ISSN: 2278-4632
Vol-12 Issue-01 No.01: 2022

PAYABLE OUTSOURCED DECRYPTION SCHEMES WITH PUBLIC-KEY ENCRYPTION (PKE)

CHENUMULLA JYOTHI Student[CSE], SRI VANI EDUCATIONAL SOCIETY GROUP OF INSTITUTIONS, A.P., India.

VADAPALLI GOPI Associate Professor& HEAD OF THE DEPARTMENT, Dept of CSE, SRI VANI EDUCATIONAL SOCIETY GROUP OF INSTITUTIONS, A.P., India.

ABSTRACT:

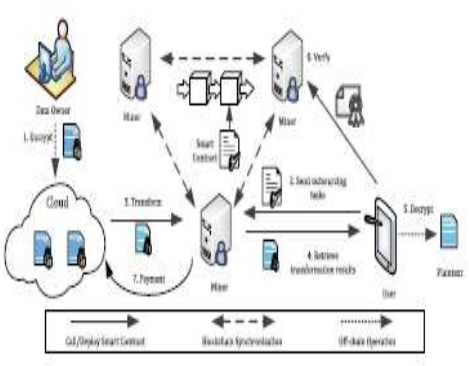
The concept of functional encryption (FE) has been introduced to address the shortcomings of public-key encryption (PKE) in many emerging applications which require both data storage and data sharing (e.g., cloud storage service). In this paper, we aim to design FE with payable outsourced decryption (FEPOD) schemes. The payment in an FEPOD scheme is achieved through a block chain-based crypto currency, which enables the user to pay a third party when it correctly completes the outsourced decryption. We define the adversarial model for FEPOD schemes, and then present a generic construction of FEPOD schemes. Also, we evaluate the performance of the proposed generic construction by implementing a concrete FEPOD scheme over a block chain platform.

INTRODUCTION:

One of the major issues existing in most FE schemes is the efficiency, as they are built from bilinear pairings of which the computation is very expensive. A widely accepted solution to this problem is outsourcing the heavy workloads to a powerful third party and leaving the user with the light computation. Nevertheless, it is impractical to assume that the third party (e.g., the cloud) will provide free services. To our knowledge, no attention has been paid to the payment procedure between the user and the third party in an FE with outsourced decryption (FEOD) scheme under the assumption that neither of them should be trusted. Suppose that Alice, a privilege user of a cloud storage application, is using a device with the constrained resource. Alice intends to access the encrypted data stored on the cloud, but she is unable to perform the heavy computation (e.g., the pairing operation) of decryption. A straightforward solution to this problem is an FE with outsourced decryption (FEOD) scheme such as an identity-based encryption

ISSN: 2278-4632
Vol-12 Issue-01 No.01: 2022

(IBE) with out-sourced decryption (e.g., [28]) or an attribute-based encryption(ABE) with outsourced decryption (e.g., [16]) scheme, which supports the user (i.e., Alice) to outsource the majority of the computation workloads in decryption to a powerful third party without leaking any sensitive information about the original data. Considering that in practice, the third parties would not like to provide a free service, and they expect to be paid for what they have done for others. Alice may send the computation task to a nearby device possessed by Bob, which is capable of conducting the computation in the network, and promise that "I will pay $1 to Bob once he provides me the correct result to this computation task". Bob receives the message, executes the computation, and sends Alice the result. To this end, two issues remain to be addressed: firstly, a mechanism that enables Alice to verify the correctness of Bob's answer before she pays Bob; secondly, a mechanism to address the worry of Bob that Alice may deny the correctness of his answer to escape the payment.



LITERATURE SURVEY

From secrecy to soundness:Efficient verification via secure computation

We study the problem of verifiable computation (VC) in which a computationally weak client wishes to delegate the computation of a function f on an input x to a computationally strong but untrusted server. We present new general approaches for constructing VC protocols, as well as solving the related problems of program checking and self-correcting. The new approaches reduce the task of verifiable computation to suitable variants of secure multiparty computation (MPC) protocols. In particular, we show how to efficiently convert the secrecy property of MPC protocols into soundness of a VC protocol via the use of a message authentication code (MAC). The new connections allow us to apply results from the area of MPC towards simplifying, unifying, and improving over previous results on VC and related

ISSN: 2278-4632
Vol-12 Issue-01 No.01: 2022

problems. In particular, we obtain the following concrete applications: (1) The first VC protocols for arithmetic computations which only make a black-box use of the underlying field or ring; (2) a non-interactive VC protocol for boolean circuits in the pre processing model, conceptually simplifying and improving the online complexity of a recent protocol of Gennaro et al. (Cryptology ePrint Archive: Report 2009/547); (3) NC0 self correctors for complete languages in the complexity class NC1 and various log-space classes, strengthening previous AC0 correctors of Goldwasser et al. (STOC 2008).

Efficient zero-knowledge contingent payments in cryptocurrencies without scripts

One of the most promising innovations offered by the cryptographic currencies (like Bitcoin) are the so-called smart contracts, which can be viewed as financial agreements between mutually distrusting participants. Their execution is enforced by the mechanics of the currency, and typically has monetary consequences for the parties. The rules of these contracts are written in the form of so-called "scripts", which are pieces of code in some "scripting language". Although smart contracts are believed to have a huge potential, for the moment they are not widely used in practice. In particular, most of Bitcoin miners allow only to post standard transactions (i.e.: those without the non-trivial scripts) on the blockchain. As a result, it is currently very hard to create non-trivial smart contracts in Bitcoin. Motivated by this, we address the following question: "is it possible to create non-trivial efficient smart contracts using the standard transactions only?" We answer this question affirmatively, by constructing efficient Zero-Knowledge Contingent Payment protocol for a large class of NP-relations. This includes the relations for which efficient sigma protocols exist. In particular, our protocol can be used to sell a factorization (p, q) of an RSA modulus n=pqn=pq, which is an example that we implemented and tested its efficiency in practice. As another example of the "smart contract without scripts" we show how our techniques can be used to implement the contract called "trading across chains".

How to use bitcoin to design fairprotocols

We study a model of fairness in secure computation in which an adversarial party that aborts on receiving output is forced to pay a mutually predefined monetary penalty. We then show how the Bitcoin network can be used to achieve the above notion of

ISSN: 2278-4632
Vol-12 Issue-01 No.01: 2022

fairness in the two-party as well as the multiparty setting (with a dishonest majority). In particular, we propose new ideal functionalities and protocols for fair secure computation and fair lottery in this model. One of our main contributions is the definition of an ideal primitive, which we call F ? CR (CR stands for "claim-or-refund"), that formalizes and abstracts the exact properties we require from the Bitcoin network to achieve our goals. Naturally, this abstraction allows us to design fair protocols in a hybrid model in which parties have access to the F ? CR functionality, and is otherwise independent of the Bitcoin ecosystem. We also show an efficient realization of F ? CR that requires only two Bitcoin transactions to be made on the network. Our constructions also enjoy high efficiency. In a multiparty setting, our protocols only require a constant number of calls to F ? CR per party on top of a standard multiparty secure computation protocol. Our fair multiparty lottery protocol improves over previous solutions which required a quadratic number of Bitcoin transactions.

PROPOSED APPROACH:

The idea of utilitarian encryption (FE) has been acquainted with address the weaknesses of public-key encryption (PKE) in many arising applications which require both information stockpiling and information sharing (e.g., distributed storage administration). One of the significant issues existing in most FE plans is the effectiveness, as they are worked from bilinear pairings of which the calculation is over the top expensive.

DISADVANTAGES OF EXISTING SYSTEM

- The computation is very expensive

PROPOSED SYSTEM:

We propose a notion of functional encryption with payable outsourced decryption (FEPOD), which allows anybody to check the correctness of the answer for the outsourcing computation task provided by an untrusted third party such that the payment can be processed by a block chain-based crypto currency.

• We delineate the security model of an FEPOD scheme, present a generic construction of it, and analyse its security.

• We implement a concrete FEPOD scheme, which is derived from the generic construction, over a blockchain platform to evaluate its feasibility and practicality.

ISSN: 2278-4632
Vol-12 Issue-01 No.01: 2022

ADVANTAGES OF PROPOSED SYSTEM

- Outsourcing Computation.
- Increasing efficience

MODULES DESCRIPTION

Data Owner:

In this application the owner is one of the main module for uploading the files and view the uploads file which are uploaded by the owner before do all these operations the owner should register with the application and the owner should authorized by the cloud.

Data User:

In this application the user also a modules to perform the bloom filter operation to access the files from the cloud, before do the search operations the user should get the search permission from the cloud then only the user can search the files after get the details of the searched file, if the user want to download the user should get the trapdoor key from the miner, then the user can able to download the file.

Cloud Server:

The cloud is the main module to operate this project in the users activations, owner activation and also the cloud can check the following operations like search permission provides to the users, can check the top-k searched keyword, top-k similarity in chart, top-k searched keyword in chart. Primarily the cloud should login. Then only the cloud can perform the above mentioned actions.

Miner:

In this Application miner is one the important module. Miner can able to login into application and view the request from the data owner and he can able to forward into cloud server or data owner. The that task will complete by the data owner.

SAMPLE RESULTS

ISSN: 2278-4632
Vol-12 Issue-01 No.01: 2022

CONCLUSION:

In this paper, in this paper, we presented a generic FE with payableoutsourced decryption (FEPOD) scheme which is publiclyverifiable to enable the third party to be paid for the service it provides via a blockchain-based cryptocurrency. Afterproving the security of the given generic FEPOD construction, we described the process of integrating FEPOD into ablockchain, and implemented an instantiation of FEPOD overa blockchain to evaluateits efficiency in practice.

REFERENCES:

[1]A Python Interface for Interacting With the Ethereum Blockchainand Ecosystem. Accessed: Jul. 10, 2019. [Online]. Available:https://github.com/ethereum/web3.py

[2] J.A.Akinyeleet al., "Charm: A framework for rapidly prototypingcryptosystems,"J. Cryptograph. Eng., vol. 3, no. 2, pp. 111–128,Jun. 2013.

[3] B. Applebaum, Y. Ishai, and E. Kushilevitz, "From secrecy to soundness:Efficient verification via secure computation," inProc. 37th Int. Colloq.Automat., Lang., Program., in Lecture Notes in Computer Science,vol. 6198. Bordeaux, France: Springer, Jul. 2010, pp. 152–163.

[4] W. Banasik, S. Dziembowski, and D. Malinowski, "Efficient zero-knowledge contingent payments in cryptocurrencies without scripts,"inProc. 21st Eur. Symp. Res. Comput. Secur., in Lecture Notes inComputer Science, vol. 9879. Heraklion, Greece: Springer, Sep. 2016,pp. 261–280, doi:10.1007/978-3-319-45741-3_14.

[5] I. Bentov and R. Kumaresan, "How to use bitcoin to design fairprotocols," inProc. 34th Annu. Cryptol. Conf., in Lecture Notes inComputer Science, vol. 8617. Santa Barbara, CA, USA: Springer,Aug. 2014, pp. 421–439.

[6] D. Boneh and M. Franklin, "Identity-based encryption from the Weilpairing,"SIAM J. Comput., vol. 32, no. 3, pp. 586–615, Jan. 2003.

[7] D. Boneh, A. Sahai, and B. Waters,"Functional encryption: Definitionsand challenges," inProc. 8th Theory Cryptogr. Conf. (TCC), in

ISSN: 2278-4632
Vol-12 Issue-01 No.01: 2022

LectureNotes in Computer Science, vol. 6597. Providence, RI, USA: Springer,Mar. 2011, pp. 253–273.

[8] V. Buterin.Ethereum White Paper: A Next Generation Smart Contract &Decentralized Application Platform. Accessed: Jul. 10, 2019. [Online].Available: https://github.com/ethereum/wiki/wiki/White-Paper

[9] J. Camenisch and V. Shoup, "Practical verifiable encryption and decryp-tion of discrete logarithms," inProc. 23rd Annu. Int. Cryptol. Conf.,inLecture Notes in Computer Science, vol. 2729. Santa Barbara, CA, USA:Springer, Aug. 2003, pp. 126–144

[10] M. Campanelli, R. Gennaro, S. Goldfeder, and L. Nizzardo, "Zero-knowledge contingent payments revisited: Attacks and payments forservices," inProc. ACM SIGSAC Conf. Comput. Commun. Secur. (CCS),Dallas, TX, USA, 2017, pp. 229–243, doi:10.1145/3133956.3134060.