Sesame Challenges: Decentralized Prizes to Incentivize Research in Computational Mathematics

June 2022

1 Introduction

Many people and organizations have used monetary 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]. Other examples include the Electronic Frontier Foundation (EFF) Cooperative Computing Awards for identifying large primes [2] and the Beal Prize related to the solutions of the equation $A^x + B^y = C^z$ [3].

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 donating 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.

Christie introduced the notion of using smart contracts as a mechanism for rewarding combinatorial optimization problems [4]. 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 also did not explore failure modes arising from the possibility of miners/validators frontrunning submitted solutions.

This page introduces the Decentralized Factoring Challenge (DFC), a next step towards providing a mechanism for creating mathematical prizes. The DFC provides a mechanism for anyone to contribute to prizes for factoring large integers and, by leveraging Ethereum smart contracts, provides a guarantee that the first party to find the factors will be able to claim the prize. This page reviews the significance and future expectations for integer factorization, as well as the mechanics of the DFC.

2 Integer factorization

Zoom out to talk about broader threat of QC to cryptographic protocols cite NISTIR8413

Integer factorization is a problem not only of fundamental mathematical interest, but also great practical interest because of 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 [5]. Although the contest was terminated in 2007, researchers have continued to pursue the challenges. For example, Boudot et al. factored RSA-250 in 2020 [6], 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 coming decades. In 1994, Peter Shor introduced a quantum algorithm that is almost exponentially faster than the best known classical algorithms for integer factorization. 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.

As a result of these breakthroughs, many experts expect quantum computing to disrupt public-key cryptography in coming decades, with 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 [7]. Similarly, the Metaculus prediction community has forecast a significant chance that quantum computers could outperform classical factoring algorithms in the coming decades [8]. 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 [9].

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 represent a smart contract that allows users to contribute to a prize for a given mathematical problem. Importantly, the mechanics of contract allow donors to have confidence that the first person to factor the integer 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 frontrunning, 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 simply by sending ETH to the contract's address. 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 factors to the numbers may use the contract's `submitClaim` method to submit a sealed claim, which should be the Keccak-256 hash of the purported factors as well as a salt value chosen by the claimant. The purpose of submitting a sealed claim is to prevent frontrunning, as described below. The role of the salt is to allow the claimant an added level of security against frontrunning in the event that the hash of the solution would provide enough additional information to allow a frontrunner 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 withdrawl 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 withdrawl 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. 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 withdrawl delay. (This withdrawl 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, i.e. a malicious miner using the claimant's factors to withdraw the prize to their own address.

add link to github repo

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 (contract `0x17Ba3367362B2A5390da1fc81BC8f9C01c28994E`). The challenge was successfully solved and the prize withdrawn by an anonymous individual within 10 days.

References

[1] *Millennium Prize Problems*. URL: http://www.claymath.org/millennium-problems/millennium-prize-problems. (accessed: 06.04.2022).

[2] *Electronic Frontier Foundation (EFF) Cooperative Computing Awards*. URL: https://www.eff.org/awards/coop. (accessed: 06.04.2022).

[3] *Beal Prize*. URL: https://www.ams.org/prizes-awards/paview.cgi?parent_id=41. (accessed: 06.04.2022).

[4] 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.

[5] *RSA Factoring Challenge*. URL: https://web.archive.org/web/20130921043459/http://www.emc.com/emc-plus/rsa-labs/historical/the-rsa-factoring-challenge.htm. (accessed: 06.04.2022).

[6] Ioana Patringenaru. *New record set for cryptographic challenge*. 2020. URL: https://phys.org/news/2020-03-cryptographic.html. (accessed: 06.04.2022).

[7] *Securing Communications in the Quantum Computing Age: Managing the Risks to Encryption*. 2020. URL: https://www.rand.org/pubs/research_reports/RR3102.html. (accessed: 06.04.2022).

[8] *Beal Prize*. URL: https://www.metaculus.com/questions/3684/when-will-a-quantum-computer-running-shors-algorithm-or-a-similar-one-be-used-to-factor-one-of-the-rsa-numbers-for-the-first-time/. (accessed: 06.04.2022).

[9] Amira Bouguera. *How Will Quantum Computing Affect Blockchain?* 2019. URL: https://consensys.net/blog/developers/how-will-quantum-supremacy-affect-blockchain. (accessed: 06.18.2022).