Sesame Challenges: Decentralized Prizes to Incentivize Research in Computational Mathematics

Maxwell D. Radin and Liane Nakamura
Sesame Foundation

September 17, 2022
v0.1.0

Abstract

Monetary prizes have long been used to incentivize research into unsolved problems in mathematics. While most prizes have been created by a single institution or individual, it can be difficult for a decentralized community to offer such prizes. In particular, it is challenging to provide assurances to donors that their contribution will be properly distributed to the individual or group that solves a given problem. This paper introduces the Sesame protocol, a decentralized protocol allowing the creation of prizes for unsolved mathematical problems in cases where verification of a correct solution is computationally efficient. This system provides guarantees that the prize can only be withdrawn to someone who solves a given problem. The Sesame protocol also provides a public interface to allow additional contracts and applications to be built on top of it. The Decentralized Factoring Challenge, inspired by the RSA Factoring Challenge, is implemented as an example.

1 Introduction

Many people and organizations have used prizes to draw attention to unsolved mathematical problems of interest. For example, the Clay Institute has offered $1 million for solutions to the so-called Millennium Prize Problems [1, 2]. Other examples include the Electronic Frontier Foundation (EFF) Cooperative Computing Awards for identifying large primes [3] and the Beal Prize related to the solutions of the equation $A^x + B^y = C^z$ [4].

Despite the success of such prizes in drawing attention to important unsolved problems, the creation of such prizes involves many complexities. These complexities arise from the difficulty in providing a strong guarantee, to those pursuing the problem as well as those contributing to the prize, that the prize would in fact be awarded to the first party to solve the problem and not to anyone else. Such guarantees can be especially difficult given the long time-scales for solving such problems.

In order to address these challenges, past prizes have often relied on trusted organizations as facilitators. For example, the Beal Prize is sponsored by billionaire mathematician D. Andrew Beal but administered by the American Mathematical Society (AMS). Such approaches have limitations however: one must persuade a trusted organization to take interest in a given problem and negotiate the financial and legal details of such an arrangement. Such challenges are particularly significant for problems where the interest is not concentrated in a single individual or entity but diffused across many stakeholders.

Blockchains provide a new platform for incentivizing and facilitating research as part of the decentralized science (DeSci) movement [5]. Christie introduced the notion of using smart contracts as a mechanism for rewarding combinatorial optimization problems [6]. This included an implementation of a proof-of-concept contract and user interface for the trivial problem of maximizing the number of ones in a bitstring (also referred to as the OneMax problem). This work mainly focused however on combinatorial optimization, and did not explore failure modes arising from the possibility of miners/validators front-running submitted solutions. Recently, the Proof of Quantum challenge launched a set of challenges related to breaking the Secp256k1 elliptic curve protocol used by Ethereum itself [7]. The mechanism introduced by this contest allows for the creation of challenges that involve the cryptographic protocols used by the underlying blockchain.

As a step towards exploring how blockchains could incentivize mathematics research more broadly, this paper introduces the Sesame protocol, a framework for creating decentralized prizes for unsolved mathematical problems. The Sesame protocol is an Ethereum contract specification that empowers individuals to create prizes that others can donate to and build additional contracts on top of. It uses a sealed claim system to prevent front-running of solutions and therefore provide assurance that the prize will be disbursed appropriately. As a concrete implementation, this work also introduces the Decentralized Factoring Challenge (DFC). Inspired by the RSA Factoring Challenge, the DFC provides rewards for factoring large integers, a problem of interest because of the role of integer factorization in public-key cryptography.

2 Integer factorization

Integer factorization is a problem of not only fundamental mathematical interest, but also great practical interest due to its central role in public-key cryptography. Many current telecommunication systems, including much of the internet, rely on public-key encryption whose security comes from the fact that large semiprimes are difficult to factor. In order to verify the difficulty of this problem, RSA Laboratories launched a factoring challenge in 1991, offering prizes totalling over $500,000 USD for the factors of large semiprimes [8]. Although the contest was terminated in 2007, researchers continue to pursue these challenges. For example, Boudot et al. factored RSA-250 in 2020 [9], the largest RSA number to be factored so far.

Although the RSA Factoring Challenge successfully demonstrated the security of public-key cryptography against current technology, the emergence of quantum computation has raised the possibility that public-key cryptography based on factorization will be broken in the coming decades. In 1994, Peter Shor introduced a quantum algorithm that is almost exponentially faster than the best known classical algorithms for integer factorization [10]. Although no quantum computers existed at that time, recent years have seen significant breakthroughs in the realization of quantum computers. For example, in 2019, Google announced that their quantum processor had achieved quantum supremacy, a major milestone in the realization of quantum computers [11].

These breakthroughs have potentially significant implications for privacy and telecommunications infrastructure. For example, the RAND Corporation estimated that quantum computers relevant for cryptography would become available by approximately 2033 [12]. Similarly, the Metaculus prediction community has forecast a significant chance that quantum computers could outperform classical factoring algorithms in the coming decades [13]. For example, the Metaculus community has forecast a 20% probability that a quantum computer will factor a previously unfactored RSA number by 2034. Note that related quantum algorithms also could have significant implications for cryptographic tools used today in many blockchains [14].

Given the expectation that quantum computation will likely break important protocols such as public-key cryptography and have major implications for communication security, there is significant value in exploring the limits of emerging methods for integer factorization. However, because there are essentially no commercial applications for integer factorization, there is limited incentive for white-hat researchers to invest significant resources in this area.

3 Challenge mechanics

A Sesame Challenge represents a smart contract that allows users to contribute to a prize for a given mathematical problem. Importantly, the mechanics of the contract allow donors to have confidence that the first person to solve the challenge will be able to withdraw the prize. A key part of providing this assurance is the use of a sealed claim mechanism in order to prevent front running, as described below.

There are three ways to interact with a Sesame Challenge contract: donating to the prize, submitting a sealed claim, and withdrawing the prize.

Users can donate ETH to the prize using the `donate` method. To avoid unintentional donations to an already solved problem, the contract should reject donations if the prize has already been withdrawn (see below). Additionally, in order to allow applications to acknowledge donors, each donation will result in a event containing the address of the donor and size of donation being emitted.

A user who claims to have found the solution to the challenge may use the contract's `submitClaim` method to submit a sealed claim, which should be the Keccak-256 hash of the solution as well as a salt value chosen by the claimant. The purpose of submitting a sealed claim is to prevent front running, *i.e.* a malicious miner using the claimant's solution to withdraw the prize to their own address. The role of the salt is to allow the claimant an added level of security against front-running in the event that the hash of the solution would provide enough additional information to allow a front-runner to obtain the solution.

Lastly, a user who has previously submitted a valid sealed claim may withdraw the prize by passing the factors to the `withdraw` method along with the salt. This `withdraw` method performs several checks:

1. The challenge must not have already have been solved.
2. The address attempting the withdrawal must have previously submitted a sealed claim that matches the Keccak-256 hash of the submitted solution and salt, and the number of blocks elapsed since the submission of the sealed claim must be greater than or equal to the withdrawal delay.
3. The submitted solution must be a valid solution to the challenge.

In addition to the above methods, Sesame Challenges contract should provide a public variable `winner` of type `address` that stores the address of the winner if the challenge has been solved, and the zero address otherwise. It should also provide a public variable `withdrawlDelay` of type `uint256` indicating the number of blocks that must elapse after the submission of a sealed claim before the prize can be withdrawn. For some challenges, such as the factoring challenge below, additional public variables may be provided to specify details of the challenge.

4 The Decentralized Factoring Challenge

The DFC encourages research into integer factorization by providing a smart contract offering a prize for the factorization of large integers.

It consists of multiple Sesame Challenge contracts, each one representing a prize for the factorization of a specific integer. In addition to the methods and public variables described above, each DCF contract also exposes a public `BigNumber` representing the integer to be factored [15]. The `withdraw` method takes two `bytes` objects representing the claimed factors, in addition to the salt. To check if submitted factors are valid, three properties are checked:

1. The two factors must multiply to the target integer.
2. Neither of the factors is one.
3. The length of the byte array is a multiple of 32.

4. The byte array representing each factor does not have excess padding (i.e., more than 31 leading zeros).

The first two criteria require the claim to contain non-trivial factors of the target integer, while the second two require the byte arrays representing the claimed factors to conform to the format required by the BigNumber library.

This `withdraw` method performs several checks. First, the submitted factors must multiply to the target product. Second, the hash of the submitted factors must match a previously submitted sealed claim from the same Ethereum address. Third, the number of blocks mined since the sealed claim was submitted must equal or exceed the withdrawal delay. (This withdrawal delay, along with the product to be factored, are the two parameters defining a DFC contract.) The second and third checks allow for claimants to prevent front running.

The source code for the DCF contract can be found online [16]. Additionally, a web interface allowing users to donate, submit claims, and withdraw the prize can also be found online [17, 18].

5 Results

As a test of this framework, a factoring challenge with an 80-digit product (small enough to be readily factored in minutes on a laptop) was deployed [19]. The challenge was successfully solved and the prize withdrawn by an anonymous individual within 10 days [20]. A second factoring challenge for RSA-896, the smallest unfactored number for which RSA Laboratories had offered a monetary prize, has also been deployed [21].

6 Conclusions

This work introduced the Sesame protocol for using Ethereum contracts to create decentralized prizes for problems in computational mathematics. As a proof-of-concept, the Decentralized Factoring Challenge, inspired by the RSA Factoring Challenge, was deployed. A trivial factoring challenge was solved by a member of the public, validating the implementation and concept. A challenge for one unsolved RSA numbers has also been deployed.

One direction for further exploration is the application of the Sesame protocol to other types of problems besides factoring. Another direction is contracts that build on top of the Sesame public API. For example, as a non-monetary reward, an artist could create a non-fungible token (NFT) that could be minted only by the solver of a challenge. Such additions could help empower a broad audience to create, contribute to, and receive prizes in computational mathematics, and ultimately accelerate the pace of discovery.

7 Acknowledgements

The authors would like to thank B. Bleakley and D. Ko for valuable discussion and contributions to Sesame Foundation graphic design.

References

[1] *Millennium Prize Problems*. (accessed: 06.04.2022).

[2] James A Carlson, Arthur Jaffe, and Andrew Wiles. *The millennium prize problems*. Clay Mathematics Institute and American Mathematical Society, 2006. ISBN: 0-8218-3679-X.

[3] *Electronic Frontier Foundation (EFF) Cooperative Computing Awards*. (accessed: 06.04.2022).

[4] *Beal Prize*. (accessed: 06.04.2022).

[5] Sarah Hamburg. "Call to join the decentralized science movement". en. In: *Nature* 600.7888 (Dec. 2021), p. 221. DOI: 10.1038/d41586-021-03642-9.

[6] Lee A. Christie. "Decentralized Combinatorial Optimization". In: *Parallel Problem Solving from Nature – PPSN XVI*. Ed. by Thomas Bäck et al. Cham: Springer International Publishing, 2020, pp. 360–372. ISBN: 978-3-030-58112-1.

[7] *Proof of Quantum*. Polygon contract 0x34A86B3B9523d2d19Bbf199329983c802B3D4760. (accessed: 08.20.2022).

[8] *RSA Factoring Challenge*. (accessed: 06.04.2022).

[9] Ioana Patringenaru. *New record set for cryptographic challenge*. 2020. (accessed: 06.04.2022).

[10] P.W. Shor. "Algorithms for quantum computation: discrete logarithms and factoring". In: *Proceedings 35th Annual Symposium on Foundations of Computer Science*. 1994, pp. 124–134. DOI: 10.1109/SFCS.1994.365700.

[11] Frank Arute et al. "Quantum supremacy using a programmable superconducting processor". en. In: *Nature* 574.7779 (Oct. 2019), pp. 505–510. DOI: 10.1038/s41586-019-1666-5.

[12] *Securing Communications in the Quantum Computing Age: Managing the Risks to Encryption*. 2020. (accessed: 06.04.2022).

[13] *Date Quantum Algorithm Factors RSA Number*. (accessed: 06.04.2022).

[14] Amira Bouguera. *How Will Quantum Computing Affect Blockchain?* 2019. (accessed: 06.18.2022).

[15] *Big Number Library for Solidity*. (accessed: 08.20.2022).

[16] github.com/sesame-foundation/decentralized-factoring-challenge. (accessed: 06.04.2022).

[17] github.com/sesame-foundation/sesame-foundation-frontend. (accessed: 06.04.2022).

[18] *Decentralized Factoring Challenge*. (accessed: 08.20.2022).

[19] Ethereum contract 0x17Ba3367362B2A5390da1fc81BC8f9C01c28994E.

[20] Ethereum transaction 0xd0c0585560c759e9739f5b5df574e2f03c3bc2bac65bd490846775fdd139167c.

[21] Ethereum contract 0x99D376f94cF6e7541c8001677FBBd3A0cbbf8c88.