Ransomware 3.0: Enhancing Risk Management and Mitigation Options with Proof-of-Decryptability and Smart Contracts

Xinyu Hou
University of Science and Technology of China
houxinyu123@mail.ustc.edu.cn

Yang Lu
University of Houston
ylu17@central.uh.edu

Rabimba Karanjai
University of Houston
rkaranjai@uh.edu

Lei Xu
Kent State University
xuleimath@gmail.com

Larry Shi
University of Houston
wshi3@uh.edu

2025 IEEE International Conference on Blockchain and Cryptocurrency (ICBC) | 979-8-3315-4135-4/25/$31.00 ©2025 IEEE | DOI: 10.1109/ICBC64466.2025.11114675

***Abstract*—Ransomware attacks have become increasingly sophisticated and dangerous, severely impacting individuals and businesses. Traditional ransomware payout models lack trust and security, leaving victims vulnerable to extortion even after paying the ransom. We proposes Ransomware 3.0, a novel blockchain-based ransomware framework that leverages zero-knowledge proofs and smart contracts to overcome these challenges. By integrating these cryptographic techniques, Ransomware 3.0 ensures secure and verifiable data recovery for victims while reducing the risk of data exposure or resale by ransomware attackers. This framework also introduces a contract-based multi-round payment scheme that allows victims to optimize their payout strategy based on the evolving value of their data over time. Counterintuitively, both attackers and victims are motivated to adopt the new model, as the victim is motivated to pay when uncertainty is reduced. We also explore the theoretical decision-making foundations of Ransomware 3.0, analyze its potential benefits and limitations, and discuss its implications for ransomware risk management and mitigation strategies.**

***Index Terms*—ransomware, ZKPs, smart contract**

I. Introduction

In recent years, ransomware has gradually evolved into one of the most widely used types of malicious software [1], [2]. This type of malicious software encrypts the victim's important data, making it impossible to access the system and retrieve the data, and then demands a ransom from the victim in exchange for restoring system functionality and data files. More and more people are falling victim to the ransomware attacks. The potential cost of these attacks is estimated to reach $7.5 billion [3]. Traditional ransomware encrypts information on a victim's computer to demand a ransom payment. It has been modeled in ransomware 1.0. The attacker only demands a ransom and decides whether to return the decryption key to the victim. Ransomware 1.5 introduced data-threat ransomware. In ransomware 2.0 [4], [5], attackers can choose to sell the victim's data for extra profit. With the situation for victims increasingly helpless, We are trying to foretell the coming and likely evolution of ransomware attacks, avoid surprises, provide analysis and decision making tools to help the victims.

The current ransomware model will certainly evolve to take advantage of the advances in technologies in order to attain a new equilibrium between the attacker and the victim. Based on this trend, we predict an emerging ransomware framework built on top of zero-knowledge protocols [6] and smart contracts [7]. We show that there exists significant incentive for the ransom attacker to adopt this new framework. On the other hand, the new framework provides certain benefits to the victim as well such as better assurance of data recovery, more options in terms of risk management, while maintaining the attacker's expected profit. We refer to this blockchain ransomware as 3.0, to distinguish it from the prior ransomware models.

II. Blockchain Based Ransomware Framework

Motivated by the observation that the emerging direction of ransomware could leverage recent advances in blockchain-based technologies and go beyond the practice of using cryptocurrencies for ransom payments, we present a likely advanced ransomware attack framework that applies blockchain fair data exchange and smart contracts.

A. Blockchain Fair Data Exchange and Verifiable Encryption

Researchers have recently applied blockchains and verifiable encryption to achieve Fair Data Exchange (FDE) [8] between a client and a server. In this scenario, a server possesses data files that a client can purchase. By applying verifiable encryption and smart contracts, FDE enables fairness and security guarantees, such that the system assures both parties engaging in the exchange. The client is assured with verifiable encryption that it will receive the requested data after paying to the smart contract. On the other hand, the contract enables the server to receive the client's payment only if it publishes the appropriate decryption key for the client to decrypt the requested data in encrypted format.

B. Assumptions

Blockchain ransomware 3.0 framework is based on a set of assumptions. Firstly, it is assumed that the attacker is financially driven, so the attacker aims to maximize its profits through ransomware attacks. Secondly, it is assumed that the value of the data to the victim may vary over time. Thirdly, we assume that the blockchain used for contract deployment and enforcement is trusted. Fair exchange is impossible without a trusted third party [9]. In this case, blockchain acts as a trusted third party. It is further assumed that the cryptographic schemes in FDE protocols, such as verifiable encryption and cryptographic commitments (e.g., KZG polynomial commitment), are secure. The attacker applies verifiable encryption over-committed keys (VECK) to encrypt the victim's data. Fourthly, it is assumed that the victim can recover the steps used by the attacker. The attacker is motivated to do so because it increases the victim's chance of paying the ransom.

Furthermore, if a payment schedule is used, when the attacker publicizes the victim's data or makes the data available for sale in public, the victim can submit a request to the

979-8-3315-4135-4/25/ $31.00 ©2025 IEEE

979-8-3315-4135-4/25/$31.00 ©2025 IEEE

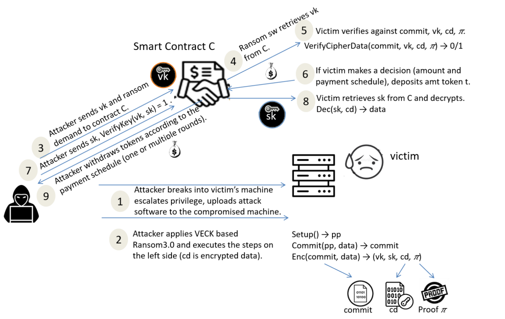
Smart Contract C
vk
sk
Ransom sw retrieves vk from C.
4
5
Victim verifies against commit, vk, cd, π.
VerifyCipherData(commit, vk, cd, π) → 0/1
6
If victim makes a decision (amount and payment schedule), deposits amt token t.
8
Victim retrieves sk from C and decrypts.
Dec(sk, cd) → data
3
Attacker sends vk and ransom demand to contract C.
7
Attacker sends sk, VerifyKey(vk, sk) = 1 .
9
Attacker withdraws tokens according to the payment schedule (one or multiple rounds).
$
victim
1
Attacker breaks into victim's machine escalates privilege, uploads attack software to the compromised machine.
2
Attacker applies VECK based Ransom3.0 and executes the steps on the left side (cd is encrypted data).
Setup() → pp
Commit(pp, data) → commit
Enc(commit, data) → (vk, sk, cd, π)
commit
cd
Proof π

Fig. 1. The high-level blockchain ransomware 3.0 protocol.

contract to cancel all the future payments to the attacker. If a network of oracle providers accepts the request, the contract will stop allowing the attacker to withdraw the remaining ransom and return the remaining money to the victim (after paying the oracle providers for the service). We assume that the majority of the oracle providers are trusted.

In addition, the victim must present evidence to the oracle providers. The evidence can be sent to the oracle providers using the providers' public keys and support of Web3 messaging. It is important to highlight that the victim does not have to disclose his/her data to the oracle providers. There are multiple implementation options to help the oracle providers. These techniques are interesting as research by themselves and beyond the scope of this paper. When making a deposit, the victim can upload a hash of metadata that describes the data.

C. Blockchain Ransomware Framework

Ransomware attacker A targets a victim V, and launches a ransomware 3.0 attack. There is a blockchain B with deployed ransom contract C (e.g., EVM-compatible contract). Fig. 1 demonstrates the process of the protocol. Then, in step 2, A applies a VECK-based scheme to encrypt selected data on the victim's machine. This involves a sequence of actions. Setup() creates security parameters required by the VECK scheme. Then, the attacker computes the commitment value, $commit$, based on the input data. The attacker runs Enc() to encrypt the data, and the results will be: vk (verification key), sk (secret decryption key), cd (ciphertext of the input data), and a proof π. After these actions, the attacker removes the original data and sk from the victim's machine. In step 3, attacker A posts to the contract C, the ransom demand, its wallet account, and vk. The attacker code in the victim's machine prompts the victim for the ransom. It demonstrates to the victim VerifyCipherData($commit$, vk, cd, π) $\rightarrow 0/1$ (step 4 and step 5). In step 6, the victim makes a decision about whether to pay the ransom. In step 7, before the timelock expires, attacker A sends the correct decryption key sk to contract C such that VerifyKey(vk, sk) = 1. In step 8, the victim reads sk from the contract. Using sk, the victim decrypts cd and recovers the data: Dec(sk, cd) → data.

III. Ransom 3.0 Decision Making Model

In this section, we develop a game-theoretical model for the new blockchain ransom 3.0 framework and conduct analysis with the model.

A. Mechanism Design

Compared with the prior ransomware models, the smart contract scheme enables trust and more payment options The contract can split the ransom into several shares and allow partial payment to the attacker across time. Enforced by the contract, if the attacker sells or leaks the data, the remaining ransom will not be paid to the attacker and will be returned to the victim. The incentives for the attacker to adopt such a contract are that the attacker may demand a higher total ransom, and the extra protection introduced against data leakage can positively affect the victim's willingness-to-pay.

The detailed timeline of our mechanism is as follows: **(i)** The attacker A launches a successful attack on victim V. The victim loses access to the original data, and the attacker steals the victim's data. Then, the attacker demands a ransom payment R with the necessary information uploaded to the contract C. **(ii)** When receiving the ransom demand, the victim decides whether to pay the ransom or not. **(iii)** Upon observing the payment transaction from the victim in phase 1, the attacker can enter the next stage. At this point, the attacker has several choices for the data: sell it, leak it to the public, or do nothing (keep the data confidential as promised to the victim). If the victim has evidence that the data is leaked, the remaining ransom will be returned to the victim by the contract. **(iv)** At the beginning of phase i $(1 < i < n)$, the attacker will receive the i-th share of ransom R_i. The attacker will face the same options for the data. If the attacker sells or discloses the data, the victim can submit the evidence to the contract. After verifying, the contract will immediately halt future payments to the attacker and refund the remaining ransom to the victim. **(v)** In the final stage, the n-th stage, when the attacker receives the last ransom share R_n, the process ends.

The new model adopts a multi-phase ransom payment approach to mitigate the risk of the victim. The goal is to prolong the period during which the attacker keeps the data confidential, thus minimizing the victim's potential financial damage caused by data leakage. From the attacker's perspective, the victim will have a higher willingness-to-pay than ransom 2.0.

B. Game Theoretical Decision Framework

The ransomware game is a sequential, multi-stage game involving the attacker and the victim. At the beginning of our model, we let p be the victim's paying decision. If the victim decides to pay the ransom, p equals 1; otherwise, p equals zero. Let $s_i (1 \leq i \leq n)$ be the selling or leaking decision of the attacker in phase i. If the attacker decides to sell the data or leak the data at the beginning of phase i, s_i equals 1. Otherwise, it equals 0. We can encode the choices in each round into a vector $\vec{s} = (s_1, s_2, \ldots, s_n)$. The elements of $\vec{s}$ can only take values from $\{0, 1\}$, and each element of $\vec{s}$ can have at most one '1'.

When the attacker has maintained the confidentiality of the data during the initial $k-1$ stages and subsequently sells the data at the kth stage, our model terminates the multi-round game, and decisions made in the future stages are no longer important. We set the attacker's decisions after selling/disclosing the data to 0 for clarity in later expressions. At this point, $\vec{s} = (0, 0, \ldots, 0, 1, 0, \ldots, 0) = e_k$ where $\vec{s}$ is a vector in the n-dimensional Euclidean space, and only the kth component is 1, all the other components are 0s. We assume that U_i is the attacker's utility in phase i. We can write down its corresponding expression in different phases as follows.

$$U_i = \begin{cases} p(R_1 - C_r + s_1 A_1 + (1-s_1)U_2) + (1-p)A_1 & i=1 \\ R_i + s_i A_i + (1-s_i)U_{i+1} & 2 \leq i \leq n-1 \\ R_n + s_n A_n & i=n \end{cases} \tag{1}$$

where $C_r \geq 0$ is the attacker's cost to return the data to the victim and $R = \sum_{j=1}^{n} R_j$ is the total ransom. $A_i \geq 0$ is the profit for the attacker A when A chooses to sell or leak the data. We assume that the profit for selling the data is D. The total profit could be written as $A_i = DI_{\{D>0\}}$, which means if selling the data is nonprofitable, the attacker chooses to leak the data. Using the recursive formula for the attacker's utility above, we can express the attacker's utility function U_a throughout the entire process as:

$$U_a{=}A_1{+}p[R_1{-}C_r{-}A_1{+}s_1A_1{+}\textstyle\sum_{j=1}^{n-1} s_{j+1}A_{j+1}\prod_{l=1}^{j}(1{-}s_l)] \tag{2}$$

We can also express the victim's corresponding utility U_v based on the different decisions made by the attacker.

$$\begin{aligned} U_v{=}&{-}L_1{+}p[L_1{-}R{+}V{+}s_1(\textstyle\sum_{j=2}^{n} R_j{-}L_1) \\ &{+}\textstyle\sum_{l=2}^{n} s_l(\sum_{j=l+1}^{n} R_j{-}L_l)\prod_{k=1}^{l-1}(1{-}s_k)] \end{aligned} \tag{3}$$

where $L_i(i = 1, 2, \ldots, n)$ is the victim's loss because of data leakage at stage i. V is the utility growth when the victim gets its data back. When the victim forms their utility, they will use the historical records in the blockchain, which contain the attacker's previous transactions. The victims could form a relatively precise probability of the attacker selling the data in each stage. $p_{s,i}$ is the probability of the attacker selling or leaking the data in phase i obtained through the historical records. We call all these probabilities the reputation of the attacker. We assume that such records are uniform with the actual choices made by the attacker.

IV. Analysis of Attackers

A. Attacker with Imperfect Reputation

In ransomware 2.0, the attacker with a positive probability (not 1) to return the data and keep the data confidential is called an attacker with an imperfect reputation. In this section, we will use a mathematical model to draw an analogy to the attacker with an imperfect reputation in the blockchain ransomware model.

Under our assumptions, the attacker will return the data in the first phase. Otherwise, the victim will get the payment back. It is unnecessary in our model to consider attackers to withhold the data and the associated attacker reputation as it does in ransomware 2.0. Recalling the mechanism design in section III-A, we use the probability vector $\vec{p_s} = (p_{s,1}, p_{s,2}, \ldots, p_{s,n})$ to denote the reputation of an attacker. Specifically, the k-th element of the vector $\vec{p_s}$, $p_{s,k}$, is the probability of the attacker selling or leaking the data at the beginning of the k-th phase and keeping the data confidential before the k-th phase. That means that $p_{s,k} = P(\vec{s} = \vec{e_k})\ \forall k \in \{1, 2, \ldots, n\}$. We have some ordinary constraints for the attacker's reputation $\vec{p_s}$.

$$\begin{cases} p_{s,1}{+}p_{s,2}{+}\cdots{+}p_{s_n}{\le}1 \\ p_{s,k}{\ge}0 \quad k{=}1{,}2{,}\ldots{,}n \end{cases} \tag{4}$$

With a fixed reputation, we could write down the expected utility for the attacker:

$$EU_a{=}A_1{+}p[R_1{-}C_r{-}A_1{+}\textstyle\sum_{j=1}^{n} p_{s,j}(\sum_{k=2}^{j} R_k{+}A_j)] \tag{5}$$

where we assume $\sum_{k=2}^{1} R_k = 0$ for simplification.

In this section, only maximizing the utility is not the target for the attacker compared with the short-sighted attacker case. The attacker pays more attention to the future profit. This means that they build their reputation by keeping the data confidential. A good reputation leads to a higher willingness-to-pay from future victims who can read the historical records in the blockchain. We first write down the expected utility of victims.

$$EU_v{=}{-}L_1{+}p[L_1{-}R{+}V{+}\textstyle\sum_{i=1}^{n} p_{s,i}(\sum_{j=i+1}^{n} R_j{-}L_i)] \tag{6}$$

Then, we can summarize the strategies made by the victim as follows:

Corollary 1. *Faced with an attacker whose reputation is $(p_{s,1}, p_{s,2}, \ldots, p_{s,n})$, the victim will pay the ransom if and only if $V + L_1 > R + \sum_{i=1}^{n} p_{s,i}(\sum_{j=i+1}^{n} R_j - L_i)$.*

This can be obtained directly from the expression of the expected utility of the victim (6). Meanwhile we can obtain the willingness-to-pay from the victim, $\mathit{WTP}_v = V + L_1 - R + \sum_{i=1}^{n} p_{s,i}(\sum_{j=i+1}^{n} R_j - L_i)$. In simple terms, the reputation set by the attacker when maximizing the expected profit is largely influenced by the distribution of the victim's data value and the ransom. In this section, we aim to study the optimal reputation for the attacker When the attacker is faced with a victim whose data has value R and its decreasing value in each phase is $(L_1, L_2, \ldots, L_n)$, the attacker's target is the expected utility $EU_a = EU_aI_{(p=1)} + A_1(1 - I_{(p=1)})$ where $I_{(p=1)}$ is the indicator function of a victim's paying decision. The objective function $I = max_{\vec{p}}EU_a$ could be written clearly as follows:

$$\begin{aligned} &I_1{=}\max_{p_{s,1},p_{s,2},\ldots,p_{s,n}} R_1{-}C_r{+}\textstyle\sum_{j=1}^{n} p_{s,i}(A_j{+}\sum_{k=2}^{j} R_k) \\ &\qquad s.t. \quad \begin{cases} \sum_{i=1}^{n} p_{s,i}{\le}1 \\ R{-}\sum_{i=1}^{n} p_{s,i}(\sum_{j=i+1}^{n} R_j{-}L_i){\le}V{+}L_1 \\ 0{\le}p_{s,i}{\le}1 \quad i{\in}\{1{,}2{,}\ldots{,}n\} \end{cases} \\ &I{=}\max\{I_1{,}A_1\} \end{aligned} \tag{7}$$

V. Case Study and Simulations

A. Multi-phase Simulations

In the multi-phase scenario, the optimal reputation that maximizes the attacker's expected utility is highly sensitive to the selection of many variables, such as the ransom amount, the arrangement of ransom payments, as well as the profits that the attacker gains from selling or leaking the data at different stages and the corresponding loss to the victim.

Below, we use the method of controlling variables to show how changes in different variables affect the decisions of both the attacker and the victim, as well as their expected returns.

1) Ransom Amount and Selling Price

We first consider how the ransom amount influences the choices of the attacker. We choose a uniform ransom paying arrangement here which means that the contract should pay the attacker the same amount in different phases. We assume that there are four phases in the whole payment process. The victim should pay the attacker $\frac{R}{4}$ in the beginning of each phase. We assume that in each phase, the profit that the attacker gains from selling or leaking data $A_i, i = 1, 2, \ldots, n$ and the loss incurred to the victim L_i are both proportional to the value of the data V. We select a special case: $A_i = \delta_a^i V, L_i = \delta_l^i V$, where δ_a and δ_l are the discount factors for the attacker's profit and victim's loss.

We first assume a relatively high discount, $\delta_a = 0.9$ and $\delta_l = 0.8$. Figure 2 shows the optimal reputation which the attacker should maintain when charging different amount of ransom ($R = 400, 600, 700, 800$) with the data value $V = 400$. Because of a high discount factor, the profit that the attacker gains from selling the data at different stages does not decrease significantly. Compared to selling the data early for profit that is similar to the profit obtained from selling it later, keeping the data confidential and collecting ransom at each stage is more attractive to the attacker. In this case, the attacker would delay selling the data and aim to collect as much ransom as possible which is shown in Figure 2.

Then we choose lower discount factors $\delta_a = 0.5$ and $\delta_l = 0.4$, so that the value of the data decreases rapidly over time, and plot the corresponding reputation curves of the attacker when demanding different ransom amounts in Figure

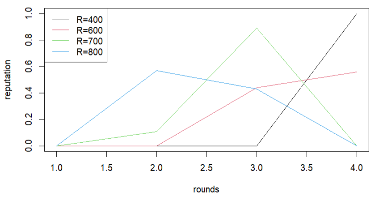

Fig. 2. High discount scenario in multi-phase ransom.

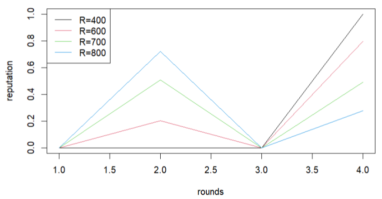

Fig. 3. Low discount scenario in multi-phase ransom.

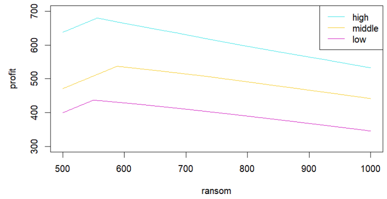

Fig. 4. Profit for different discount factor

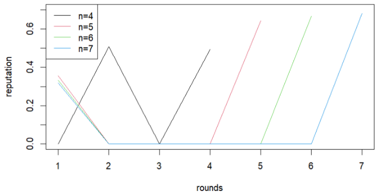

Fig. 5. Optimal attacker reputation under different rounds.

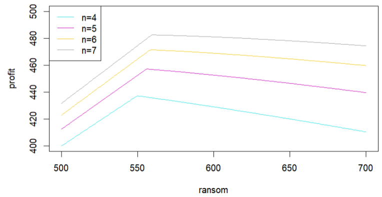

Fig. 6. Attacker profit vs. number of payment rounds.

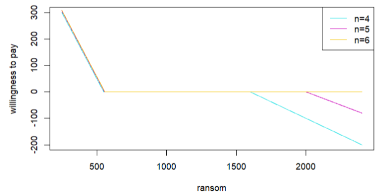

Fig. 7. Victim's willingness-to-pay vs. number of payment rounds.

3. When the ransom amount is not large ($R = 400$), the situation is similar to that with higher discount factorsAs the ransom amount increases, attackers will gradually shift the time of sale from the start of the last stage to the start of the second stage. Although selling the data in the beginning of the second stage incurs some additional ransom loss, the profit from selling two stages earlier is more attractive to the attacker.

One of the key issues in this process is the attacker's expected profit under different scenarios. Figure 4 shows the expected profit of the attacker corresponding to different discount factors as the ransom amount increases. In Figure 4, the cases with high and low discount factors are the same as the previous assumptions, while the moderate case corresponds to $\delta_a = 0.7$ and $\delta_l = 0.6$. We can observe that as the data value decreases at a faster rate over time, the expected return that the attacker can obtain would also decrease.

2) *Number of Rounds*

In the subsection above, we examined the optimal reputation that the attacker should maintain and the corresponding expected profit under different ransom amounts. In this subsection, we will explore the impact of the number of stages on the decisions of both the attacker and the victim.

Figure 5 shows the optimal reputation that the attacker should maintain in order to maximize its expected utility. We select a lower discount factor for the data value here. We assume that the data value V is 400, and the ransom amount R requested by the attacker is 700. The optimal reputation shown in Figure 5 aligns with the previous explanation: a lower discount factor requires the attacker to attain a trade off between obtaining as much ransom as possible and selling the data early to gain higher profit. When the total number of stages increases, the ransom to be paid in each stage decreases. In the first two stages, the file price drops significantly, which may lead the attacker to choose to sell the data early for profit. However, in middle stages, the amount paid in each stage is greater than the decline in the data price, so it is not worthwhile for the attacker to sell the data in these stages. Therefore, the attacker will focus on selling the data at the earlier and later stages.

For the selection of the number of payment stages, we first consider how the attacker's profit changes with the number of stages. Figure 6 shows the attacker's expected profit under different total numbers of stages as the ransom amount increases. It shows that, in this case, as the number of stages increases, the attacker's expected profit continuously increases. From Figure 5, we can observe that with the increase in the total number of stages, the attacker would sell the data with a higher probability in the last stage. The more stages there are, the smaller the loss to the victim caused by the data leakage in the final stage. This increases the victim's willingness-to-pay. This allows the attacker to achieve a relatively lower reputation and still has the victim pay the ransom, ultimately increasing the attacker's expected profit.

Next, let's consider the impact of the number of payment stages on the victim's decision-making. We can plot the curve of the victim's willingness-to-pay under different total numbers of stagesFrom Figure 7, it can be observed that when the ransom amount does not exceed 550, the number of stages has little impact on the victim's decision. The victim's willingness-to-pay is roughly the same, and the decision is to pay the ransom. As the ransom amount increases, in the case of fewer payment stages, the victim starts to refuse to pay ransom at a lower threshold. When the number of payment stages is six, even if the ransom amount exceeds 2000 by a large margin, the victim still has a certain probability of paying the ransom. As mentioned in the previous analysis, with more payment stages, the victim's willingness-to-pay will be higher under the same ransom amount.

VI. Conclusion

In this paper, we present an emerging framework of ransomware attack that applies verifiable encryption (proof-of-decryability) and smart contracts. Comparing with the prior ransomware models (1.0 and 2.0), the new framework is able to achieve improved decisions by minimizing uncertainty and enhancing risk management options. Through game theoretical analysis, it is clear that the blockchain based mechanism helps to ensure that the victim can recover their data with improved prospect and delay the leakage of their data, while simultaneously safeguarding the attacker's expected profit. This suggests that the blockchain ransomware framework will likely supersede the existing ransomware models.

References

[1] P. O'Kane, S. Sezer, and D. Carlin, "Evolution of ransomware," *Iet Networks*, vol. 7, no. 5, pp. 321–327, 2018.

[2] A. Gazet, "Comparative analysis of various ransomware virii," *Journal in computer virology*, vol. 6, pp. 77–90, 2010.

[3] D. Asatryan, "Unprecedented $75 million ransomware payout: Lessons learned and how to protect your data," 2024. [Online]. Available: https://spin.ai/blog/unprecedented-75-million-ransomware-payout/

[4] Z. Li and Q. Liao, "Game theory of data-selling ransomware," *Journal of Cyber Security and Mobility*, pp. 65–96, 2021.

[5] ——, "Ransomware 2.0: to sell, or not to sell a game-theoretical model of data-selling ransomware," in *Proceedings of the 15th International Conference on Availability, Reliability and Security*, 2020, pp. 1–9.

[6] O. Goldreich and Y. Oren, "Definitions and properties of zero-knowledge proof systems," *Journal of Cryptology*, vol. 7, no. 1, pp. 1–32, 1994.

[7] S. Wang, Y. Yuan, X. Wang, J. Li, R. Qin, and F.-Y. Wang, "An overview of smart contract: architecture, applications, and future trends," in *2018 IEEE Intelligent Vehicles Symposium (IV)*. IEEE, 2018, pp. 108–113.

[8] E. N. Tas, I. A. Seres, Y. Zhang, M. Melczer, M. Kelkar, J. Bonneau, and V. Nikolaenko, "Atomic and fair data exchange via blockchain," Cryptology ePrint Archive, Paper 2024/418, 2024. [Online]. Available: https://eprint.iacr.org/2024/418

[9] H. Pagnia and F. C. G. Darmstadt, "On the impossibility of fair exchange without a trusted third party," 1999. [Online]. Available: https://api.semanticscholar.org/CorpusID:11671049