proof-of-stake (PoS) blockchains. However, the security gain they provide has not been quantified. In this work, we present a comparison of PoS longest-chain protocols that are based on Single Secret Leader Elections (SSLE) – that elect exactly one leader per round – versus those based on Probabilistic Leader Elections (PLE) – where one leader is elected on expectation. Our analysis shows that when considering the private attack – the worst attack on longest-chain protocols [14] – the security gained from using SSLE is substantial: the settlement time is decreased by $\sim$ 25% for a 33% or 25% adversary. Furthermore, when considering grinding attacks, we find that the security threshold is increased by 10% (from 0.26 in the PLE case to 0.36 in the SSLE case) and the settlement time is decreased by roughly 70% for a 20% adversary in the SSLE case.

ACM Reference Format:
Sarah Azouvi and Daniele Cappelletti. 2021. Private Attacks in Longest Chain Proof-of-stake Protocols with Single Secret Leader Elections. In *3rd ACM Conference on Advances in Financial Technologies (AFT '21), September 26–28, 2021, Arlington, VA, USA.* ACM, New York, NY, USA, 13 pages. https://doi.org/10.1145/3479722.3480996

1 INTRODUCTION

Proof-of-stake has been proposed as a more energy-efficient alternative consensus protocol to proof-of-work for cryptocurrencies. In proof-of-work, miners need to solve a computational puzzle in order to earn the right to create a block and receive the associated financial rewards. The amount of blocks that they mine is, hence, proportional to their computational power. In contrast, the idea behind PoS is that participants mine a fraction of blocks that is proportional to the relative amount of coins they own. One crucial component of PoS consensus protocols is their *leader election* [2], used to decide which participants will get to create the next block. Although leader election protocols have been studied widely in the traditional field of distributed systems, the setting of blockchains – which are decentralized and provide financial rewards for block creation – poses new challenges that are paramount to the security of the whole consensus protocol. For example, leader election protocols must be fair, in the sense that miners must be elected proportionally to their power (stake or compute resources); this is to ensure they are compensated fairly for their investment and to prevent Sybil attacks. They should also be private, i.e., no actor should be able to guess the next leader until they broadcast their block with a proof-of-eligibility; this prevents denial-of-service (DoS) attacks against the next leader. Many attempts have been made to design an adequate leader election protocol, such as using hash functions [2, 11] or coin tossing protocols [18]. Another method that has been adopted by many protocols [3, 12, 12, 16] is the use of verifiable random functions (VRFs) [20].

Most leader election protocols used in blockchains are private but probabilistic [11, 12, 16], meaning that one leader will privately be elected on expectation, but it could be that zero or several leaders are elected in some rounds. Probabilistic Leader Elections (PLEs) can be problematic since having multiple leaders elected in a round leads to different views or *forks* in the system. Ouroboros [18] proposed a leader election where exactly one leader is elected per round. However, the election is not private and exposes the leader to DoS attacks.

To solve these problems, Single Secret Leader Election (SSLE), where *exactly* one leader is privately elected at each round, has been proposed in [8, 10]. Although current SSLEs are more complex to implement than, for example, PLEs based on VRFs, they intuitively improve the security of PoS blockchains because they reduce the probability of honest forks. However, there is no formal proof of this statement and, if true, the exact gain in security that they achieve, compared to PLEs, has yet to be quantified.

Our contribution. In this work, we perform a comparison between SSLEs and PLEs and investigate the gain in security against private attacks, where the adversary grows a private chain of blocks so as to outpace the honest chain. Focusing our analysis on private attacks, and not general adversaries, is motivated by the work of Dembo et al. [14], who showed that private attacks are the worst attack in longest-chain blockchains, in the sense that the true security threshold of a longest-chain protocol is the same as the security threshold against private attacks.

We start our analysis by considering leader elections that have access to a perfect source of randomness in Section 4, before considering randomness that is derived from the blockchain itself – as is the case for many PoS longest-chain protocols [3, 12, 12] – and the resulting grinding attack in Section 5. We find that, with perfect randomness, the persistence parameter (or settlement time) against private attacks in a synchronous network is decreased by roughly 25% against a 33% or 25% adversary. In the grinding case, the security threshold is higher by 10 percentage points in the SSLE case (36%) than in the PLE case (26%) and the persistence parameter is decreased by roughly 70%. Although it is not surprising that SSLEs perform better than PLEs in longest-chain blockchains, we did not expect to see such significant improvements.

AFT '21, September 26–28, 2021, Arlington, VA, USA

ACM ISBN 978-1-4503-9082-8/21/09…$15.00
https://doi.org/10.1145/3479722.3480996

These results are very encouraging and could motivate the switch from PLE to SSLE in current PoS blockchains. Since SSLEs are still less efficient than PLEs, quantifying the gain in security is paramount for evaluating fairly the trade-offs between the two. We leave a full analysis of PoS with SSLE in a partially synchronous network and against all possible attacks as future work. Whether with SSLE or PLE, obtaining an exact formula for the security of the protocol against any adversary (i.e., the probability of breaking the consensus) is an open and non-trivial problem. All the known analyses that consider a general adversary are based on bounds that would not be useful for our comparison, and hence a general adversary is beyond the scope of this work.

Lastly, we note that our analysis does not make use of the secret property of the SSLE (that protects against DoS attacks), and thus the results would hold for a public single leader election as well.

2 RELATED WORK

Previous works have proposed formal analyses of PoS blockchains based on PLEs [3, 7, 13]. In most of them, the proof revolves around finding a *special* block that guarantees security of the protocol and bounding the probability that this type of block does not appear in a sequence of n consecutive blocks. For example, Kiayias et al. [17] consider *Catalan* blocks, Bagaria et al. [3] and Deb et al. [13] consider Nakamoto blocks, Daian et al. [11] consider *pivot* point. In [3], the interpretation given for that special block is that no private chain started by the adversary at any point in time before that block will be able to catch up with any chain that includes that block at any point in the future. They show that any attack can be modelled as a composition of private attacks.

Dembo et al. [14] considered three different types of longest-chain blockchain protocols: proof-of-work, proof-of-stake, and proof-of-space, and showed that in all three cases the true security threshold of the protocol (i.e., the security threshold when considering all attacks) is exactly the same as the security threshold of the private attack. They also prove that the private attack is the worst attack in the synchronous proof-of-work case. In the case of PoS protocols, based on the fact that other attacks do not appear in the security threshold, they conjecture that these attacks should have at least the same exponent as the private attack. Even though this conjecture is left open, we decide in this paper to focus on the private attack and leave a full analysis as future work. While this limits the generality of our work, we believe that, due to the conjecture above, studying the private attack alone gives a good proxy for the security of PoS longest-chain protocols. Furthermore, this simplifying assumption allows us to get exact bounds, as opposed to the loose bounds obtained by other analyses [3, 7, 11, 13], usually based on Chebyshev's inequality or similar inequalities. Loose bounds on the probability of successful attacks would not be sufficient to meaningfully quantify the difference between SSLE and PLE in longest-chain PoS blockchains as even if one bound is smaller than the other, this does not say anything about how the exact probabilities compare. Simplifying our model to only consider private attacks alleviates this problem as it allows us to get closed-form solutions that we are meaningfully able to compare.

3 BACKGROUND, MODEL, AND DEFINITIONS

We start by giving some background on PoS blockchains before presenting our model. Because we limit our adversary to private attacks, we will consider a rather simplistic model and abstract away most of the blockchain concepts.

3.1 Proof-of-stake blockchains

A blockchain is a digital distributed ledger of transactions. Transactions are grouped into *blocks* that are chronologically ordered in a linked chain. In a PoS system, participants – also called miners or validators– are eligible to create blocks based on their relative stake, as measured from the number of coins that they own. In longest-chain protocols, eligible miners create their block and append it to the longest chain of blocks that they are aware of, i.e., the new block should link to the block atop the longest existing chain of blocks.

Briefly, the protocol works as follows. Whenever miners receive a block, they add it to their local view. At each round, they run a leader election protocol that randomly decides their eligibility for that round, proportionally to their stake. For now we assume access to a perfect *random beacon* [5] that emits a random number at the beginning of each round. This beacon is then given as an input to the leader election. If they are elected in that round, miners create a block on top of their longest chain and broadcast their block to the network. In the case where there exist two chains of the same length, e.g., if more than one leader was elected in the previous round, miners break the tie in a random way. In our model, we will ignore the content of the blocks as well as associated reward or financial incentives.

Informally, a blockchain should verify two security properties [14]: (1) *persistence*, meaning that once a block has been confirmed by an honest miner, it should stay in the chain of that miner indefinitely i.e., it is not possible for an adversary to revert the chain of an honest miner, except for the last k blocks; and (2) *liveness*, meaning that new blocks should be appended to the chain continually, even in the presence of an adversary.

One important parameter in longest-chain blockchains is the *persistence parameter* – or settlement time – k, which, informally, represents the number of rounds after which an honest miner will consider a block confirmed, i.e., after which it will stay in the chain forever.

3.2 Private attacks

Private attacks are a specific type of attack on longest-chain blockchains, in which an adversary keeps its blocks private (instead of sending them to the rest of the miners) and does not mine on other participants' blocks. In other words, the adversary is creating its own chain, parallel to the honest chain. The adversary succeeds if it can create a private chain longer than the honest chain. If that is the case, it will broadcast its chain to the rest of the players, forcing them to abandon their own chain and mine on top of the adversarial chain. It is clear that this attack will almost surely be successful if the adversary can mine blocks at a rate faster than the rest of the players. In PoS blockchains, this rate is proportional to the relative amount of stake that one owns. Hence this attack will succeed if the adversary owns more than half of the total stake. If not, then

there exists a time after which this attack becomes very unlikely to succeed.

3.3 Model

3.3.1 Assumptions. We consider a set of N participants, a fraction α of which are controlled by an adversary $\mathcal{A}$ performing a private attack as specified above. The remaining participants are *honest* and follow the protocol. Time is divided into discrete time-steps. We assume that we have access to a broadcast algorithm and we consider a synchronous model meaning that each message, i.e., block, sent by an honest participant reaches everyone after at most Δ time steps with $\Delta > 0$. We consider a round-based protocol that relies on an underlying leader election mechanism as defined below. We assume that the duration of each round is strictly more than Δ, such that any message sent by any honest participant at the beginning of a round will be received before the end of that round. We assume that we have access to a perfect random beacon that emits new randomness at the beginning of each round. We will relax this assumption in Section 5.

For simplicity, we consider a flat model, meaning that each miner accounts for one unit of stake and that the set of participants is static.

3.3.2 Leader Election. Every PoS protocol uses a leader election to decide which miners are eligible to create blocks. In this paper, we treat it as a black box algorithm that takes as input a random number r and a set of participants and outputs a (potentially empty) set of leaders. We consider two types of leader election: Single Secret Leader Election (SSLE) [8], where *exactly* one leader is elected per round, and Probabilistic Leader Election (PLE) [16], where one leader is elected per round *on expectation*. This means that, when using a PLE, there could be multiple leaders or no leader at all in a round. We denote a_n and h_n the number of adversarial and honest leaders elected at round n, respectively.

We model the PLE case as follows: at each round, every player "tosses their own coin" (using the randomness given by the random beacon) to determine their eligibility; each player wins with probability $1/N$. In practice, this is achieved using a verifiable random function [20]: each player uses the VRF to compute their own random number; if it falls below a threshold, they are elected leader. An adversary controlling a fraction α of the players will thus have a number of leaders that follows a Binomial distribution with $N \times \alpha$ trials and success probability $1/N$, seeing as they get to toss a coin for each of the players they control. We assume that the number of participants N is big and hence the number of adversarial leaders elected can be approximated as a Poisson distribution with parameter α. Similarly, the number of honest participants elected in a round follows a Poisson distribution with parameter $1 - \alpha$. Furthermore, the number of adversarial and honest leaders are independent from each other.

In the SSLE case, exactly one of the players is elected, hence the number of adversarial leaders follows a Bernoulli distribution with parameter α. However, the number of adversarial and honest leaders are not independent. In particular, the number of honest leaders is the complement of the number of adversarial leaders, i.e., $h_n = 1 - a_n$ for every $n \in \mathbb{N}^*$.

In this paper, where we consider a static adversary that performs a private attack, we ignore some of the practical requirements for leader elections in PoS blockchains (such as unpredictability and secrecy) that are not relevant to the model.

3.3.3 Security games. We consider an adversary mounting a private attack against the honest players. Accordingly, we define the following games that capture whether or not the adversary succeeds in a private attack according to the assumptions above.

Definition 3.1 ((L, α)-PLE Private Game). The PLE private game with parameters (L, α) is defined as follows: at each round $n \in [1, \dots, L]$ a number a_n of adversarial leaders and h_n of honest leaders are selected at random from, respectively, Poisson distributions of parameters α and $1 - \alpha$. We say that the adversary wins the PLE private game of length L and power α if the number of rounds with non-zero adversarial leaders is greater than or equal to the number of rounds with non-zero honest leaders, i.e.:

$$|\{n \in [1, \dots, L] : a_n > 0\}| \geq |\{n \in [1, \dots, L] : h_n > 0\}|.$$

Definition 3.2 ((L, α)-SSLE Private Game). The SSLE private game with parameters (L, α) is defined as follows: at each round $n \in [1, \dots, L]$, exactly one leader is elected. This leader is adversarial ($a_n = 1, h_n = 0$) with probability α and honest ($a_n = 0, h_n = 1$) with probability $1 - \alpha$. We say that the adversary wins the SSLE private game of length L and power α if the number of rounds with adversarial leaders is greater than or equal to the number of rounds with honest leaders, i.e.:

$$|\{n \in [1, \dots, L] : a_n = 1\}| \geq |\{n \in [1, \dots, L] : h_n = 1\}|.$$

We now define the persistence parameter n_0 of the games, parametric in $0 < \epsilon < 1$, that intuitively represents the number of rounds after which the adversary cannot win the private game, except with probability ϵ.

Definition 3.3 (ϵ−persistence parameter). We say that n_0 is the ϵ−persistence parameter of the SSLE, resp. PLE, private game if the probability that there exists any $n \geq n_0$ such that the adversary wins the SSLE, resp. PLE, game of length n is ϵ.

In order to study the private games, we define the concept of gap, already introduced by Blum et al. [7].

Definition 3.4 (Gap). The gap at round $n \in [1, \dots, L]$ is the difference between the number of adversarial rounds and honest rounds in rounds 1 to n. Let $\mathcal{G}_n^{SSLE}(\alpha)$ and $\mathcal{G}_n^{PLE}(\alpha)$ denote, respectively, the gap in the PLE and SSLE private games of parameters (L, α). For $n \in [1, \dots, L]$, we have:

$$\mathcal{G}_n^{SSLE}(\alpha) = |\{i \in [1, \dots, n] : a_i = 1\}| - |\{n \in [1, \dots, n] : h_i = 1\}|$$

$$\mathcal{G}_n^{PLE}(\alpha) = |\{i \in [1, \dots, n] : a_i > 0\}| - |\{n \in [1, \dots, n] : h_i > 0\}|$$

If we consider a general analysis that applies to both settings (SSLE and PLE), we simply write $\mathcal{G}_n$ and talk about the private game. It is clear that the adversary wins the private game of length n if and only if $\mathcal{G}_n \geq 0$. In the next section, we will study the behaviour of the gap. We are specifically interested in the probability that the adversary wins the PLE and SSLE games for any $n \geq n_0$, $n_0 \in \mathbb{N}$.

Before this, we briefly explain why the PLE and SSLE private games are an accurate description of the private attack in PoS

systems. In the SSLE case, since there is exactly one leader per round and the network is synchronous, it is clear that the honest chain will be exactly the same length as the number of honest rounds and the adversarial chain will be at most the same length as the number of adversarial rounds.

In the PLE case, however, there could be honest forks due to multiple honest leaders being elected in the same round. If that happens, and since we consider a synchronous network, the longest chain will still increase by one even if some of the blocks at that round are being abandoned. Even in the worst case where there are multiple longest chains for several rounds, each of them will still be as long as the number of honest rounds. Similarly, in the adversarial case, even if the adversary has more than one block on one round, it can only append one block per round and, hence, its longest chain is bounded by the number of eligible rounds.

4 ANALYSIS

In this section we prove our main theorems, Theorem 4.5 and 4.6, where we express for $n_0 \in \mathbb{N}^*$ the probability that an adversary succeeds in winning the private game for any length greater than or equal to n_0. This probability corresponds to the value ϵ for the corresponding ϵ−persistence parameter n_0. We will then compare the ϵ−persistence parameter in the SSLE and PLE cases.

SSLE and PLE games as biased random walks. In the SSLE game, it is straightforward to see that the gap will increase by one with probability α and decrease by one with probability $1-\alpha$.

In the PLE game, there are two events in which the gap will not change. The first event is when no leader is elected. The second event is when an honest leader is elected at the same round as an adversarial leader. We call these events *null events* and denote p_0 the probability that they happen.

On the other hand having multiple adversarial leaders and no honest leader is equivalent to having exactly one adversarial leader since the gap will grow by one at that round regardless of the exact number of adversarial leaders and vice versa for honest leaders. We note $p_a = \Pr[a_n > 0 \text{ and } h_n = 0]$ the probability that an adversary is the unique leader in a round - which does not depend on n - and $p_h = \Pr[a_n = 0 \text{ and } h_n > 0]$ the probability that the honest players are unique leaders. The PLE game can be modeled as a random walk that increases by one with probability p_a, decreases by one with probability p_h and stays the same with probability $p_0 = 1 - p_a - p_h$. Since a_n and h_n are independent, we have:

$$\begin{aligned} p_a &= \Pr[h_n = 0] \times \Pr[a_n \geq 1] \\ &= e^{\alpha-1}(1 - e^{-\alpha}) \\ &= e^{\alpha-1} - e^{-1} \end{aligned}$$

And similarly: $p_h = e^{-\alpha} - e^{-1}$.

We now move on to prove our first lemma. For the rest of the paper, for $p \in (0,1)$ and $n \in \mathbb{N}$, we note

$$\mathsf{Bin}(p,n,k) = \begin{cases} \binom{n}{k} p^k (1-p)^{n-k}, & \text{for } k \in \mathbb{N} \\ 0, & \text{for } k \in \mathbb{R} \setminus \mathbb{N} \end{cases}$$

Lemma 4.1. *For every $(n,v) \in \mathbb{N}^2$ and $\alpha \in (0,1)$:*

$$\Pr[\mathcal{G}_n^{SSLE}(\alpha) = v] = \mathit{Bin}(\alpha, n, \frac{1}{2}(n+v)).$$

Proof. We already noted that $(\mathcal{G}_n^{SSLE}(\alpha))_{n\in\mathbb{N}}$ is a random walk such that

$$\mathcal{G}_{n+1}^{SSLE}(\alpha) = \begin{cases} \mathcal{G}_n^{SSLE}(\alpha) + 1, & \text{with probability } \alpha \\ \mathcal{G}_n^{SSLE}(\alpha) - 1, & \text{with probability } 1-\alpha \end{cases}$$

The proof of the lemma follows from standard results on random walks and can be found in [1]. Briefly, for $n \in \mathbb{N}$: $-n \leq \mathcal{G}_n^{SSLE}(\alpha) \leq n$. We note u the number of times that $\mathcal{G}_n^{SSLE}(\alpha)$ increased by one and d the number of times that $\mathcal{G}_n^{SSLE}(\alpha)$ decreased by one. If $\mathcal{G}_n^{SSLE}(\alpha) = v$ for $v \in [-n, n]$, we have $u - d = v$ and $u + d = n$ hence $u = \frac{1}{2}(v+n)$. There are exactly $\binom{n}{u}$ different ways to reach v, starting from 0 and hence $\Pr[\mathcal{G}_n^{SSLE}(\alpha) = v] = \binom{n}{u}\alpha^u(1-\alpha)^{n-u}$. □

We now look at the equivalent lemma, in the PLE case.

Lemma 4.2. *For every $(n,v) \in \mathbb{N}^2$ and $\alpha \in (0,1)$:*

$$\Pr[\mathcal{G}_n^{PLE}(\alpha) = v] = \sum_{l=0}^{n-v} \mathit{Bin}(p_0, n, l) \Pr[\mathcal{G}_{n-l}^{SSLE}(\frac{p_a}{1-p_0}) = v]$$

Proof. If $\mathcal{G}_n^{PLE}(\alpha) = v$, there can be between 0 and $n - v$ null slots (i.e., slots where the gap does not change from the previous step); hence, we have:

$$\begin{aligned} \Pr[\mathcal{G}_n^{PLE}(\alpha) = v] = \sum_{l=0}^{n-v} \Pr[\mathcal{G}_n^{PLE}(\alpha) = v | \#\text{ null events} = l] \times \\ \Pr[\#\text{ null events} = l] \end{aligned}$$

We start by assuming that there exist exactly l null events in the PLE game. After removing the l null events from the PLE game, the remaining $n-l$ slots are either fully adversarial or fully honest; this is equivalent to an SSLE game of length $n-l$. The power of the adversary in this new SSLE game needs to be adjusted, accounting for the fact that null events have been removed. Hence the equivalent SSLE game has parameters $n-l$ and $\frac{p_a}{1-p_0}$ due to the independence of each round. We thus have the following:

$$\Pr[\mathcal{G}_n^{PLE}(\alpha) = v] = \sum_{l=0}^{n-v} \Pr[\mathcal{G}_{n-l}^{SSLE}(\frac{p_a}{1-p_0}) = v] \times \Pr[\#\text{ null events} = l]$$

Following standard results on Binomial distribution we have that $\Pr[\#\text{ null events} = l] = \mathsf{Bin}(p_0, n-v, l)$, hence:

$$\Pr[\mathcal{G}_n^{PLE}(\alpha) = v] = \sum_{l=0}^{n-v} \Pr[\mathcal{G}_{n-l}^{SSLE}(\frac{p_a}{1-p_0}) = v] \times \mathsf{Bin}(p_0, n-v, l).$$

This result can also be derived using the multinomial distribution. □

Having computed the probability that the gap is equal to some value v, and in order to study the persistence parameter, we are also interested in the probability that the gap, starting at some negative value $-M$, goes back up to zero. In blockchain terms, if the adversarial chain is behind the honest chain by M blocks, we compute the probability that it eventually catches back.

Lemma 4.3. *For $\alpha < 1/2$, if in round $n_0 \in \mathbb{N}$, $\mathcal{G}_{n_0} = -M$ for $M > 0$, then the probabilities r_M that $\mathcal{G}_n$ ever reaches 0 for any $n \geq n_0$ in the SSLE and PLE cases are:*

$$r_M^{SSLE} = \left(\frac{\alpha}{1-\alpha}\right)^M ; \; r_M^{PLE} = \left(\frac{e^\alpha - 1}{e^{1-\alpha} - 1}\right)^M$$

Proof. We start by considering the PLE case. The probability r_M that $\mathcal{G}_n^{PLE}$ reaches 0 starting from a position $-M$ for $M > 0$ is the same as the probability that $\mathcal{G}_n^{PLE}$ ever reaches M when starting at 0 (i.e., $\mathcal{G}_n^{PLE}$ has a net increase of M).

Let's note $r_1 = r$. Then we have $r_M = r^M$ ($\mathcal{G}_n^{PLE}$ needs to have M net increase of 1). Furthermore, we have $r = p_a + p_h r^2 + p_0 r$ since $\mathcal{G}_n^{PLE}$ either increases straight away by one (with probability p_a), or decreases by one (with probability p_h) in which case $\mathcal{G}_n^{PLE}$ needs to increase by 2 to have a net increase of 1, or $\mathcal{G}_n^{PLE}$ stays the same (with probability p_0) in which cases $\mathcal{G}_n^{PLE}$ still needs to increase by 1.

We have $r = p_a + p_h r^2 + p_0 r \iff p_a + p_h r^2 + (p_0 - 1)r = 0$, with $p_a + p_h + p_0 = 1$. Hence, r satisfies: $p_a + p_h r^2 - (p_a + p_h)r = 0$. The solutions to this equation are $(1, p_a/p_h)$. Because $\alpha < 1/2$, the random walk is transient with drift towards $-\infty$, hence $r < 1$. This means that $r = p_a/p_h$ and $r_M = \left(\frac{p_a}{p_h}\right)^M = \left(\frac{e^{\alpha-1}-e^{-1}}{e^{-\alpha}-e^{-1}}\right)^M = \left(\frac{e^{\alpha}-1}{e^{1-\alpha}-1}\right)^M$.

The analysis in the SSLE case works the same but with $p_0 = 0$, $p_a = \alpha$ and $p_h = 1 - \alpha$. We then have $r = \alpha + (1-\alpha)r^2$. The two solutions of this equation are 1 and $\frac{\alpha}{1-\alpha}$. Since $r < 1$, we have: $r_M = \left(\frac{\alpha}{1-\alpha}\right)^M$.

□

Interestingly, since $x \mapsto (e^x - 1)(1-x)$ is increasing on $[0, 1/2]$, we notice that for $\alpha < 1/2$, $r_M^{PLE} < r_M^{SSLE}$. Starting from $-M$, the adversary is, hence, more likely to catch up the honest chain in the SSLE case than in the PLE case. However, this does not say anything about whether the adversary is more likely to win the private game of length n as one process may be decreasing faster than the other. We shall now compute the expected value of the gap in both cases to get a sense of their evolution.

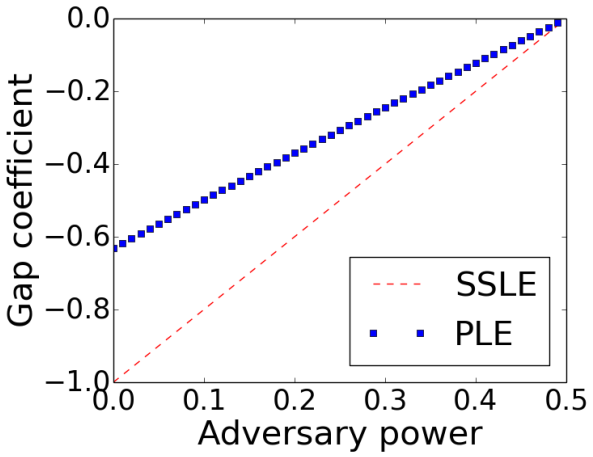
Intuitively, since longest-chain protocols have been proven secure for an adversary that has less than half of the power [14], the gap should decrease with time. The longer the chain is, the harder it is for an adversary to catch up with the honest chain and hence the bigger their disadvantage is, and hence their gap. In the next lemma, we prove that the expected gap is linear in n and that the linear coefficient is bigger for the SSLE than PLE gap, and, therefore, the disadvantage of the adversary grows faster in the SSLE case, consistently with the intuition that SSLE is more secure. Figure 1a show this coefficient for different values of $\alpha < 1/2$.

Lemma 4.4. *For every $n \in \mathbb{N}$ and $\alpha \in (0, 1)$:*

$$\mathbb{E}[\mathcal{G}_n^{SSLE}(\alpha)] = (2\alpha - 1)n$$
$$\mathbb{E}[\mathcal{G}_n^{PLE}(\alpha)] = (e^{\alpha-1} - e^{-\alpha})n$$

Proof. Let $(R_n)_{n\in\mathbb{N}}$ denote a random walk that has a probability p of going up by 1, a probability q of going down by 1 and a probability $1-p-q$ of staying the same in each round. Additionally we assume $R_0 = 0$.

It is trivial to verify that $X_n = R_n - (p-q)n$ is a martingale. Since martingales have constant expectations, we have $\mathbb{E}[X_n] = X_0 = 0$ and hence $\mathbb{E}[R_n] = (p-q)n$. We apply this result to $\mathcal{G}_n^{SSLE}$ and $\mathcal{G}_n^{PLE}$. In the SSLE case, $p - q = \alpha - (1-\alpha) = 2\alpha - 1$ and in the PLE case $p - q = p_a - p_h = e^{\alpha-1} - e^{-\alpha}$. This proves the result. □



(a) Linear Coefficient for the Expected Gap

We now move on to prove the main result that gives the probability of success of the adversary in the SSLE game.

Theorem 4.5. *The probability that the adversary wins the SSLE game for any length greater or equal than n is:*

$$P_{SSLE}^{\alpha}(n) = \sum_{\substack{v=0,\\ n+v\equiv 0[2]}}^{n} Bin(\alpha, n, \frac{1}{2}(n+v)) + \sum_{\substack{v=1,\\ n-v\equiv 0[2]}}^{n} Bin(1-\alpha, n, \frac{1}{2}(n-v))$$

Proof. For readability, in this proof we note $\mathcal{G}_n$ instead of $\mathcal{G}_n^{SSLE}$. We start by noting that $-n \le \mathcal{G}_n \le n$ for every $n \in \mathbb{N}$. There are two scenarios where the adversary can have a private chain longer than the honest chain for a length of at least n: either the gap at n is positive, or the gap is negative and the adversary catches up in the future (i.e., the gap reaches 0 in the future). This gives us the following:

$$\begin{aligned}
P_{\text{SSLE}}^{\alpha}(n) &= \Pr[\mathcal{G}_n \ge 0] + \Pr[\mathcal{G}_n < 0]\Pr[\mathcal{G}_n \text{catches up} | \mathcal{G}_n < 0] \\
&= \sum_{v=0}^{n} \Pr[\mathcal{G}_n = v] + \sum_{v=1}^{n} \Pr[\mathcal{G}_n = -v] r_v \\
&= \sum_{\substack{v=0,\\ n+v\equiv 0[2]}}^{n} \binom{n}{\frac{1}{2}(n+v)} \alpha^{\frac{1}{2}(n+v)} (1-\alpha)^{\frac{1}{2}(n-v)} \\
&+ \sum_{\substack{v=1,\\ n-v\equiv 0[2]}}^{n} \binom{n}{\frac{1}{2}(n-v)} \alpha^{\frac{1}{2}(n-v)} (1-\alpha)^{\frac{1}{2}(n+v)} \left(\frac{\alpha}{1-\alpha}\right)^{v} \\
&= \sum_{\substack{v=0,\\ n+v\equiv 0[2]}}^{n} \binom{n}{\frac{1}{2}(n+v)} \alpha^{\frac{1}{2}(n+v)} (1-\alpha)^{\frac{1}{2}(n-v)} \\
&+ \sum_{\substack{v=1,\\ n-v\equiv 0[2]}}^{n} \binom{n}{\frac{1}{2}(n-v)} \alpha^{\frac{1}{2}(n+v)} (1-\alpha)^{\frac{1}{2}(n-v)} \\
&= \sum_{\substack{v=0,\\ n+v\equiv 0[2]}}^{n} \text{Bin}(\alpha, n, \frac{1}{2}(n+v)) + \sum_{\substack{v=1,\\ n-v\equiv 0[2]}}^{n} \text{Bin}(1-\alpha, n, \frac{1}{2}(n-v))
\end{aligned}$$

□

We now move on to prove the result in the PLE case.

Theorem 4.6. *The probability that the adversary wins the PLE private game for any length greater or equal than n is:*

$$P^{\alpha}_{PLE}(n) = \sum_{v=0}^{n}\sum_{l=0}^{n-v} Bin(p_0,n,l)\,Bin(\frac{p_a}{1-p_0}, n-l, \frac{1}{2}(n-l+v))$$
$$+ \sum_{v=1}^{n}\sum_{l=0}^{n-v} Bin(p_0,n,l)\,Bin(\frac{p_a}{1-p_0}, n-l, \frac{1}{2}(n-l-v))\left(\frac{e^{\alpha}-1}{e^{1-\alpha}-1}\right)^{v}$$

Proof. We use the same technique as in the previous theorem.

$$P^{\alpha}_{\mathrm{PLE}}(n) = \Pr[\mathcal{G}_n \geq 0] + \Pr[\mathcal{G}_n < 0]\Pr[\mathcal{G}_n \text{catches up} | \mathcal{G}_n < 0]$$
$$= \sum_{v=0}^{n}\Pr[\mathcal{G}_n = v] + \sum_{v=1}^{n}\Pr[\mathcal{G}_n = -v] r_v$$
$$= \sum_{v=0}^{n}\sum_{l=0}^{n-v}\mathsf{Bin}(p_0,n,l)\Pr[\mathcal{G}^{SSLE}_{n-l}(\frac{p_a}{1-p_0}) = v]$$
$$+ \sum_{v=1}^{n}\sum_{l=0}^{n-v}\mathsf{Bin}(p_0,n,l)\Pr[\mathcal{G}^{SSLE}_{n-l}(\frac{p_a}{1-p_0}) = v]\left(\frac{e^{\alpha}-1}{e^{1-\alpha}-1}\right)^{v}$$
$$= \sum_{v=0}^{n}\sum_{l=0}^{n-v}\mathsf{Bin}(p_0,n,l)\mathsf{Bin}(\frac{p_a}{1-p_0}, n-l, \frac{1}{2}(n-l+v))$$
$$+ \sum_{v=1}^{n}\sum_{l=0}^{n-v}\mathsf{Bin}(p_0,n,l)\mathsf{Bin}(\frac{p_a}{1-p_0}, n-l, \frac{1}{2}(n-l-v))\left(\frac{e^{\alpha}-1}{e^{1-\alpha}-1}\right)^{v}$$

□

Results interpretation. In order to compare these two probabilities, we plot them for different values of n and α. In Figure 2, we see that, as expected, SSLE performs much better than PLE: the adversary wins the PLE game with higher probability than the SSLE game. To cite a few concrete examples, for $n = 300$ and a 33% adversary, the probability of success drops from 10^{-7} to 10^{-9} (Figure 2d). For $\alpha = 0.33$ and $\epsilon = 10^{-12}$ the persistence parameter is $n_0 = 400$ in the SSLE case and 550 in the PLE case. For a blockchain where one block is emitted every 30 seconds, breaking persistence with probability 10^{-12} would roughly occur once every million years.

For all the values of α that we plotted, we see that, at least for n big enough, the probability of violating persistence is exponential (as the graph is linear in logarithmic scale), meaning that it is of the form e^{-an} for $a \in \mathbb{R}^*_+$ in both the PLE and SSLE cases – for different values of a that we denote a^{SSLE} and a^{PLE}. We remark that this form is consistent with the bound found by Gazi et al. [15] and Li et al. [19] in the context of Bitcoin and is tighter than the one of $e^{-\Omega(\sqrt{n})}$ from Dembo et al. [14] which is based on a looser inequality (as they consider every attack possible).

For a fixed ϵ and large n, the persistence parameter of the SSLE game, n_0^{SSLE}, can thus be expressed as follows: $n_0^{SSLE} = \frac{a_{PLE}}{a_{SSLE}} n_0^{PLE}$, where n_0^{PLE} is the ϵ-persistence parameter of the PLE game. Finding the value of a_{PLE} and a_{SSLE} is straightforward by computing a specific value of ϵ for some big enough n: $a = -\ln(\epsilon)/n$. We find that the ϵ−persistence parameter decreases by 17% for a 49% adversary, by roughly 25% for a 33% or 25% adversary, and by 32% for a 10% adversary in the SSLE case compared to the PLE case. The improvement is substantial.

5 GRINDING ANALYSIS

In the previous section, we have assumed that the random beacon given as input to the leader election at each round came from a perfect source of randomness. Such an assumption can be achieved with a decentralized random beacon [21], for example. However most PoS protocols rely on an internal random beacon, which is based on the state of the chain. Such random beacons are vulnerable, to some extent, to grinding attacks where an adversary grinds through the block space in order to bias the randomness and find a value for which an unfair advantage can be extracted. Bagaria et al. [3] studied the security of PoS protocols where the randomness is updated every c blocks and the trade-offs between updating the randomness more frequently (and being more vulnerable to grinding) or less frequently (and being more predictable). In this section, we study this specific case and compare the security guarantees of PoS blockchains against private attacks that use grinding in the PLE and SSLE case. We start by presenting this new model in Section 5.1 before moving on to the analysis in Section 5.2.

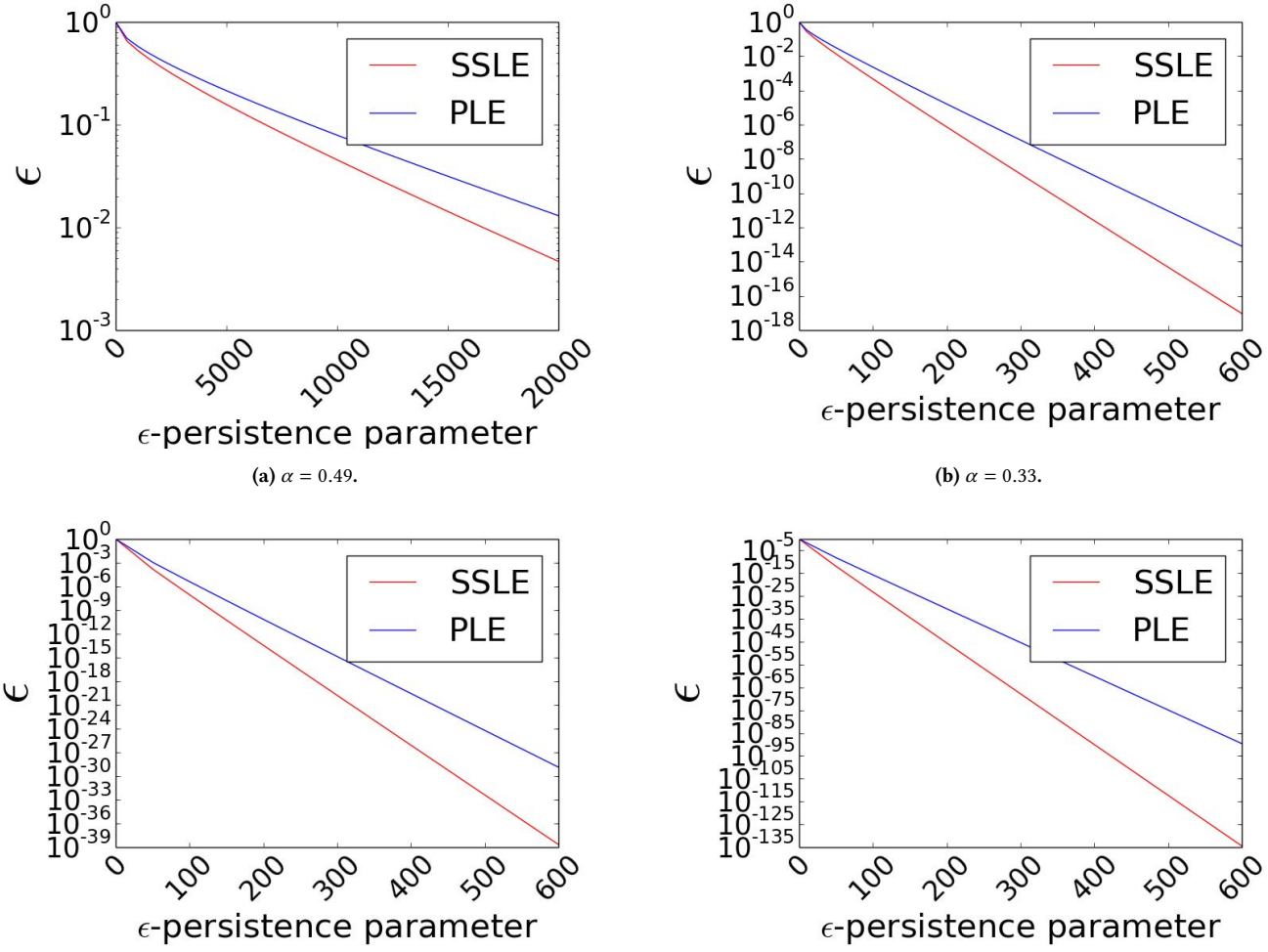
5.1 Model

5.1.1 Random Beacon and Grinding. We assume that the random beacon given as input to the leader election in each round is based on the data in the block created in the previous round. A block created by different miners or a round skipped will all produce different beacons and thus election results in the next round but two blocks created by the same leader at the same round with different content (e.g., different transaction sets) will produce the same random beacon.

As a concrete, simplified, example, we assume that a random beacon r_0 is initialized using a multiparty computation protocol [4, 5, 9] at round 0. Randomness at round i is then defined as $r_i = \sigma_{sk}(r_{i-1} \oplus i)$, where σ is the deterministic signature of the player creating the block in round i. In practice a Verifiable Random Function [20] will be used instead of a signature scheme but this does not matter for our analysis. If no block is created at round $i-1$ because the elected leader was offline or no leader was elected (in the PLE case), then the previous randomness is used, i.e., $r_i = \sigma_{sk}(r_{i-2} \oplus i)$.

This scenario corresponds to $c = 1$ in [3] and it could be extended to the more general case where the randomness is updated every c blocks instead (i.e., $r_i = \sigma_{sk}(r_{i-(i\%c)} \oplus i)$) as Bagaria et al. [3] showed. In the case where $c > 1$ the grinding is more limited but the protocol is more predictable which is an undesirable property (that we ignore in this work). As far as grinding is concerned $c = 1$ corresponds to the worst-case scenario.

The grinding works as follows: whenever the adversary is elected leader, it can decide to publish its block or skip the round, thus biasing the randomness. Furthermore, in the PLE case (where the adversary can potentially produce more than one block per round), the adversary could decide which of its blocks to use, if it was elected more than once, or simply skip this round. By trying out different combinations of blocks or skipping rounds, the adversary



(a) $\alpha = 0.49$.

(b) $\alpha = 0.33$.

(c) $\alpha = 0.25$.

(d) $\alpha = 0.1$.

Figure 2: ϵ−persistence parameter for an adversary with power α (logarithmic scale).

may find a particular chain where it is luckier and create a longer chain.

5.1.2 Branching Random Walks. Similarly to Bagaria et al. [3], we use the theory of branching random walks to study the problem of grinding. The rest of the assumptions (e.g., about synchronicity) are as presented in Section 3.

We consider the following branching random walk that we note BRW. We first define it using standard branching process vocabulary, before explaining how it relates to grinding and blockchains. As before, time is divided into discrete time steps. An initial ancestor is located at the origin. At each time step a particle gives birth to a random number of children before dying. Each child is randomly scattered through $\mathbb{N}$. In the next time-step, each child will give birth to their own children before dying. The process repeats indefinitely. Each particle is independent of the others and verifies the following properties:

(1) Each particle has at least one child. The first child is located at the same position as its parent.
(2) v denotes the position of a particle. The other children of the particle, if they exist, are located at position $v + 1$.
(3) The number of children located at position $v + 1$ follows a distribution noted Z.

Such a process is illustrated in Figure 3. We will consider two different distributions for Z. In the first case, Z will follow a Bernoulli distribution of parameter α. There is exactly one particle at height $v+1$ with probability α (and 0 otherwise). In this case, each particle will have one child with probability $1 - \alpha$ and two children (one at position v and one at position $v + 1$) with probability α. We denote by BRW1 the associated branching random walk.

In the second case, Z follows a Poisson distribution of parameter α. There will be i particle in position $v + 1$ with probability $\frac{e^{-\alpha}\alpha^i}{i!}$. In this case, each particle has one child with probability $e^{-\alpha}$ and

a total of $i > 1$ children with probability $\frac{e^{-\alpha}\alpha^{i-1}}{(i-1)!}$. We denote by BRW2 this branching random walk.

Intuitively, a particle having exactly one child corresponds to the case where the adversary is not elected leader hence its chain does not increase by one and the child particle stays at the same position as its parent. This happens with probability $1-\alpha$ in the SSLE case and $e^{-\alpha}$ in the PLE case. A particle having more than one child corresponds to the case where the adversary was elected leader in that round, in which case its private chain increases by one and, analogously, the position of the other children increases by one.

In the case where the adversary has $m > 1$ of its miners elected leader, then all of them will create a new block with a new random value and thus a new potential chain. Each particle therefore corresponds to a new chain that will grow independently of the other from then on. The maximum position for BRW at time n corresponds to the longest chain that the adversary has been able to create by grinding. We are interested in comparing this value with the length of the honest chain.

Honest players do not form a coalition and are not grinding, hence their chain evolves as a random walk that increases by one with probability δ_+ and stays unchanged with probability $\delta_0 = 1-\delta_+$. In the SSLE case, we have $\delta_+^{SSLE} = 1-\alpha$ and in the PLE $\delta_+^{PLE} = 1-e^{\alpha-1}$. Let M_i denote the maximum position of the generic

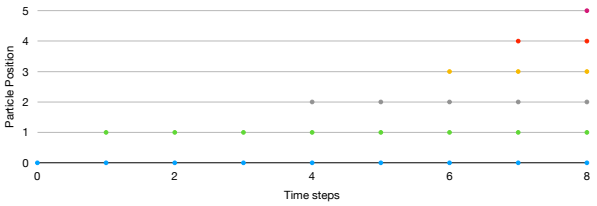


Figure 3: Example of a branching random walk. There may exist more than one particle at each position.

branching random walk BRW described above at time $i \in \mathbb{N}$. When Z is specified to be a Bernoulli distribution of parameter α we will write this process $M^{SSLE}(\alpha)$ and when Z is a Poisson distribution of parameter α, we will write the corresponding stochastic process $M^{PLE}(\alpha)$. Similarly, we denote by S_i the position of the generic honest random walk at time i and will specify $S_i^{SSLE}(\alpha)$ or $S_i^{PLE}(\alpha)$ when needed.

The SSLE and PLE grinding games are then defined as follows.

Definition 5.1 ((α, L)-SSLE Grinding Game). For $M^{SSLE}(\alpha)$ and $S^{SSLE}(\alpha)$ as defined above, we say that the adversary wins the SSLE grinding game of length L and power α if at timestep L, $M_L^{SSLE}(\alpha) \geq S_L^{SSLE}(\alpha)$.

Definition 5.2 ((α, L)-PLE Grinding Game). For $M^{PLE}(\alpha)$ and $S^{PLE}(\alpha)$ as defined above, we say that the adversary wins the PLE grinding game of length L and power α if at timestep L, $M_L^{PLE}(\alpha) \geq S_L^{PLE}(\alpha)$.

As before, we will sometimes refer to simply *the grinding game* when talking about the generic processes M and S.

We are interested in comparing the persistence parameter for SSLE compared to PLE and, hence, comparing the probabilities of winning the SSLE grinding game and the PLE grinding game.

Additionally, we are interested in the security threshold of the two grinding games. In the non-grinding case, we already know that the protocol is secure against private attacks if and only if $\alpha < 0.5$ for obvious reasons (i.e., the ϵ−persistence parameter exists and is finite for every $\epsilon \in (0,1)$ and $\alpha < 0.5$). With the grinding attack, an adversary may win an unfair advantage in the private attack and take over the honest chain of any length even with less than half of the stake. For the PLE case, Bagaria et al. [3] proved that the protocol is secure in the case of grinding if and only if $\alpha < 1/(1+e)$ for a network delay $\Delta = 0$. In the next section, we first derive the security threshold, i.e., the biggest value α_0 such that there exists a finite ϵ−persistence parameter for every $\epsilon \in (0,1)$. We then look at the probabilities of winning the game of length n for an adversary with power less than α_0 and again, compare the two cases.

Before moving on to the analysis, we make one important remark in the SSLE case with grinding. Since the randomness on the honest and adversarial chains are now different, the leader elections on each of these chains become independent. This is unlike the non-grinding case, where we had $h_n = 1 - a_n$. The SSLE thus does not act like an SSLE anymore. There could be more than one winner per round, each on a different chain, or no block created at all in some rounds (e.g., if in the adversarial chain the leader elected is honest and in the honest chain the leader elected is adversarial). The SSLE thus acts more like a probabilistic leader election where one leader is elected with probability α at each round on each adversarial chain, and one leader is elected with probability $1-\alpha$ on the honest chain. Unlike with the PLE private game, however, there can be at most one leader elected per round on each chain.

5.2 Analysis

5.2.1 Security Threshold. We start by defining the security threshold of the grinding game. Intuitively, the security threshold captures the threshold for which the protocol is secure after a certain length. In other words, the probability of winning the game can be made as small as desired by choosing a long enough length.

Definition 5.3 (Security threshold). If there exists $\alpha_0 \in (0,1)$ such that for every $\alpha > \alpha_0$, $M_i(\alpha) > S_i(\alpha)$ asymptotically a.s. and for $\alpha < \alpha_0$, $S_i(\alpha) > M_i(\alpha)$ asymptotically a.s., we call such α_0 the *security threshold* of the grinding game.

We note that the above definition does not say anything about the behaviour of the game with power exactly α_0. In order to compute the security threshold in both cases, we use results from Biggins [6, Section 6] that we adapt to our discrete time model. Specifically, Biggins considers a continuous model where particles are scattered through $\mathbb{R}$ and particles are of different types, whereas we only have one type of particle that is scattered through $\mathbb{N}$. Additionally, he considers the minimum position of the branching random walk rather than the maximum. With this mind, we can directly derive

the following result:

$$\lim_{n\to\infty} \frac{M_n}{n} \to -\gamma \text{ a.s. where } \gamma \text{ is defined as below:}$$
$$\phi(\theta) := \mathbb{E}[1 + e^{\theta} Z]$$
$$\mu(a) := \inf\{e^{\theta a}\phi(\theta) : \theta \geq 0\}$$
$$\gamma := \inf\{a : \mu(a) \geq 1\}$$

Similarly following the law of large numbers (applied to the Binomial distribution), we have $\lim_{n\to\infty} \frac{S_n}{n} \to \delta_+$ a.s. Hence, we conclude that $M_n > S_n$ a.s. asymptotically if $-\gamma > \delta_+$ and $M_n < S_n$ a.s. asymptotically if $-\gamma < \delta_+$. As a consequence, the security threshold corresponds to the case $-\gamma = \delta_+$. Based on this observation, we prove the following theorem.

Theorem 5.4. *The security threshold of the SSLE, resp. PLE, grinding games are:* $\alpha^{SSLE} \simeq 0.36$ *and* $\alpha^{PLE} \simeq 0.265$.

Proof. In order to prove the theorem, we compute the value of γ. We start by computing ϕ.

$$\begin{aligned}\phi &= \mathbb{E}[1 + e^{\theta} Z] \\ &= 1 + e^{\theta}\alpha\end{aligned}$$

We first note that this expression is independent of whether the number of blocks (or children at position 1) follows a Bernoulli or Poisson distribution since they both have the same expectation. This is in itself an interesting observation since it means that, asymptotically, both processes M_i^{SSLE} and M_i^{PLE} behave similarly (although this is not the case for S_i^{SSLE} and S_i^{PLE}).

We now move on to compute μ and γ, which will be equal for both branching random walks since they only depend on ϕ.

First, we compute $\mu(a) = \inf\{e^{\theta a}\phi(\theta) : \theta \geq 0\}$. We have $e^{\theta a}\phi(\theta) = e^{\theta a}(1 + e^{\theta}\alpha)$, hence, $\mu(a) = 1 + \alpha$ for $a \geq 0$, $\mu(a) = 0$ for $a \leq -1$, so it remains to compute $\mu(a)$ for $-1 < a < 0$. We have $\frac{\partial}{\partial\theta}(e^{\theta a}\phi(\theta)) = e^{\theta a}(a + \alpha(a+1)e^{\theta}) \geq 0 \Leftrightarrow \theta \geq \ln(-\frac{a}{\alpha(a+1)})$. Hence $\theta \mapsto e^{\theta a}\phi(\theta)$ is decreasing up until $\theta_0 = \ln(-\frac{a}{\alpha(a+1)})$ and increasing after this. We therefore have that for $-1 < a < 0$, $\mu(a) = e^{\theta_0 a}\phi(\theta_0) = (\frac{-a}{\alpha(a+1)})^a \frac{1}{a+1}$.

$$\mu(a) = \begin{cases} 1+\alpha, & \text{for } a \geq 0 \\ \left(\frac{-a}{\alpha(a+1)}\right)^a \frac{1}{a+1}, & \text{for } -1 < a < 0 \\ 0, & \text{for } a < -1 \end{cases}$$

Next we compute $\gamma = \inf\{a : \mu(a) \geq 1\}$. We denote $\zeta(a) = (\frac{-a}{\alpha(a+1)})^a \frac{1}{a+1}$ for $-1 < a < 0$. We have $\frac{\partial}{\partial a}\zeta = \frac{1}{a+1}(\frac{-a}{\alpha(a+1)})^a \log(\frac{-a}{\alpha(a+1)})$ and $\frac{\partial}{\partial a}\zeta \geq 0 \Leftrightarrow a \leq -\frac{\alpha}{\alpha+1}$. Hence ζ is first increasing then decreasing. Furthermore, we have that $\lim_{a\to 0}\zeta(a) = 1$ and $\lim_{a\to -1}\zeta(a) = \alpha$ hence ζ reaches one exactly once and γ is the unique solution to $\zeta(a) = 1$ for $-1 < a < 0$. We thus know that gamma solves the following equation:

$$\left(\frac{-\gamma}{\alpha(\gamma+1)}\right)^{\gamma} \frac{1}{\gamma+1} = 1$$

In the SSLE case, the threshold corresponds to the case where $-\gamma = 1 - \alpha^{SSLE}$ and, in the PLE case, the threshold corresponds to the case $-\gamma = 1 - e^{\alpha^{PLE}-1}$. Hence α^{SSLE} satisfies the following equation:

$$\left(\frac{1-\alpha^{SSLE}}{(\alpha^{SSLE})^2}\right)^{\alpha^{SSLE}-1} = \alpha^{SSLE}$$

The solution can be computed numerically giving $\alpha^{SSLE} \simeq 0.360$. α^{PLE} on the other hand, satisfies the following equation:

$$\left(\frac{1-e^{\alpha^{PLE}-1}}{\alpha e^{\alpha^{PLE}-1}}\right)^{e^{\alpha^{PLE}-1}-1} = e^{\alpha^{PLE}-1}$$

which can be solved numerically giving $\alpha^{PLE} \simeq 0.265$. □

The above theorem shows that SSLE significantly increases the security threshold in the grinding private attack. We also remark that the threshold found in the PLE case is similar to the one found by Bagaria et al. [3] ($1/(1+e)$), although there is a slight difference between the models. The reference considers a continuous-time model (i.e., the slot duration δ is very small) and a delay of propagation of zero. This means that, for example, if two honest leaders were to find a block at δ milliseconds of interval, where δ is very small, then these two blocks will be added to the honest chain even though in practice they were found at the same time. In our model, we consider a discrete time model and a synchronous network with a strictly positive delay, and so two blocks found in the same round cannot be added to the same chain.

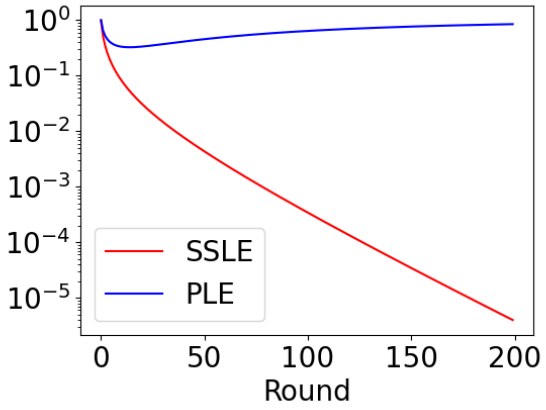
5.2.2 Persistence parameter. In the previous section, we computed the security threshold of the SSLE and PLE grinding games. We therefore know that, for an adversary below that power, we can find a length L such that the probability of winning the grinding game of parameter (α, L) is as small as desired. However, our analysis was only asymptotic and did not give any information about the persistence parameter or the behaviour of the game for a shorter time period. In this section, we study the probability of winning the SSLE and PLE grinding games of length n in order to give an estimate of the $\epsilon-$persistence parameter.

We are interested in the variable $(D_i = M_i - S_i)_{i\in\mathbb{N}}$ and especially the event $M_i - S_i \geq 0$ that corresponds to the adversarial chain being longer or equal than the honest chain and hence the adversary winning the grinding game.

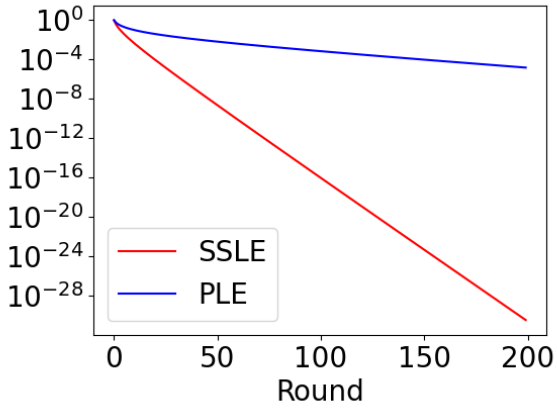
We denote $a_{i,j} = P(M_i < j)$. Since, by definition, $M_0 = 0$, we have $a_{0,j} = 1$ for $j > 0$. We now define the following recursive formula for $a_{i,j}$:

$$\begin{aligned}a_{i,j} &= \sum_{m=0}^{\infty} P(Z = m)\Pr[M_i < j | Z = m] \\ &= a_{i-1,j}\sum_{m=0}^{\infty} P(Z = m)(a_{i-1,j-1})^m\end{aligned}$$

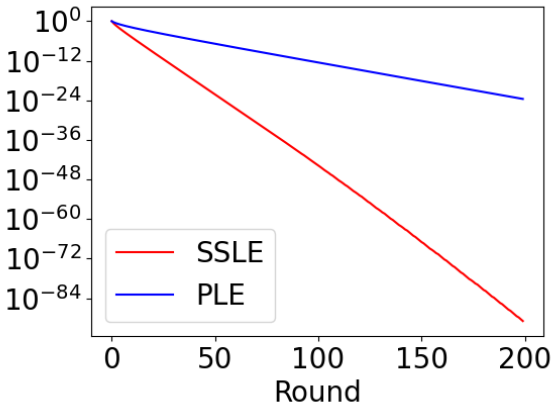
This equality is explained as follows. We start at time 0. We denote m the number of children at position 1 of the initial particle (i.e., the total number of children is $m+1$). The first child stays at position 0, whereas the other m children will increase position by one. All of the children generate independent processes similar to their ancestor, except starting at $(i+1, 0)$ for the first child, and $(i+1, 1)$ for the other m children. The process M will not reach j at step i if and only if none of the processes engendered by the



(a) $\alpha = 0.3$



(b) $\alpha = 0.2$



(c) $\alpha = 0.1$

Figure 4: Probability that the adversarial chain is greater or equal than the honest chain when grinding for a chain of length n

children of the original particle reach j. Since all the processes are independent, the probability that the process engendered by the first child never reaches j is $a_{i-1,j}$. For the rest of the m children, this probability is $a_{i-1,j-1}$. Conditional on the particle having $m+1$ children, the probability that M does not reaches j by time i is equal to $a_{i-1,j}(a_{i-1,j-1})^m$.

Adapting the above probabilities to the SSLE grinding game yields the following:

$$a_{i,j}^{SSLE} = a_{i-1,j}^{SSLE}(1 - \alpha + \alpha a_{i-1,j-1}^{SSLE})$$

Whereas in the PLE game, it becomes:

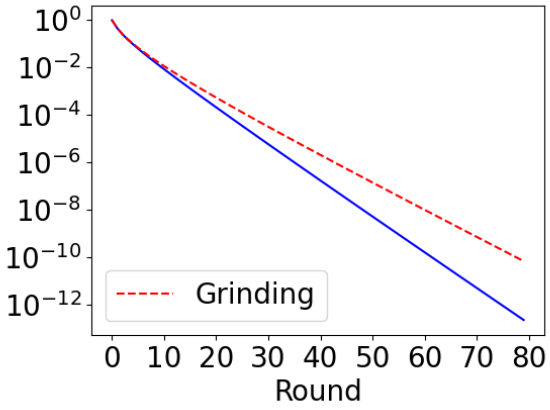
$$a_{i,j}^{PLE} = a_{i-1,j}^{PLE} e^{\alpha(a_{i-1,j-1}^{PLE} - 1)}$$

We note we have found a recursive formula for $a_{i,j}$ that depends on $(a_{i-1,j-1}, a_{i-1,j})$.

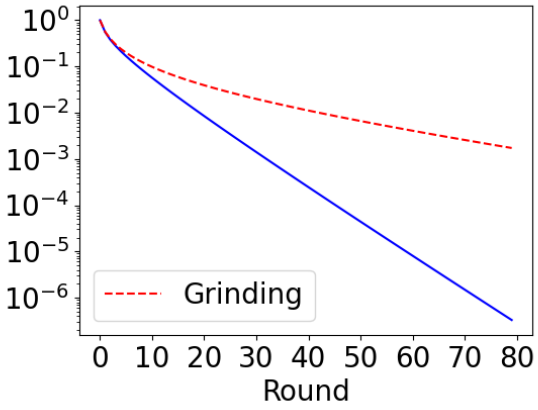
Next, we are interested in $b_i = \Pr[Mi - S_i \geq 0]$. We have:

$$\begin{aligned} b_i &= \Pr[Mi - S_i \geq 0] \\ &= \sum_{s=0}^{i} \Pr[S_i = s] \times \Pr[M_i \geq s] \\ &= \sum_{s=0}^{i} \Pr[S_i = s] \times (1 - a_{i,s}) \\ &= \sum_{s=0}^{i} \text{Bin}(\delta_+, i, s) \times (1 - a_{i,s}) \end{aligned}$$

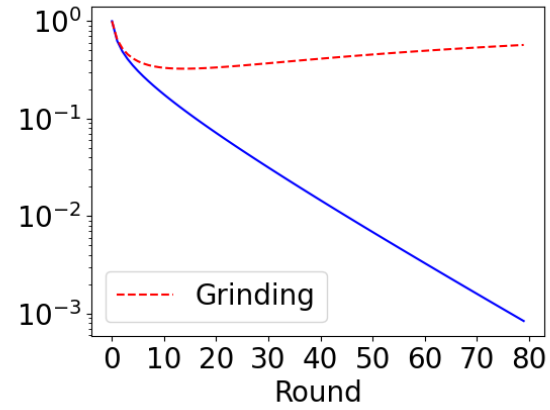
We plot this probability for different values of α and n in Figure 4. Here, we quickly remark that this probability does not allow us to find the exact ϵ–persistence parameter as it is not the probability that the adversary violates the persistence of the blockchain for any length greater than n but the probability that the adversary wins the game of length exactly n. In theory, the probability that



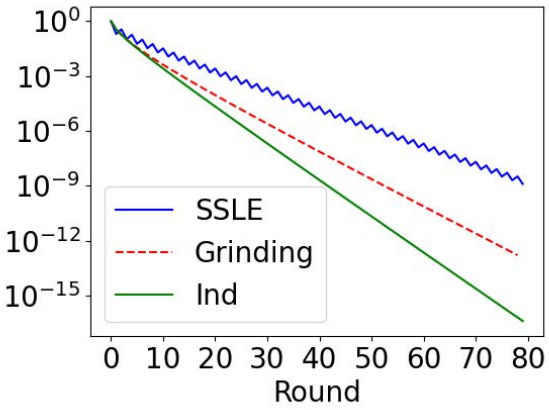
(a) PLE with and without grinding: $\alpha = 0.1$.



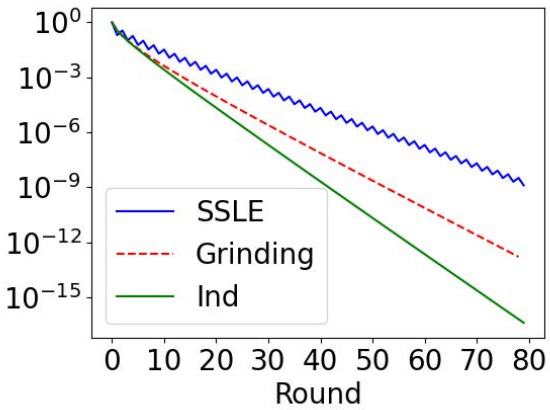
(b) PLE with and without grinding: $\alpha = 0.2$.



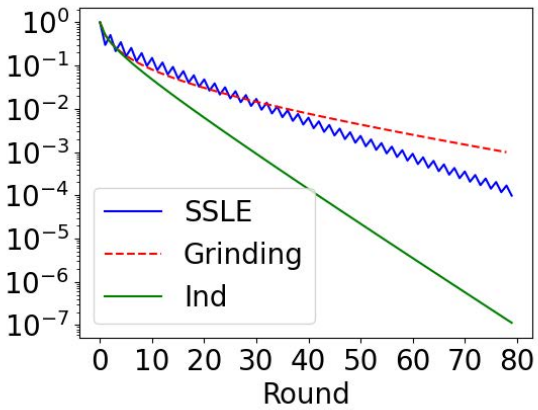
(c) PLE with and without grinding: $\alpha = 0.3$.



(d) SSLE: $\alpha = 0.1$.



(e) SSLE:$\alpha = 0.2$.



(f) SSLE: $\alpha = 0.3$.

Figure 5: Probability of winning the simple or grinding private games of length exactly n for different values of α (logarithmic scale). "Ind" corresponds to the Independent SSLE private game.

11

the adversary wins the game for any $n \geq n_0$ is slightly bigger than the probability we computed as there is always a small chance that an adversary that did not win at length n could catch up in the future. In the grinding case, this probability is much more complex to compute than in the previous section. However, the probability that the adversary wins the grinding game of length exactly n_0 still provides an interesting proxy measure for the security of the underlying protocol.

Results Interpretation. In Figure 4 we see that SSLE performs consistently better than PLE in the sense that the probability of winning the grinding game is smaller for SSLE than for PLE. We also notice, as before, that for n big enough, this probability can be approximated as e^{-an}. Using the same method as before, we can find that SSLE reduces the persistence parameter by roughly 70% in the case of a 10% adversary and 80% for a 20% adversary. In this case, the improvement is even more drastic than in the private game. Unlike the private game, however, the reduction is more noticeable for a 20% than for a 10% adversary.

It is also interesting to compare the probabilities of winning the private game vs winning the grinding game. We plot these probabilities in Figure 5. In the simple (i.e., non-grinding) case, we compute the probability that the gap is positive instead of using the probability in Section 4, as this matches the probability we have computed in the grinding case. Intuitively, grinding should increase the probability of winning the game of length n, which is what we observe for PLE in Figure 5a, 5b and 5c. However, in the SSLE case, we observe the opposite for a 10 and 20% adversary. As we have discussed before, in the case of grinding, the SSLE game does not act anymore as a single secret leader election since the adversarial and honest chains are now independent as they operate on different random beacons. The SSLE grinding game is thus more similar to the PLE private game than to the SSLE private game, except that the probabilities p_a and p_h should be adapted accordingly. We now define the following game:

Definition 5.5 ((L, α)-idependent SSLE Private Game). The independent SSLE private game with parameters (L, α) is defined as follows: at each round $n \in [1, \ldots, L]$ a number a_n of adversarial leaders is selected from a Bernoulli distribution of parameter α and a number h_n of honest leaders is selected from an independent Bernoulli distribution of parameter $1 - \alpha$. We say that the adversary wins the PLE private game of length L and power α if the number of rounds with non-zero adversarial leaders is greater or equal than the number of rounds with non-zero honest leaders, i.e.:

$$|\{n \in [1, \ldots, L] : a_n = 1\}| \geq |\{n \in [1, \ldots, L] : h_n = 1\}|.$$

This game is equivalent to a PLE private game, except that we now have $p_a = \Pr[a_n > 0] \times \Pr[h_n = 0] = \alpha^2$ and similarly $p_h = (1 - \alpha)^2$ and $p_0 = 2\alpha(1 - \alpha)$. We plot the probabilities of winning the independent SSLE private game and compare them to the SSLE grinding game in Figure 5d, 5e and 5f. We indeed notice that the independent SSLE game performs much better than the grinding game but, surprisingly, also better than the SSLE game. The difference between the independent SSLE and SSLE private games can be explained by the fact that the adversary in the independent game is much less likely to be elected sole leader (α^2 vs α) and, hence, the gap in this case increases less often. The difference between the independent SSLE and PLE is also explained similarly: the probability of the gap increase goes from $e^{1-\alpha} - e^{-1}$ to $(1 - \alpha)^2$. Although the persistence parameter is smaller in this case, it is also expected that there will be rounds with no winner as well as more natural forks (unlike in the SSLE case).

6 CONCLUSION AND FUTURE WORK

In this work, we have performed a comparison of private attacks against longest-chain PoS protocols that use SSLE and PLE. We have found that the persistence parameter under this specific attack is reduced significantly when using SSLE, by around 25% against a 25 or 33% adversary. We also found that the security threshold against grinding private attacks is higher ($\simeq 0.36$) in the SSLE case than in the PLE case ($\simeq 0.26$). These results are encouraging and should help convince real-world system designers to make the switch to SSLE in their PoS blockchains. For future work, it will be interesting to compare the results based on other attacks (e.g., balance attack [3] or general case), as well as to relax assumptions such as the synchronous network or static adversary.

ACKNOWLEDGMENTS

The authors would like to thank Jorge Soares for his valuable feedback on this paper. D.C. was supported by the MIUR grant 'Dipartimenti di Eccellenza 2018-2022' (E11G18000350001).

REFERENCES

[1] Lecture2: Random walks, reflection and reversal. http://cgm.cs.mcgill.ca/~breed/MATH671/lecture2corrected.pdf.
[2] Sarah Azouvi, Patrick McCorry, and Sarah Meiklejohn. Winning the caucus race: Continuous leader election via public randomness. *arXiv preprint arXiv:1801.07965*, 2018.
[3] Vivek Bagaria, Amir Dembo, Sreeram Kannan, Sewoong Oh, David Tse, Pramod Viswanath, Xuechao Wang, and Ofer Zeitouni. Proof-of-stake longest chain protocols: Security vs predictability. *arXiv preprint arXiv:1910.02218*, 2019.
[4] Amos Beimel, Eran Omri, and Ilan Orlov. Protocols for multiparty coin toss with dishonest majority. In *Annual Cryptology Conference*, pages 538–557. Springer, 2010.
[5] Michael Ben-Or and Nathan Linial. Collective coin flipping, robust voting schemes and minima of banzhaf values. In *26th Annual Symposium on Foundations of Computer Science (sfcs 1985)*, pages 408–416. IEEE, 1985.
[6] JD Biggins. The first-and last-birth problems for a multitype age-dependent branching process. *Advances in Applied Probability*, pages 446–459, 1976.
[7] Erica Blum, Aggelos Kiayias, Cristopher Moore, Saad Quader, and Alexander Russell. The combinatorics of the longest-chain rule: Linear consistency for proof-of-stake blockchains. In *Proceedings of the Fourteenth Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 1135–1154. SIAM, 2020.
[8] Dan Boneh, Saba Eskandarian, Lucjan Hanzlik, and Nicola Greco. Single secret leader election. In *Proceedings of the 2nd ACM Conference on Advances in Financial Technologies*, pages 12–24, 2020.
[9] Ignacio Cascudo and Bernardo David. Scrape: Scalable randomness attested by public entities. In *International Conference on Applied Cryptography and Network Security*, pages 537–556. Springer, 2017.
[10] Dario Catalano, Dario Fiore, and Emanuele Giunta. Efficient and universally composable single secret leader election from pairings.
[11] Phil Daian, Rafael Pass, and Elaine Shi. Snow white: Robustly reconfigurable consensus and applications to provably secure proof of stake. In *International Conference on Financial Cryptography and Data Security*, pages 23–41. Springer, 2019.
[12] Bernardo David, Peter Gaži, Aggelos Kiayias, and Alexander Russell. Ouroboros praos: An adaptively-secure, semi-synchronous proof-of-stake blockchain. In *Annual International Conference on the Theory and Applications of Cryptographic Techniques*, pages 66–98. Springer, 2018.
[13] Soubhik Deb, Sreeram Kannan, and David Tse. Posat: Proof-of-work availability andunpredictability, without the work. *arXiv preprint arXiv:2010.08154*, 2020.
[14] Amir Dembo, Sreeram Kannan, Ertem Nusret Tas, David Tse, Pramod Viswanath, Xuechao Wang, and Ofer Zeitouni. Everything is a race and nakamoto always

wins. In *Proceedings of the 2020 ACM SIGSAC Conference on Computer and Communications Security*, pages 859–878, 2020.

[15] Peter Gaži, Aggelos Kiayias, and Alexander Russell. Tight consistency bounds for bitcoin. Cryptology ePrint Archive, Report 2020/661, 2020. https://eprint.iacr.org/2020/661.

[16] Yossi Gilad, Rotem Hemo, Silvio Micali, Georgios Vlachos, and Nickolai Zeldovich. Algorand: Scaling byzantine agreements for cryptocurrencies. In *Proceedings of the 26th Symposium on Operating Systems Principles*, pages 51–68, 2017.

[17] Aggelos Kiayias, Saad Quader, and Alexander Russell. Consistency of proof-of-stake blockchains with concurrent honest slot leaders. *arXiv preprint arXiv:2001.06403*, 2020.

[18] Aggelos Kiayias, Alexander Russell, Bernardo David, and Roman Oliynykov. Ouroboros: A provably secure proof-of-stake blockchain protocol. In *Annual International Cryptology Conference*, pages 357–388. Springer, 2017.

[19] Jing Li, Dongning Guo, and Ling Ren. Close latency-security trade-off for the nakamoto consensus. *CoRR*, abs/2011.14051, 2020.

[20] Silvio Micali, Michael Rabin, and Salil Vadhan. Verifiable random functions. In *40th annual symposium on foundations of computer science (cat. No. 99CB37039)*, pages 120–130. IEEE, 1999.

[21] Ewa Syta, Philipp Jovanovic, Eleftherios Kokoris Kogias, Nicolas Gailly, Linus Gasser, Ismail Khoffi, Michael J Fischer, and Bryan Ford. Scalable bias-resistant distributed randomness. In *2017 IEEE Symposium on Security and Privacy (SP)*, pages 444–460. Ieee, 2017.