# On Countering Ransomware Attacks Using Strategic Deception

Roshan Lal Neupane[1(✉)], Bishnu Bhusal[1], Kiran Neupane[1], Preyea Regmi[1], Tam Dinh[1], Lilliana Marrero[1], Sayed M. Saghaian N. E.[1], Venkata Sriram Siddhardh Nadendla[2], and Prasad Calyam[1]

[1] University of Missouri, Columbia, MO 65211, USA
{neupaner,bhusalb,kngbq,prrgfb,tdbhr,lmmbd8,ssddd,calyamp}@missouri.edu
[2] Missouri University of Science and Technology, Rolla, MO 65409, USA
nadendla@mst.edu

**Abstract.** Ransomware attacks continue to be a major concern for critical systems that are vital for society e.g., healthcare, finance, and transportation. Traditional cyber defense mechanisms fail to pose dynamic measures to stop ransomware attacks from progressing through various stages in the attack process. To this end, intelligent cyber deception strategies can be effective when they leverage information about attacker strategies and deploy deceptive assets to increase the cost or complexity of a successful exploit or discourage continued attacker efforts. In this paper, we present a novel game theoretic approach that uses deception-based defense strategies at each of the ransomware attack stages for optimization of the decision-making to outsmart attacker advances. Specifically, we propose a multistage ransomware game model that deploys a combination of deception assets i.e., honeytokens, honeypots, honeyfiles, and network honeypots in subgames. Using closed-form backward induction, we evaluated Subgame-Perfect Nash Equilibrium (SPNE). We perform a numerical analysis using real-world data and statistics pertaining to the impact of ransomware attacks in the healthcare sector. Our healthcare case study evaluation results show that the use of deception technologies is favorable to the defender. This work elucidates the profound implications of strategic deception in cybersecurity, demonstrating its capacity to complicate successful exploits and consequently bolster the defense of key societal infrastructures.

**Keywords:** ransomware · cyber deception · game theory · attacker/defender game

## 1 Introduction

Deception plays a crucial role in the ever-evolving landscape of ransomware attacks. Deception techniques, such as honey tokens, honeypots, honey files, and network honeypots, are vital tools in combating those attacks [29] so that defenders can gain a competitive edge in the intricate game of ransomware,

A. Sinha et al. (Eds.): GameSec 2024, LNCS 14908, pp. 149–176, 2025.
https://doi.org/10.1007/978-3-031-74835-6_8

disrupting the attackers' strategies and turning the tables in favor of cybersecurity resilience. This paper presents a novel approach to ransomware defense via modeling the attacker-defender interaction as a multi-stage game and investigates effective techniques to deceive the ransomware attacker at each stage of the attack using tools such as honeytokens, honeypots, honeyfiles, and network honeypots.

Ransomware is a malware type that encrypts target users' data, rendering it inaccessible without a decryption key that the attacker exclusively holds. Typically, victims are directed to pay a ransom to decrypt their data [37]. The Internet Crime Report published by the FBI lists 2385 registered complaints for Ransomware attacks in the year 2022 [3] alone, which has led to economic losses that totalled to more than $34.3 million. These attacks have impacted diverse application domains ranging from healthcare (e.g. emergency departments in San Diego County's healthcare delivery organizations [15]), critical infrastructure (e.g. Colonial Pipeline attack [30]), transportation (e.g. ransomware attacks on Toyota and Kojima), government facilities (e.g. ransomware attacks on the City of Detroit and Washington DC police department), IT (e.g. Acer), and finance (e.g. supply chain attack [13] and Travelex).

To make things worse, ransomware is also offered as a service by some malicious organizations. One such example is REvil [19] (which stands for Ransomware Evil), which is a ransomware-as-a-service (RaaS) platform that has been run by organized criminal groups in Russia. Given the widespread impact of ransomware attacks on diverse application domains, there is an urgent need to investigate advanced defense and impact mitigation techniques to counter these attacks.

In the past, ransomware has been studied as a multi-stage game [42] to thoroughly model the diverse interactions between the attacker and the defender in time. While a detailed account of the past literature on game-theoretic approaches to counter ransomware attacks is presented in Sect. 2.1, there is little work on the design of strategic deception to counter ransomware attacks to the best of our knowledge. Therefore, this paper is the first of its kind to investigate the effectiveness of deception techniques (e.g. honeypots, honeytokens and honeyfiles) against ransomware attacks using multi-stage games. The optimal strategy at each of the stages are computed using backward induction [11] via breaking the game into smaller and manageable sub-games. Such an approach leads to the identification of subgame perfect equilibria where optimal strategies are met in each subgame.

The main contributions of this paper are three-fold. Firstly, the proposed multi-stage game is comprehensive and considers most stages of real-world ransomware attacks (e.g. infection, installation and encryption) and their corresponding defenders' deception assets (e.g. honeytokens, honeypots and honeyfiles) in real-world adversarial environments. In addition, the model also encompasses sub-stages for ransom payment and arrest, aiming to capture attackers and limit defender losses. Secondly, the best-response strategies for both the attacker and the defender are formally evaluated except for the root node. At the root node, the best-response search reduces to a binary quadratic program, which is solved using state-of-the-art integer-programming methods. Thirdly, the SPNE is algorithmically evaluated in numerical experiments, and results are discussed in comparison with state-of-the-art literature.

The remainder of this paper is organized as follows: Sect. 2 presents background of related works and cyber deception techniques. Section 3 models the interaction between a ransomware attacker and a defender. Section 4 presents an equilibrium analysis. Section 5 makes a case for healthcare industry. Section 6 details the numerical results, comparison, and their analysis. Section 7 concludes the paper.

## 2 Background

In this section, we go over the related literature and background on the deception techniques applicable in mitigating different stages in a ransomware attack.

### 2.1 Related Works

The related literature is presented briefly in three different themes and gaps are identified in the state-of-the-art to defend against ransomware attacks.

**Ransomware Detection and Defense.** There have been many approaches to the detection and defense of ransomware attacks. Kolodenker *et al.* in [20] presented a prototype called *Paybreak* that is able to recover files by decrypting them based on insights gained during the process of secure file encryption. A data backup solution is used in [24], called *AMOEBA* which has high ransomware detection accuracy with negligible performance overhead on the backup process. *Ransomwall* [33] is a machine learning approach to defend against ransomware attacks by learning suspicious ransomware behavior processes to initiate a data backup for preserving user data. Patyal *et al.* in [28] proposed a multi-layer architecture defense, with each layer employing different techniques starting from improved policies for enhanced security, recursive folder creation for ransomware detection, process monitoring, to data backup.

These methods target specific stages of a ransomware attack and fall short of considering every stage the attack process for defense. Our novelty lies in considering every stage of the process for the defense against the attack using a game theoretic approach.

**Game Theoretic Approaches for Ransomware.** There are several forms of game-theoretic models that are applied for the mitigation of ransomware attacks. Authors in [14] treat the problem by presenting finances as the primary motive of the attacker via the models developed by [32] and [21]. Authors in [42] present a multi-stage game that comprehensively models ransomware attack and defense with a fully observable environment set up. To combat ransomware in IoT, authors in [43] present a multi-stage game framework for cyber and economic phases of a ransomware attack. Similarly, there are several consequences discussed in how the stages progress and how the outcome of the previous stage sets the ground for the upcoming stage. Different tangents are discussed in the articles such as [22,40] where the ransomware attacks are dealt in terms of attacker-defender game, defender-insurer game where the strategies and ransom amounts differ based on the strategy applied by each of the participant in the game.

There are different approaches to defend against ransomware attacks using game theory models showcased in the literature mentioned. Our novelty lies in considering deception techniques as defender strategies within various stages of a ransomware attack and defense process.

**Deception-Based Defense Against Ransomware.** Authors in [16] pose a solution that uses deception to stop a crypto ransomware attack with minimal spatial or system computation requirement. They deploy a honeyfile that recognizes when an API function is called between software components using a hook and a monitoring file that checks whether the honeyfile has been encrypted or not. Another mitigation technique implemented is called *R-Sentry* [34] that aids how to optimally place honeyfiles based on the file traversal patterns of ransomware variants. Some authors leverage a combination of these tools to come up with a stronger deception-based defense that they claim as an auxiliary ransomware traceable system called *RansomTracer* [38]. Authors in [36] present a stealthy approach to backing up data in order to isolate them from the attacker no matter the level of privilege acquired by them attacker. *RTrap* in [17] is a systematic strategy that utilizes machine learning to create deceptive files, luring attackers or ransomware to access them upon detecting potential access.

As a multi-stage process that involves infection, installation, encryption, data movement or deletion, etc., ransomware attacks need a more robust defense system that considers every stage of the process. Our novelty lies in considering deception for each of the attack stage processes with state-of-the-art deception technologies that we discuss in the next section.

### 2.2 Cyber Deception Techniques

There are different deception techniques that can be used to deceive attackers accomplished by deliberate placing or positioning of resources that look real and of interest to the attackers. Similar attempts are made by ransomware attackers as well. For safeguarding the system from ransomware attack, we leverage three types of deception techniques. These are:

**Honeytokens.** Honeytokens are artifacts such as e.g., access tokens, credentials that can be strategically placed in the organizational network (such as e.g., in data stores, code base) for attackers to use [31]. There are various available tools that can generate such honeytokens such as e.g., HoneyGen [12], Canary Tokens [1], SpaceSiren [8]. These tokens are designed such that when they are triggered, they can alert security teams or simply lead the attackers to deception systems such as honeypots.

**Honeypots.** Honeypots [35] are systems or software applications that are built to monitor hacker activities, or interact with them depending on the level of interactions (low, medium, or high) [25]. A high-interaction honeypot is able to deploy real network services, applications, and operating systems. This can aid in capturing extensive information. In the context of ransomware, attackers can be deceived into infecting and installing malware into the decoy honeypot and subsequently encrypting files that are of no use by containing them in a honeypot. Some examples are: low-interaction (Glastopf [26]), medium-interaction (Kippo [6]), and high-interaction (HIHAT [27]).

Network honeypots can be leveraged for performing protocol inspections to monitor network traces. In the ransomware defense context, we use network honeypots as a strategy to monitor exfiltration of data with the hopes of redirecting the exfiltration attempts to a controlled system and not to the attacker's intended system.

**Honeyfiles.** Honeyfiles [41], similar to other deception methods are artifacts that can be used as baits to grab an attacker's attention. In the context of the paper, we can leverage honeyfiles portraying them as real files the ransomware attackers might want to encrypt. Encrypting these files can lead to triggering of alarm, or can simply be treated as a fail-safe for the encryption phase of the ransomware attack cycle. To delay the detection of honeyfiles, there are advanced

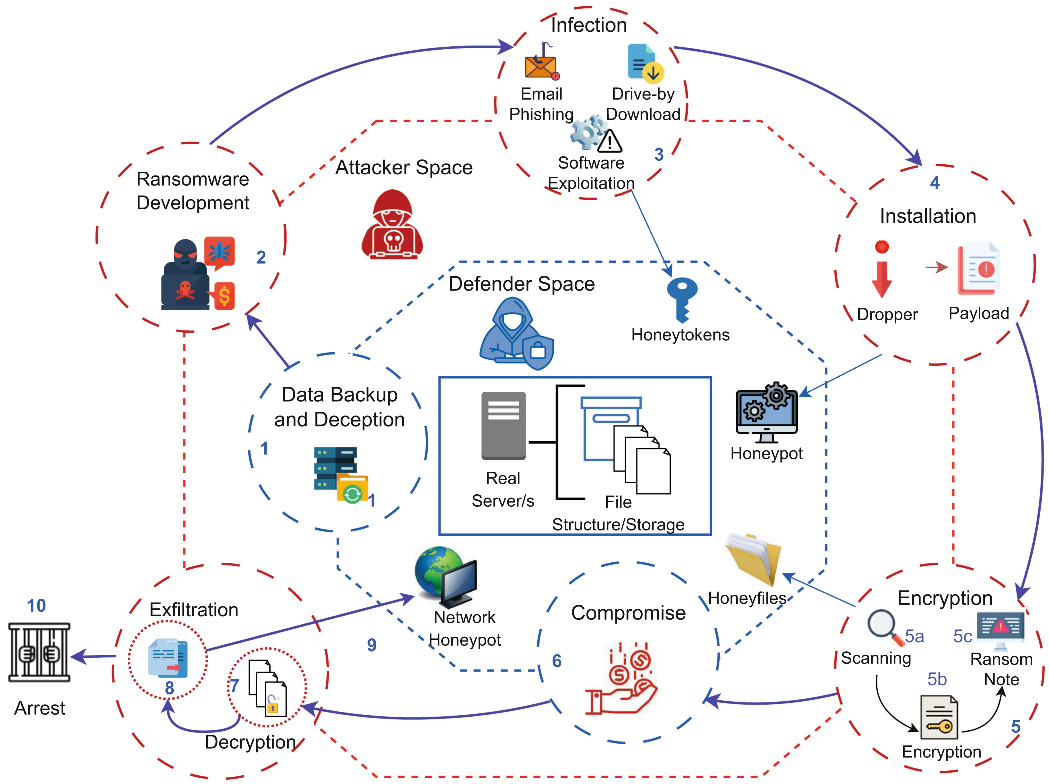

**Fig. 1.** Attacker and defender interactions in the ransomware game model.

methods such as in [23], where Generative Adversarial Networks (GANs) have been shown to be useful to create effective deception against ransomware using decoy/honeyfiles. RLocker [18] is an example honeyfile-based deception tool.

## 3 Ransomware Game Model

Consider an adversarial setting with two agents, a ransomware attacker and a defender, and a system-of-interest with $N$ subsystems containing valuable data. Let $V_i$ denote the value of data present in the $i^{th}$ subsystem. The attacker wishes to lock one/more subsystems within a system-of-interest for certain ransom. On the other hand, the defender's goal is to safeguard the entire system from the ransomware threat using three deception techniques, namely *honeytokens*, *honeypots* and *honeyfiles*. This interaction between the attacker and the defender occurs in multiple stages, as illustrated in Fig. 1, and discussed below. For the sake of reader's convenience, a table of notation is also included in Table 1.

**Stage 1 - Data Backup and Deception:** In this first stage, the defender makes a binary decision $b_i$ whether or not to backup the data on the $i^{th}$ subsystem upon system deployment. Without any loss of generality, let $b_i = 1$ denote the defender's decision to backup data, in which case, the defender incurs a cost $B$, accounting for cost of data back up and recovery. In other words, the ransomware attack is destined to fail if $b_i = 1$. On the contrary, the attacker may launch a successful ransomware attack if $b_i = 0$. Furthermore, in an attempt to protect the system, the defender has to choose whether or not to deploy a *three-pronged deception technique using honeytokens, honeypots, and honeyfiles*, to prevent the ransomware attacker from gaining access into the $i^{th}$ subsystem. Let $h_i = 1$ denote the binary decision to develop and deploy the aforementioned three-pronged deception strategy, and $h_i = 0$ denote otherwise. If $h_i = 1$, assume that the defender deploys $n_{ht}$ honeytokens, one honeypot and $n_{hf}$ honeyfiles in order to deceive the ransomware attacker. In other words, the defender's decision is a tuple $\Big((b_1, h_1), \cdots, (b_i, h_i), \cdots, (b_N, h_N)\Big) \in \{0, 1\}^{2N}$. Without any loss of generality, for each subsystem, let $B$ be the cost of data backup, $C_{ht}$ denotes the cost of deploying a single honeytoken, $C_{hp}$ is the cost of deploying the honeypot, and $C_{hf}$ represents the cost of deploying a single honeyfile. For simplicity, we denote the total cost of deception as

$$C_H = n_{ht} \cdot C_{ht} + C_{hp} + n_{hf} \cdot C_{hf}. \tag{1}$$

In addition to the three aforementioned deception strategies, assume that the defender always deploys an additional network honeypot to deceive attackers from data exfiltration [39].

**Table 1.** Notations used in this paper

| Notation | Description |
|---|---|
| $B$ | Cost of data backup |
| $C_{ht}$ | Cost of honeytokens deployment |
| $n_{ht}$ | Number of honeytokens deployed |
| $C_{hp}$ | Cost of honeypot deployment |
| $C_{hf}$ | Cost of honeyfiles deployment |
| $n_{hf}$ | Number of honeyfiles deployed |
| $C_{nh}$ | Cost of network honeypot deployment |
| $C_H$ | Total deception cost |
| $C_D$ | Cost of ransomware development |
| $C_x$ | Cost of exfiltration |
| $\rho$ | Probability of successful honeytoken-based deception |
| $\tau$ | Probability of successful honeypot-based deception |
| $\gamma$ | Probability of successful honeyfile-based deception |
| $\zeta$ | Probability of successful network honeypot-based deception |
| $V_i$ | Value of data owned by $i^{th}$ target for the defender |
| $V_i'$ | Value of data for the attacker after exfiltration |
| $V_p$ | Value of privacy of data |
| $R_i$ | Ransom demand proposed by attacker to $i^{th}$ target |
| $b_i$ | Defender's decision on data backup |
| $d_i$ | Attacker's decision on ransomware development |
| $c_i$ | Defender's decision on whether to compromise |
| $e_i$ | Attacker's decision on whether to decrypt of data |
| $x_i$ | Attacker's decision on whether to exfiltrate the data |
| $p_0$ | Natural probability of attacker being arrested when $e_i = 1$, $x_i = 0$ |
| $p_1$ | Natural probability of attacker being arrested when $e_i = 1$, $x_i = 1$ |
| $p_2$ | Natural probability of attacker being arrested when $e_i = 0$, $x_i = 0$ |
| $p_3$ | Natural probability of attacker being arrested when $e_i = 0$, $x_i = 1$ |
| $T$ | Attacker's reputation. $T > 0$ if attacker decrypts data after defender pays, or attacker does not decrypt data when defender does not pay. $T = 0$ for all other cases |
| $F$ | Loss of attacker for being arrested ($F > 0$) |

**Stage 2 - Ransomware Development/Delivery:** When no data backup is present, the attacker decides whether or not to develop ransomware. Let the decision to develop a ransomware for the $i^{th}$ subsystem be denoted as $d_i$. If $d_i = 0$, i.e. if the attacker's choice is to not develop ransomware, the game ends. On the other hand, when the attacker chooses to develop ransomware (i.e.

$d_i = 1$), the attacker develops the ransomware attack and incurs a cost $C_D$ for the development of ransomware.

**Stage 3 - Infection:** Upon the development of ransomware, the attacker uses diverse delivery mechanisms such as email phishing, drive-by download, or software vulnerability exploitation approaches, to infect the desired subsystem. The *honeytokens* lure the attacker into using fake access tokens with independent and identical Bernoulli distribution with probability

$$\rho = 1 - e^{-\frac{C_{ht} * n_{ht}}{C_D}}. \tag{2}$$

The exponential function suggests that the probability decreases exponentially with the product of the cost of honeytokens ($C_{ht}$), the number of honeytokens ($n_{ht}$), and the reciprocal of the cost of ransomware development ($C_D$). All these parameters influence the likelihood of the attacker getting deceived by the usage of honeytokens.

**Stage 4 - Malware Installation:** Once the attacker has access to some server, the next step in the attack process will be the execution of a dropper. The dropper program leads to running of a successful installation of ransomware payload to the victim's computer. To counter the dropper program at each subsystem, honeypots lure the attacker into installing their malware in a fake server randomly with probability

$$\tau = 1 - e^{-\frac{C_{hp}}{C_D}}. \tag{3}$$

The exponential function suggests that the probability decreases exponentially with the cost of deploying honeypots ($C_{hp}$), and the reciprocal of the cost of ransomware development ($C_D$). All these parameters influence the likelihood of the attacker getting deceived by the usage of honeyfiles during the malware installation stage.

**Stage 5 - Encryption:** Once the attacker gains access to either the original file structure or the honeypot within a given subsystem, the ransomware scans for specific files that are deemed valuable, and locks them using a robust encryption algorithm to restrict user access. However, the honeyfiles deployed by the defender in Stage 3 can steer the attacker away from the actual subsystem, and entices the attacker to encrypt them with probability

$$\gamma = 1 - e^{-\frac{C_{hf} * n_{hf}}{C_D}}. \tag{4}$$

The exponential function suggests that the probability decreases exponentially with the product of the cost of honeyfiles ($C_{hf}$), the number of honeyfiles ($n_{hf}$), and the reciprocal of the cost of ransomware development ($C_D$). All these parameters influence the likelihood of the attacker getting deceived by the usage of honeyfiles during the encryption stage. Upon successful encryption, a note demanding a ransom of $R_i$ for the release of the $i^{th}$ subsystem is sent to the user.

Once the attacker has access to the data, they can engage in exfiltration at any stage of the game, even before administering the ransomware in the file system. It is likely for attackers to perform data exfiltration before administering the ransomware in the file system. For this work, we are discussing a generic ransomware setting, such as the one discussed in [2], where the attacker can release data after the compromise stage regardless of the compromise outcome.

**Stage 6 - Compromise:** Upon successful lock-down and the receipt of a ransom note, the defender makes a binary decision $c_i$ regarding the payment of ransom. Let $c_i = 1$ denote the decision to pay the ransom, and $c_i = 0$ otherwise. However, if the attacker was successfully deceived (i.e. the attacker encrypted honeyfiles), the defender will not pay the ransom.

**Stage 7 - Decryption:** If the defender decides to compromise and pay the ransom, the attacker decides whether or not to decrypt the data and release the subsystem back to the defender. Let $e_i = 1$ denote the decision to decrypt the subsystem and give back access to the defender. Otherwise, if $e_i = 0$, the attacker will not decrypt the data and the defender loses the data permanently. Note that if the attacker keeps the promise (i.e. decrypts data upon receiving the ransom, or does not decrypt if the ransom is not paid), the attacker gains a reputation $T$. Otherwise, the attacker receives a zero reputation.

**Stage 8 - Exfiltration:** In addition to collecting ransom, assume that the attacker may also exfiltrate data and cause privacy breach. Let $x_i = 1$ denote the decision to exfiltrate the data from the $i^{th}$ subsystem, and $x_i = 0$ otherwise. If the attacker chooses to exfiltrate (i.e. $x_i = 1$), then the attacker incurs a cost of $C_x$ for each subsystem.

**Stage 9 - Exfiltration Deception:** During exfiltration (i.e. when $x_i = 1$), the attacker moves the data/files to another database through a network. In order to prevent successful exfitration, the network honeypot lures the attacker into moving exfiltrated data through a fake network with probability

$$\zeta = 1 - e^{-\frac{C_{nh}}{C_D}}. \tag{5}$$

If the data in the $i^{th}$ subsystem is successfully exfiltrated, the attacker obtains a value of $V_i'$. On the other hand, the defender incurs a cost $V_p$ for the privacy breach.

**Stage 10 - Arrest:** Depending on the defender's decision to pay ransom $c_i$ and the attacker's decryption and exfiltration decisions $(e_i, x_i)$ respectively, the attacker may get identified, caught and arrested with a different probability according to one of the following three cases: (i) Let $p_0$ denote the probability of the attacker getting arrested after decrypting the data ($e_i = 1$), while not performing exfiltration ($x_i = 0$), (ii) Let $p_1$ denote the probability of the attacker getting arrested after decrypting the data ($e_i = 1$) and performing exfiltration ($x_i = 1$), (iii) Let $p_2$ denote the probability of the attacker getting arrested upon deciding not to decrypt the data ($e_i = 0$) and not performing exfiltration ($x_i = 0$), and (iv) Let $p_3$ denote the probability of the attacker getting arrested

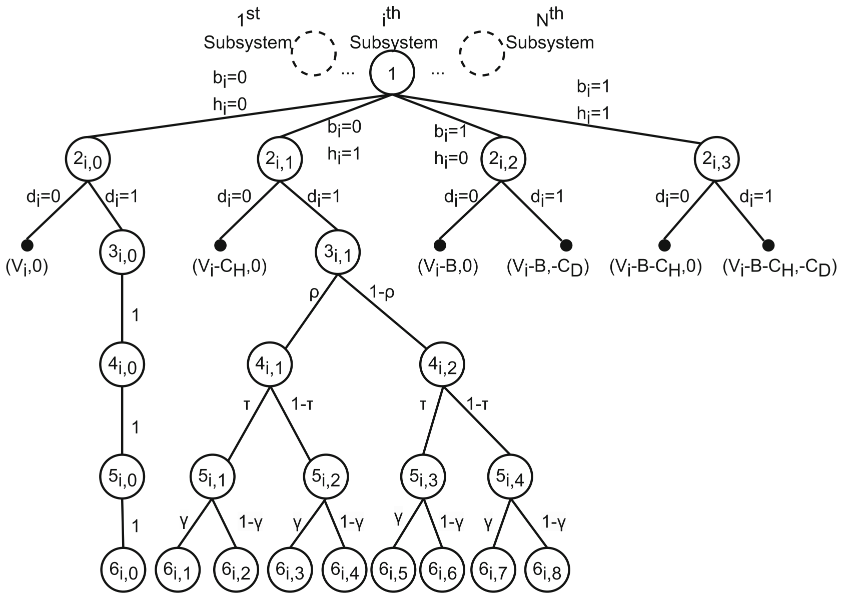

**Fig. 2.** Subtree of the attacker-defender game, comprising of Stages 1–6.

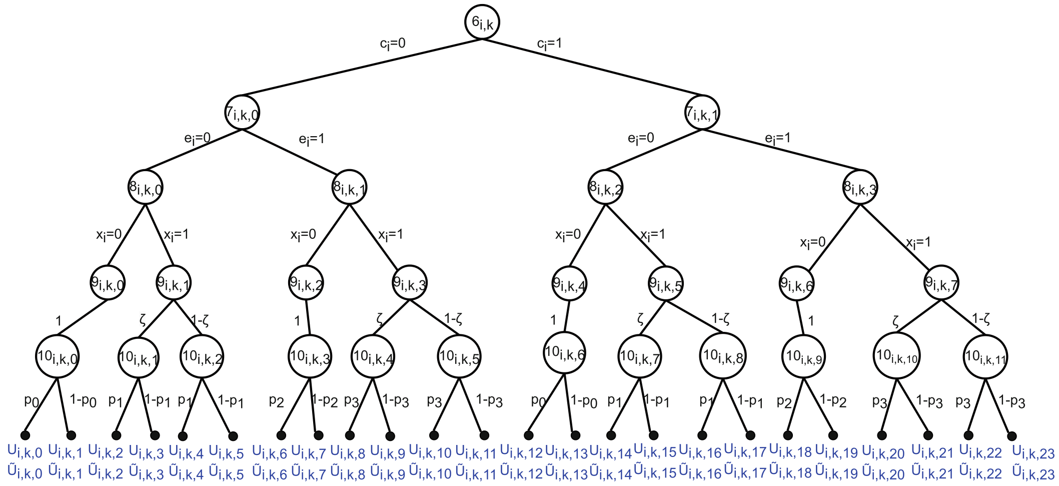
$6_{i,k}$
$c_i=0$
$c_i=1$
$7_{i,k,0}$
$7_{i,k,1}$
$e_i=0$
$e_i=1$
$8_{i,k,0}$
$8_{i,k,1}$
$8_{i,k,2}$
$8_{i,k,3}$
$x_i=0$
$x_i=1$
$9_{i,k,0}$
$9_{i,k,1}$
$9_{i,k,2}$
$9_{i,k,3}$
$9_{i,k,4}$
$9_{i,k,5}$
$9_{i,k,6}$
$9_{i,k,7}$
1
$\zeta$
$1-\zeta$
$10_{i,k,0}$
$10_{i,k,1}$
$10_{i,k,2}$
$10_{i,k,3}$
$10_{i,k,4}$
$10_{i,k,5}$
$10_{i,k,6}$
$10_{i,k,7}$
$10_{i,k,8}$
$10_{i,k,9}$
$10_{i,k,10}$
$10_{i,k,11}$
$p_0$ $1-p_0$ $p_1$ $1-p_1$ $p_1$ $1-p_1$ $p_2$ $1-p_2$ $p_3$ $1-p_3$ $p_3$ $1-p_3$ $p_0$ $1-p_0$ $p_1$ $1-p_1$ $p_1$ $1-p_1$ $p_2$ $1-p_2$ $p_3$ $1-p_3$ $p_3$ $1-p_3$
$U_{i,k,0}$ $\tilde{U}_{i,k,0}$
$U_{i,k,1}$ $\tilde{U}_{i,k,1}$
$U_{i,k,2}$ $\tilde{U}_{i,k,2}$
$U_{i,k,3}$ $\tilde{U}_{i,k,3}$
$U_{i,k,4}$ $\tilde{U}_{i,k,4}$
$U_{i,k,5}$ $\tilde{U}_{i,k,5}$
$U_{i,k,6}$ $\tilde{U}_{i,k,6}$
$U_{i,k,7}$ $\tilde{U}_{i,k,7}$
$U_{i,k,8}$ $\tilde{U}_{i,k,8}$
$U_{i,k,9}$ $\tilde{U}_{i,k,9}$
$U_{i,k,10}$ $\tilde{U}_{i,k,10}$
$U_{i,k,11}$ $\tilde{U}_{i,k,11}$
$U_{i,k,12}$ $\tilde{U}_{i,k,12}$
$U_{i,k,13}$ $\tilde{U}_{i,k,13}$
$U_{i,k,14}$ $\tilde{U}_{i,k,14}$
$U_{i,k,15}$ $\tilde{U}_{i,k,15}$
$U_{i,k,16}$ $\tilde{U}_{i,k,16}$
$U_{i,k,17}$ $\tilde{U}_{i,k,17}$
$U_{i,k,18}$ $\tilde{U}_{i,k,18}$
$U_{i,k,19}$ $\tilde{U}_{i,k,19}$
$U_{i,k,20}$ $\tilde{U}_{i,k,20}$
$U_{i,k,21}$ $\tilde{U}_{i,k,21}$
$U_{i,k,22}$ $\tilde{U}_{i,k,22}$
$U_{i,k,23}$ $\tilde{U}_{i,k,23}$

**Fig. 3.** Subtree of the Attacker-Defender Game, comprising of Stages 6–10.

upon deciding not to decrypt the data ($e_i = 0$), but perform exfiltration ($x_i = 1$). If arrested, the attacker incurs a large cost of $F$.

**Remark:** Note that Stages 3, 4, 5, 9, and 10 are chance stages, which introduce uncertainty in the interaction outcome. This uncertainty emerges from the inherent stochasticity present within the interaction, but does not arise due to any agent's decision.

The above multi-stage interaction between the ransomware attacker and the defender is modeled as a complete-information extensive-form game $\Gamma = \{\mathcal{N}, G, \mathcal{U}, \tilde{\mathcal{U}}\}$, where

- $\mathcal{N} = \{D, A\}$ comprises of the two players ($D$ stands for defender and $A$ stands for attacker),
- $G$ represents the decision tree shown in Figs. 2 and 3 that includes the play-order, chance probabilities, and strategies at both attacker and defender, and
- $\mathcal{U}$ and $\tilde{\mathcal{U}}$ denotes the utility functions at the defender and attacker respectively, which are defined in Table 2.

The goal of this paper is to evaluate the subgame-perfect Nash equilibrium (SPNE) for the game $\Gamma$. A closed-form equilibrium analysis is presented in the following section using backward induction principles.

**Table 2.** Utility functions and their payoffs

| | | |
|---|---|---|
| $U_{i,k,0} = \begin{cases} 0, & \text{if } k = 0 \\ -C_H, & \text{otherwise} \end{cases}$<br>$\tilde{U}_{i,k,0} = T - C_D - F$ | $U_{i,k,1} = \begin{cases} 0, & \text{if } k = 0 \\ -C_H, & \text{otherwise} \end{cases}$<br>$\tilde{U}_{i,k,1} = T - C_D$ | $U_{i,k,2} = \begin{cases} 0, & \text{if } k = 0 \\ -C_H, & \text{otherwise} \end{cases}$<br>$\tilde{U}_{i,k,2} = T - C_D - F - C_x$ |
| $U_{i,k,3} = \begin{cases} 0, & \text{if } k = 0 \\ -C_H, & \text{otherwise} \end{cases}$<br>$\tilde{U}_{i,k,3} = T - C_D - C_x$ | $U_{i,k,4} = \begin{cases} -V_p, & \text{if } k = 0 \\ -V_p - C_H, & \text{otherwise} \end{cases}$<br>$\tilde{U}_{i,k,4} = T - C_D + V_i' - F - C_x$ | $U_{i,k,5} = \begin{cases} -V_p, & \text{if } k = 0 \\ -V_p - C_H, & \text{otherwise} \end{cases}$<br>$\tilde{U}_{i,k,5} = T - C_D + V_i' - C_x$ |
| $U_{i,k,6} = \begin{cases} V_i, & \text{if } k = 0 \\ V_i - C_H, & \text{otherwise} \end{cases}$<br>$\tilde{U}_{i,k,6} = -C_D - F$ | $U_{i,k,7} = \begin{cases} V_i, & \text{if } k = 0 \\ V_i - C_H, & \text{otherwise} \end{cases}$<br>$\tilde{U}_{i,k,7} = -C_D$ | $U_{i,k,8} = \begin{cases} V_i, & \text{if } k = 0 \\ V_i - C_H, & \text{otherwise} \end{cases}$<br>$\tilde{U}_{i,k,8} = -C_D - F - C_x$ |
| $U_{i,k,9} = \begin{cases} V_i, & \text{if } k = 0 \\ V_i - C_H, & \text{otherwise} \end{cases}$<br>$\tilde{U}_{i,k,9} = -C_D - C_x$ | $U_{i,k,10} = \begin{cases} V_i - V_p, & \text{if } k = 0 \\ V_i - V_p - C_H, & \text{otherwise} \end{cases}$<br>$\tilde{U}_{i,k,10} = -C_D + V_i' - F - C_x$ | $U_{i,k,11} = \begin{cases} V_i - V_p, & \text{if } k = 0 \\ V_i - V_p - C_H, & \text{otherwise} \end{cases}$<br>$\tilde{U}_{i,k,11} = -C_D + V_i' - C_x$ |
| $U_{i,k,12} = \begin{cases} -R_i, & \text{if } k = 0 \\ -R_i - C_H, & \text{otherwise} \end{cases}$<br>$\tilde{U}_{i,k,12} = R_i - C_D - F$ | $U_{i,k,13} = \begin{cases} -R_i, & \text{if } k = 0 \\ -R_i - C_H, & \text{otherwise} \end{cases}$<br>$\tilde{U}_{i,k,13} = R_i - C_D$ | $U_{i,k,14} = \begin{cases} -R_i, & \text{if } k = 0 \\ -R_i - C_H, & \text{otherwise} \end{cases}$<br>$\tilde{U}_{i,k,14} = R_i - C_D - F - C_x$ |
| $U_{i,k,15} = \begin{cases} -R_i, & \text{if } k = 0 \\ -R_i - C_H, & \text{otherwise} \end{cases}$<br>$\tilde{U}_{i,k,15} = R_i - C_D - C_x$ | $U_{i,k,16} = \begin{cases} -R_i - V_p, & \text{if } k = 0 \\ -R_i - V_p - C_H, & \text{otherwise} \end{cases}$<br>$\tilde{U}_{i,k,16} = R_i - C_D + V_i' - F - C_x$ | $U_{i,k,17} = \begin{cases} -R_i - V_p, & \text{if } k = 0 \\ -R_i - V_p - C_H, & \text{otherwise} \end{cases}$<br>$\tilde{U}_{i,k,17} = R_i - C_D + V_i' - C_x$ |
| $U_{i,k,18} = \begin{cases} -R_i + V_i, & \text{if } k = 0 \\ V_i - R_i - C_H, & \text{otherwise} \end{cases}$<br>$\tilde{U}_{i,k,18} = R_i + T - C_D - F$ | $U_{i,k,19} = \begin{cases} V_i - R_i, & \text{if } k = 0 \\ V_i - R_i - C_H, & \text{otherwise} \end{cases}$<br>$\tilde{U}_{i,k,19} = R_i + T - C_D$ | $U_{i,k,20} = \begin{cases} V_i - R_i, & \text{if } k = 0 \\ V_i - R_i - C_H, & \text{otherwise} \end{cases}$<br>$\tilde{U}_{i,k,20} = R_i + T - C_D - F - C_x$ |
| $U_{i,k,21} = \begin{cases} V_i - R_i, & \text{if } k = 0 \\ V_i - R_i - C_H, & \text{otherwise} \end{cases}$<br>$\tilde{U}_{i,k,21} = T + R_i - C_D - C_x$ | $U_{i,k,22} = \begin{cases} V_i - R_i - V_p, & \text{if } k = 0 \\ V_i - R_i - V_p - C_H, & \text{otherwise} \end{cases}$<br>$\tilde{U}_{i,k,22} = R_i + T - C_D + V_i' - F - C_x$ | $U_{i,k,23} = \begin{cases} V_i - R_i - V_p, & \text{if } k = 0 \\ V_i - R_i - V_p - C_H, & \text{otherwise} \end{cases}$<br>$\tilde{U}_{i,k,23} = R_i + T - C_D + V_i' - C_x$ |

## 4 Equilibrium Analysis

In this section, SPNE of the game $\Gamma$ is evaluated in closed-form using backward induction principles. Given the large size of the tree, we present our analysis for every decision stage (i.e. Stages 8, 7, 6, 2 and 1 in the order of backward induction) individually in the following subsections.

### 4.1 Stage 8: Attacker's Best-Response Exfiltration Strategy

The first decision stage that manifests during the running of backward induction approach is to evaluate the attacker's best-response exfiltration strategy in Stage 8.

**Lemma 1.** *If the defender compromises ($c_i = 1$) and the attacker opts to decrypt the data ($e_i = 1$), the best response for the attacker on data exfiltration ($x_i$) is:*

$$x_i^*(8_{i,k,3}) = \begin{cases} 1, & \textit{if } \zeta \geq \zeta^*(8_{i,k,3}) \\ 0, & \textit{otherwise} \end{cases} \tag{6}$$

*where the threshold $\zeta^*(8_{i,k,3})$ is given by*

$$\zeta^*(8_{i,k,3}) = \frac{\Big(p_2\tilde{U}_{i,k,18} + (1-p_2)\tilde{U}_{i,k,19}\Big) - \Big(p_3\tilde{U}_{i,k,22} + (1-p_3)\tilde{U}_{i,k,23}\Big)}{\Big(p_3\tilde{U}_{i,k,20} + (1-p_3)\tilde{U}_{i,k,21}\Big) - \Big(p_3\tilde{U}_{i,k,22} + (1-p_3)\tilde{U}_{i,k,23}\Big)} \tag{7}$$

*Proof.* In Fig. 3, the node in Stage 8 with a history $c_i = 1$ and $e_i = 1$ is labeled as $8_{i,k,3}$. At this node, the attacker has to pick $x_i \in \{0, 1\}$ such that its expected utility is maximized.

The expected utility obtained by the attacker at node $8_{i,k,3}$ for choosing $x_i = 1$ and $x_i = 0$ are respectively given by

$$\tilde{U}(x_i = 1|8_{i,k,3}) = \zeta\Big(p_3\tilde{U}_{i,k,20} + (1-p_3)\tilde{U}_{i,k,21}\Big) + (1-\zeta)\Big(p_3\tilde{U}_{i,k,22} + (1-p_3)\tilde{U}_{i,k,23}\Big), \tag{8}$$

$$\text{and } \tilde{U}(x_i = 0|8_{i,k,3}) = p_2\tilde{U}_{i,k,18} + (1-p_2)\tilde{U}_{i,k,19}. \tag{9}$$

Note that $x_i = 1$ is the best response exfiltration strategy if $\tilde{U}(x_i = 1|8_{i,k,3}) \geq \tilde{U}(x_i = 0|8_{i,k,3})$, i.e.,

$$\zeta\Big(p_3\tilde{U}_{i,k,20} + (1-p_3)\tilde{U}_{i,k,21}\Big) + (1-\zeta)\Big(p_3\tilde{U}_{i,k,22} + (1-p_3)\tilde{U}_{i,k,23}\Big) > p_2\tilde{U}_{i,k,18} + (1-p_2)\tilde{U}_{i,k,19}. \tag{10}$$

The inclination for data exfiltration stems primarily from the attacker's perception of the exfiltrated data as a strategic asset. This strategic value lies in its potential to provide added leverage for subsequent attacks or negotiations. Additionally, a financial motive is present with the stolen information being seen as valuable on the illicit markets. Moreover, the decision to exfiltrate data may be driven by a lack of trust or opportunistic behavior of the attacker and using it as a means of insurance or an alternative revenue source post-ransom payment. Upon rearranging the terms, the attacker's best response to exfiltrate is to choose $x_i = 1$ if the network honeypot deceives the attacker with probability $\zeta \geq \zeta^*(8_{i,k,3})$, where $\zeta^*(8_{i,k,3})$ is defined in Eq. (7).

**Lemma 2.** *If the defender compromises and the attacker does not decrypt the data, the optimal strategy for the attacker on data exfiltration ($x_i$) is:*

$$x_i^*(8_{i,k,2}) = \begin{cases} 1, & \textit{if } \zeta \geq \zeta^*(8_{i,k,2}) \\ 0, & \textit{otherwise,} \end{cases} \tag{11}$$

*where the threshold $\zeta^*(8_{i,k,2})$ is given by*

$$\zeta^*(8_{i,k,2}) = \frac{\Big(p_0\tilde{U}_{i,k,12} + (1-p_0)\tilde{U}_{i,k,13}\Big) - \Big(p_1\tilde{U}_{i,k,16} + (1-p_1)\tilde{U}_{i,k,17}\Big)}{\Big(p_1\tilde{U}_{i,k,14} + (1-p_1)\tilde{U}_{i,k,15}\Big) - \Big(p_1\tilde{U}_{i,k,16} + (1-p_1)\tilde{U}_{i,k,17}\Big)} \tag{12}$$

*Proof.* In Fig. 3, the node in Stage 8 with a history $c_i = 1$ and $e_i = 0$ is labeled as $8_{i,k,2}$. At this node, the attacker has to pick $x_i \in \{0, 1\}$ such that its expected utility is maximized.

The expected utility obtained by the attacker at node $8_{i,k,2}$ for choosing $x_i = 1$ and $x_i = 0$ are respectively given by

$$\tilde{U}(x_i = 1|8_{i,k,2}) = \zeta\Big(p_1\tilde{U}_{i,k,14} + (1-p_1)\tilde{U}_{i,k,15}\Big) + (1-\zeta)\Big(p_1\tilde{U}_{i,k,16} + (1-p_1)\tilde{U}_{i,k,17}\Big) \tag{13}$$

$$\text{and } \tilde{U}(x_i = 0|8_{i,k,2}) = p_0\tilde{U}_{i,k,12} + (1-p_0)\tilde{U}_{i,k,13}. \tag{14}$$

Note that $x_i = 1$ is the best response exfiltration strategy if $\tilde{U}(x_i = 1|8_{i,k,2}) \geq \tilde{U}(x_i = 0|8_{i,k,2})$, i.e.

$$\zeta\Big(p_1\tilde{U}_{i,k,14} + (1-p_1)\tilde{U}_{i,k,15}\Big) + (1-\zeta)\Big(p_1\tilde{U}_{i,k,16} + (1-p_1)\tilde{U}_{i,k,17}\Big) > p_0\tilde{U}_{i,k,12} + (1-p_0)\tilde{U}_{i,k,13}. \tag{15}$$

The attacker's intent for exfiltration is similar to what is discussed in Lemma 1. Upon rearranging the terms, the attacker's best response to exfiltrate is to choose $x_i = 1$ if the network honeypot deceives the attacker with probability $\zeta \geq \zeta^*(8_{i,k,2})$, where $\zeta^*(8_{i,k,2})$ is defined in Eq. (12).

**Lemma 3.** *If the defender does not compromise ($c_i = 0$) and the attacker opts to decrypt the data ($e_i = 1$), the best response for the attacker on data exfiltration ($x_i$) is:*

$$x_i^*(8_{i,k,1}) = \begin{cases} 1, & \text{if } \zeta \geq \zeta^*(8_{i,k,1}) \\ 0, & \text{otherwise,} \end{cases} \tag{16}$$

*where the threshold $\zeta^*(8_{i,k,1})$ is given by*

$$\zeta^*(8_{i,k,1}) = \frac{\Big(p_2\tilde{U}_{i,k,6} + (1-p_2)\tilde{U}_{i,k,7}\Big) - \Big(p_3\tilde{U}_{i,k,10} + (1-p_3)\tilde{U}_{i,k,11}\Big)}{\Big(p_3\tilde{U}_{i,k,8} + (1-p_3)\tilde{U}_{i,k,9}\Big) - \Big(p_3\tilde{U}_{i,k,10} + (1-p_3)\tilde{U}_{i,k,11}\Big)} \tag{17}$$

*Proof.* In Fig. 3, the node in Stage 8 with a history $c_i = 0$ and $e_i = 1$ is labeled as $8_{i,k,1}$. At this node, the attacker has to pick $x_i \in \{0, 1\}$ such that its expected utility is maximized.

The expected utility obtained by the attacker at node $8_{i,k,1}$ for choosing $x_i = 1$ and $x_i = 0$ are respectively given by

$$\tilde{U}(x_i = 1|8_{i,k,1}) = \zeta\Big(p_3\tilde{U}_{i,k,8} + (1-p_3)\tilde{U}_{i,k,9}\Big) + (1-\zeta)\Big(p_3\tilde{U}_{i,k,10} + (1-p_3)\tilde{U}_{i,k,11}\Big), \tag{18}$$

$$\text{and } \tilde{U}(x_i = 0|8_{i,k,1}) = p_2\tilde{U}_{i,k,6} + (1-p_2)\tilde{U}_{i,k,7}. \tag{19}$$

Note that $x_i = 1$ is the best response exfiltration strategy if $\tilde{U}(x_i = 1|8_{i,k,1}) \geq \tilde{U}(x_i = 0|8_{i,k,1})$, i.e.

$$\zeta\Big(p_3\tilde{U}_{i,k,8} + (1-p_3)\tilde{U}_{i,k,9}\Big) + (1-\zeta)\Big(p_3\tilde{U}_{i,k,10} + (1-p_3)\tilde{U}_{i,k,11}\Big) > p_2\tilde{U}_{i,k,6} + (1-p_2)\tilde{U}_{i,k,7}. \tag{20}$$

The attacker's intent for exfiltration is similar to what is discussed in Lemma 1. Upon rearranging the terms, the attacker's best response to exfiltrate is to choose $x_i = 1$ if the network honeypot deceives the attacker with probability $\zeta \geq \zeta^*(8_{i,k,1})$, where $\zeta^*(8_{i,k,1})$ is defined in Eq. (17).

**Lemma 4.** *If the defender does not compromise ($c_i = 0$) and the attacker does not decrypt the data ($e_i = 0$), the best response for the attacker on data exfiltration ($x_i$) is:*

$$x_i^*(8_{i,k,0}) = \begin{cases} 1, & \textit{if } \zeta \geq \zeta^*(8_{i,k,0}) \\ 0, & \textit{otherwise,} \end{cases} \tag{21}$$

*where the threshold $\zeta^*(8_{i,k,0})$ is given by*

$$\zeta^*(8_{i,k,0}) = \frac{\left(p_0\tilde{U}_{i,k,0} + (1-p_0)\tilde{U}_{i,k,1}\right) - \left(p_1\tilde{U}_{i,k,4} + (1-p_1)\tilde{U}_{i,k,5}\right)}{\left(p_1\tilde{U}_{i,k,2} + (1-p_1)\tilde{U}_{i,k,3}\right) - \left(p_1\tilde{U}_{i,k,4} + (1-p_1)\tilde{U}_{i,k,5}\right)} \tag{22}$$

*Proof.* In Fig. 3, the node in Stage 8 with a history $c_i = 0$ and $e_i = 0$ is labeled as $8_{i,k,0}$. At this node, the attacker has to pick $x_i \in \{0, 1\}$ such that its expected utility is maximized.

The expected utility obtained by the attacker at node $8_{i,k,0}$ for choosing $x_i = 1$ and $x_i = 0$ are respectively given by

$$\begin{aligned} \tilde{U}(x_i = 1|8_{i,k,0}) = {} & \zeta\left(p_1\tilde{U}_{i,k,2} + (1-p_1)\tilde{U}_{i,k,3}\right) \\ & + (1-\zeta)\left(p_1\tilde{U}_{i,k,4} + (1-p_1)\tilde{U}_{i,k,5}\right), \end{aligned} \tag{23}$$

$$\text{and } \tilde{U}(x_i = 0|8_{i,k,0}) = p_0\tilde{U}_{i,k,0} + (1-p_0)\tilde{U}_{i,k,1}. \tag{24}$$

Note that $x_i = 1$ is the best response exfiltration strategy if $\tilde{U}(x_i = 1|8_{i,k,0}) \geq \tilde{U}(x_i = 0|8_{i,k,0})$, i.e.,

$$\begin{aligned} \zeta\left(p_1\tilde{U}_{i,k,2} + (1-p_1)\tilde{U}_{i,k,3}\right) + (1-\zeta)\left(p_1\tilde{U}_{i,k,4} + (1-p_1)\tilde{U}_{i,k,5}\right) \\ > p_0\tilde{U}_{i,k,0} + (1-p_0)\tilde{U}_{i,k,1}. \end{aligned} \tag{25}$$

The attacker's intent for exfiltration is similar to what is discussed in Lemma 1. Upon rearranging the terms, the attacker's best response to exfiltrate is to choose $x_i = 1$ if the network honeypot deceives the attacker with probability $\zeta \geq \zeta^*(8_{i,k,0})$, where $\zeta^*(8_{i,k,0})$ is defined in Eq. (22).

### 4.2 Stage 7: Attacker's Best-Response Decryption Strategy

Per the attacker's optimal decision on data exfiltration, the attacker's decision on whether to decrypt the data or not is given by the following lemmas.

**Lemma 5.** *If the defender compromises ($c_i = 1$), the best response for the attacker on data decryption ($e_i$) is:*

$$e_i^*\left(7_{i,k,1}\middle|x_i^*(8_{i,k,3}), x_i^*(8_{i,k,2})\right) = \begin{cases} 1, & \textit{if } \lambda(7_{i,k,1}) \geq 0, \\ 0, & \textit{otherwise,} \end{cases} \tag{26}$$

*where*

$$\begin{aligned} \lambda(7_{i,k,1}) = {} & x_i^*(8_{i,k,3})\left[\tilde{U}(x_i = 1|8_{i,k,3}) - \tilde{U}(x_i = 0|8_{i,k,3})\right] - x_i^*(8_{i,k,2})\left[\tilde{U}(x_i = 1|8_{i,k,2}) - \tilde{U}(x_i = 0|8_{i,k,2})\right] \\ & + \tilde{U}(x_i = 0|8_{i,k,3}) - \tilde{U}(x_i = 0|8_{i,k,2}) \end{aligned} \tag{27}$$

*Proof.* In Fig. 3, the node in Stage 7 with a history $c_i = 1$ is labeled as $7_{i,k,1}$. At this node, the attacker has to pick $e_i \in \{0, 1\}$ such that its expected utility is maximized.

The expected utility obtained by the attacker at node $7_{i,k,1}$ for choosing $e_i = 1$ and $e_i = 0$ are respectively given by

$$\tilde{U}(e_i = 1|7_{i,k,1}) = x_i^*(8_{i,k,3}) \cdot \tilde{U}(x_i = 1|8_{i,k,3}) + \Big(1 - x_i^*(8_{i,k,3})\Big) \cdot \tilde{U}(x_i = 0|8_{i,k,3}), \tag{28}$$

$$\text{and } \tilde{U}(e_i = 0|7_{i,k,1}) = x_i^*(8_{i,k,2}) \cdot \tilde{U}(x_i = 1|8_{i,k,2}) + \Big(1 - x_i^*(8_{i,k,2})\Big) \cdot \tilde{U}(x_i = 0|8_{i,k,2}). \tag{29}$$

The attacker's decision to decrypt the data can be attributed to several factors. A pivotal consideration is the establishment of trustworthiness as fulfilling the agreement enhances the attacker's reputation for reliability within the criminal landscape. Additionally, adhering to an implicit criminal code of conduct and seeking to avoid law enforcement attention provide strong motivations for the attacker to proceed with decryption. The strategic move of honoring the agreement not only fosters a perception of dependability but may also encourage future victims to comply with ransom demands.

Hence $e_i = 1$ is the best response decryption strategy if

$$\lambda(7_{i,k,1}) \triangleq \tilde{U}(e_i = 1|7_{i,k,1}) - \tilde{U}(e_i = 0|7_{i,k,1}) \geq 0.$$

On the contrary, $e_i = 0$ is the best response decryption strategy if $\lambda(7_{i,k,1}) < 0$.

**Lemma 6.** *If the defender does not compromise ($c_i = 0$), the best response for the attacker on data decryption ($e_i$) is:*

$$e_i^*\Big(7_{i,k,0}\Big|x_i^*(8_{i,k,1}), x_i^*(8_{i,k,0})\Big) = \begin{cases} 1, & \textit{if } \lambda(7_{i,k,0}) \geq 0, \\ 0, & \textit{otherwise,} \end{cases} \tag{30}$$

*where* $\lambda(7_{i,k,0}) = x_i^*(8_{i,k,1}) \cdot \tilde{U}(x_i = 1|8_{i,k,1}) + (1 - x_i^*(8_{i,k,1})) \cdot \tilde{U}(x_i = 0|8_{i,k,1}) - \Big(x_i^*(8_{i,k,0}) \cdot \tilde{U}(x_i = 1|8_{i,k,0}) + (1 - x_i^*(8_{i,k,0})) \cdot \tilde{U}(x_i = 0|8_{i,k,0})\Big)$

*Proof.* In Fig. 3, the node in Stage 7 with a history $c_i = 0$ is labeled as $7_{i,k,0}$. At this node, the attacker has to pick $e_i \in \{0, 1\}$ such that its expected utility is maximized.

The expected utility obtained by the attacker at node $7_{i,k,0}$ for choosing $e_i = 1$ and $e_i = 0$ are respectively given by

$$\tilde{U}(e_i = 1|7_{i,k,0}) = x_i^*(8_{i,k,1}) \cdot \tilde{U}(x_i = 1|8_{i,k,1}) + \Big(1 - x_i^*(8_{i,k,1})\Big) \cdot \tilde{U}(x_i = 0|8_{i,k,1}) \tag{31}$$

and

$$\tilde{U}(e_i = 0|7_{i,k,0}) = x_i^*(8_{i,k,0}) \cdot \tilde{U}(x_i = 1|8_{i,k,0}) + \Big(1 - x_i^*(8_{i,k,0})\Big) \cdot \tilde{U}(x_i = 0|8_{i,k,0}) \tag{32}$$

The attacker's decision for this lemma is similar to that in Lemma 5. Note that $e_i = 1$ is the best response decryption strategy if

$$\lambda(7_{i,k,0}) \triangleq \tilde{U}(e_i = 1|7_{i,k,0}) - \tilde{U}(e_i = 0|7_{i,k,0}) \geq 0.$$

On the contrary, $e_i = 0$ is the best response decryption strategy if $\lambda(7_{i,k,0}) < 0$.

### 4.3 Stage 6: The Defender's Decision-Making on Compromise

**Lemma 7.** *The defender's decision-making on compromise ($c_i$) is given by:*

$$c_i^*(6_{i,k}|\boldsymbol{e}_i^*, \boldsymbol{x}_i^*) = \begin{cases} 1, & \textit{if } \beta(\boldsymbol{e}_i^*, \boldsymbol{x}_i^*) > 0, \\ 0, & \textit{otherwise,} \end{cases} \tag{33}$$

*where $\beta(\boldsymbol{e}_i^*, \boldsymbol{x}_i^*)$ is given by*

$$\begin{aligned} \beta(\boldsymbol{e}_i^*, \boldsymbol{x}_i^*) &= e_i^*(7_{i,k,1})\Big[x_i^*(8_{i,k,3})U(9_{i,k,7}) + \big(1 - x_i^*(8_{i,k,3})\big)U(9_{i,k,6})\Big] \\ &+\Big(1 - e_i^*(7_{i,k,1})\Big)\Big[x_i^*(8_{i,k,2})U(9_{i,k,5}) + \big(1 - x_i^*(8_{i,k,2})\big)U(9_{i,k,4})\Big] \\ &-e_i^*(7_{i,k,0})\Big[x_i^*(8_{i,k,1})U(9_{i,k,3}) + \big(1 - x_i^*(8_{i,k,1})\big)U(9_{i,k,2})\Big] \\ &-\Big(1 - e_i^*(7_{i,k,0})\Big)\Big[x_i^*(8_{i,k,0})U(9_{i,k,1}) + \big(1 - x_i^*(8_{i,k,0})\big)U(9_{i,k,0})\Big] \end{aligned} \tag{34}$$

*Proof.* The defender chooses to either compromise ($c_i = 1$), or not pay the ransom ($c_i = 0$) such that the expected utility at node $6_{i,k}$ is maximized.

The defender's expected utility of choosing $c_i = 1$ and $c_i = 0$ at node $6_{i,k}$ is given by

$$\begin{aligned} U(c_i = 1|6_{i,k}) &= e_i^*(7_{i,k,1})\Big[x_i^*(8_{i,k,3})U(9_{i,k,7}) + \big(1 - x_i^*(8_{i,k,3})\big)U(9_{i,k,6})\Big] \\ &+\Big(1 - e_i^*(7_{i,k,1})\Big)\Big[x_i^*(8_{i,k,2})U(9_{i,k,5}) + \big(1 - x_i^*(8_{i,k,2})\big)U(9_{i,k,4})\Big] \end{aligned} \tag{35}$$

and

$$\begin{aligned} \tilde{U}(c_i = 0|6_{i,k}) &= e_i^*(7_{i,k,0})\Big[x_i^*(8_{i,k,1})U(9_{i,k,3}) + \big(1 - x_i^*(8_{i,k,1})\big)U(9_{i,k,2})\Big] \\ &+\Big(1 - e_i^*(7_{i,k,0})\Big)\Big[x_i^*(8_{i,k,0})U(9_{i,k,1}) + \big(1 - x_i^*(8_{i,k,0})\big)U(9_{i,k,0})\Big] \end{aligned} \tag{36}$$

respectively, where

$$\begin{aligned} U(9_{i,k,7}) &= \zeta\left[p_3 U_{i,k,20} + (1 - p_3)U_{i,k,21}\right] + (1 - \zeta)\left[p_3 U_{i,k,22} + (1 - p_3)U_{i,k,23}\right], \\ U(9_{i,k,6}) &= p_2 U_{i,k,18} + (1 - p_2)U_{i,k,19}, \\ U(9_{i,k,5}) &= \zeta\left[p_1 U_{i,k,14} + (1 - p_1)U_{i,k,15}\right] + (1 - \zeta)\left[p_1 U_{i,k,16} + (1 - p_1)U_{i,k,17}\right], \\ U(9_{i,k,4}) &= p_0 U_{i,k,12} + (1 - p_0)U_{i,k,13}, \\ U(9_{i,k,3}) &= \zeta\left[p_3 U_{i,k,8} + (1 - p_3)U_{i,k,9}\right] + (1 - \zeta)\left[p_3 U_{i,k,10} + (1 - p_3)U_{i,k,11}\right], \\ U(9_{i,k,2}) &= p_2 U_{i,k,6} + (1 - p_2)U_{i,k,7}, \\ U(9_{i,k,1}) &= \zeta\left[p_1 U_{i,k,2} + (1 - p_1)U_{i,k,3}\right] + (1 - \zeta)\left[p_1 U_{i,k,4} + (1 - p_1)U_{i,k,5}\right], \\ U(9_{i,k,0}) &= p_0 U_{i,k,0} + (1 - p_0)U_{i,k,1}. \end{aligned} \tag{37}$$

Defenders choose to pay a ransom when their data is encrypted due to practical considerations. If the encrypted data is crucial for business operations or contains sensitive information, the cost of downtime and potential damage to reputation becomes a driving factor. The complexity of decryption and the absence of reliable backups can limit options, making payment the quickest way to regain access. The defender will choose $c_i = 1$ if

$$\beta(\boldsymbol{e}_i^*, \boldsymbol{x}_i^*) \triangleq U(c_i = 1|6_{i,k}) - U(c_i = 0|6_{i,k}) \geq 0.$$

Otherwise, the best response strategy of the defender is $c_i = 0$.

### 4.4 Stage 2: Attacker's Best-Response Ransomware Development Strategy

**Lemma 8.** *The attacker's best response is to not develop the ransomware, i.e.* $d_i^*(2_{i,3}) = 0$*, when the defender backs up the data (i.e.* $b_i = 1$*) and uses deception (i.e.* $h_i = 1$*).*

*Proof.* In Fig. 2, the node in Stage 2 with a history $b_i = 1$ and $h_i = 1$ is labeled as $2_{i,3}$. At this node, the attacker has to pick $d_i \in \{0, 1\}$ such that its expected utility is maximized.

The expected utility obtained by the attacker at node $2_{i,3}$ for choosing $d_i = 1$ and $d_i = 0$ are respectively given by

$$\tilde{U}(d_i = 1|2_{i,3}) = -C_D, \quad \text{and} \quad \tilde{U}(d_i = 0|2_{i,3}) = 0. \tag{38}$$

When the defender has made backups, developing ransomware becomes pointless for the attacker. Backups allow the defender to quickly recover, eliminating the need to pay any ransom. This reduces the attacker's leverage and makes their efforts ineffective. Financial motivation for the attacker also decreases when there's a low chance of getting paid. Instead of gaining, the attacker incurs a loss in the cost of ransomware development. The defender's proactive approach with backups not only safeguards against data loss but also makes ransomware development an impractical and costly endeavor for the attacker.

Since $\tilde{U}(d_i = 0|2_{i,3}) > \tilde{U}(d_i = 1|2_{i,3})$, $d_i = 0$ is the best response ransomware development strategy.

**Lemma 9.** *Attacker's best response is to not develop the ransomware, i.e.* $d_i^*(2_{i,2}) = 0$*, when the defender backs up (i.e.* $b_i = 1$*) but does not use deception (i.e.* $h_i = 0$*).*

*Proof.* In Fig. 2, the node in Stage 2 with a history $b_i = 1$ and $h_i = 0$ is labeled as $2_{i,2}$. At this node, the attacker has to pick $d_i \in \{0, 1\}$ such that its expected utility is maximized.

The expected utility obtained by the attacker at node $2_{i,2}$ for choosing $d_i = 1$ and $d_i = 0$ are respectively given by

$$\tilde{U}(d_i = 1|2_{i,2}) = -C_D \quad \text{and} \quad \tilde{U}(d_i = 0|2_{i,2}) = 0 \tag{39}$$

With the reasons similar to Lemma 8, since $\tilde{U}(d_i = 0|2_{i,2}) > \tilde{U}(d_i = 1|2_{i,2})$, $d_i = 0$ is the best response ransomware development strategy.

**Lemma 10.** *Attacker's best response when defender does not back up (i.e.,*$b_i = 0$*) but uses deception (i.e.,*$h_i = 1$*) given by* $d_i$ *is:*

$$d_i^*(2_{i,1}) = \begin{cases} 1, & \text{\textit{if} } \alpha(\boldsymbol{c}_i^*) > 0, \\ 0, & \textit{otherwise} \end{cases} \tag{40}$$

*where*

$$\begin{aligned}\alpha(\boldsymbol{c}_i^*) = \rho\tau\Big[&\gamma\Big\{c_i^*(6_{i,1})\tilde{U}(7_i,1,0) + (1-c_i^*(6_{i,1}))\tilde{U}(7_i,1,1)\Big\}\\ &+(1-\gamma)\Big\{c_i^*(6_{i,2})\tilde{U}(7_i,2,0) + (1-c_i^*(6_{i,2}))\tilde{U}(7_i,2,1)\Big\}\Big]\\ +\ \rho(1-\tau)\Big[&\gamma\Big\{c_i^*(6_{i,3})\tilde{U}(7_i,3,0) + (1-c_i^*(6_{i,3}))\tilde{U}(7_i,3,1)\Big\}\\ &+(1-\gamma)*\Big\{c_i^*(6_{i,4})*\tilde{U}(7_i,4,0) + (1-c_i^*(6_{i,4}))*\tilde{U}(7_i,4,1)\Big\}\Big]\\ +\ (1-\rho)\tau\Big[&\gamma\Big\{c_i^*(6_{i,5})\tilde{U}(7_i,5,0) + (1-c_i^*(6_{i,5}))\tilde{U}(7_i,5,1)\Big\}\\ &+(1-\gamma)\Big\{c_i^*(6_{i,6})\tilde{U}(7_i,6,0) + (1-c_i^*(6_{i,6}))\tilde{U}(7_i,6,1)\Big\}\Big]\\ +\ (1-\rho)(1-\tau)\Big[&\gamma\Big\{c_i^*(6_{i,7})\tilde{U}(7_i,7,0) + (1-c_i^*(6_{i,7}))\tilde{U}(7_i,7,1)\Big\}\\ &+(1-\gamma)\Big\{c_i^*(6_{i,8})\tilde{U}(7_i,8,0) + (1-c_i^*(6_{i,8}))\tilde{U}(7_i,8,1)\Big\}\Big]\end{aligned}\tag{41}$$

*Proof.* In Fig. 2, the node in Stage 2 with a history $b_i = 0$ and $h_i = 1$ is labeled as $2_{i,1}$. At this node, the attacker has to pick $d_i \in \{0, 1\}$ such that its expected utility is maximized.

The expected utility obtained by the attacker at node $2_{i,1}$ for choosing $d_i = 1$ and $d_i = 0$ are respectively given by

$$\begin{aligned}\tilde{U}(d_i=1|2_{i,1}) = \rho\tau\Big[&\gamma\Big\{c_i^*(6_{i,1})\tilde{U}(7_i,1,0) + (1-c_i^*(6_{i,1}))\tilde{U}(7_i,1,1)\Big\}\\ &+(1-\gamma)\Big\{c_i^*(6_{i,2})\tilde{U}(7_i,2,0) + (1-c_i^*(6_{i,2}))\tilde{U}(7_i,2,1)\Big\}\Big]\\ +\rho(1-\tau)\Big[&\gamma\Big\{c_i^*(6_{i,3})\tilde{U}(7_i,3,0) + (1-c_i^*(6_{i,3}))\tilde{U}(7_i,3,1)\Big\}\\ &+(1-\gamma)*\Big\{c_i^*(6_{i,4})*\tilde{U}(7_i,4,0) + (1-c_i^*(6_{i,4}))*\tilde{U}(7_i,4,1)\Big\}\Big]\\ +(1-\rho)\tau\Big[&\gamma\Big\{c_i^*(6_{i,5})\tilde{U}(7_i,5,0) + (1-c_i^*(6_{i,5}))\tilde{U}(7_i,5,1)\Big\}\\ &+(1-\gamma)\Big\{c_i^*(6_{i,6})\tilde{U}(7_i,6,0) + (1-c_i^*(6_{i,6}))\tilde{U}(7_i,6,1)\Big\}\Big]\\ +(1-\rho)(1-\tau)\Big[&\gamma\Big\{c_i^*(6_{i,7})\tilde{U}(7_i,7,0) + (1-c_i^*(6_{i,7}))\tilde{U}(7_i,7,1)\Big\}\\ &+(1-\gamma)\Big\{c_i^*(6_{i,8})\tilde{U}(7_i,8,0) + (1-c_i^*(6_{i,8}))\tilde{U}(7_i,8,1)\Big\}\Big]\end{aligned}\tag{42}$$

$$\text{and } \tilde{U}(d_i=0|2_{i,1}) = 0.\tag{43}$$

The absence of data backups increases the attacker's leverage as valuable and critical data becomes vulnerable to encryption. This vulnerability persists even if the defender employs deception techniques to lure or mislead the attacker as the potential gains from exploiting the lack of data backups outweighs the risks associated with potential deception. This situation raises the likelihood of the defender paying the ransom to regain access to crucial information and amplifies the attack's impact by causing significant disruption to business operations. Financial motivation and the limited recovery options for the defender further incentivize the attacker to pursue ransomware development as an effective means of achieving their goals. Note that $d_i = 1$ is the best response ransomware development strategy if $\tilde{U}(d_i = 1|2_{i,1}) \geq \tilde{U}(d_i = 0|2_{i,1})$, i.e., $\alpha(\boldsymbol{c}_i^*) > 0$. On the contrary, $d_i = 0$ is the best response decryption strategy if $\alpha(\boldsymbol{c}_i^*) < 0$.

**Lemma 11.** *Attacker's best response when defender does not back up (i.e. $b_i = 0$) and does not use deception (i.e. $h_i = 0$) given by $d_i$ is:*

$$d_i^*(2_{i,0}) = \begin{cases} 1, & \textit{if } \tilde{U}(6_{i,0}) > 0,\\ 0, & \textit{otherwise}\end{cases}\tag{44}$$

*Proof.* In Fig. 2, the node in Stage 2 with a history $b_i = 0$ and $h_i = 1$ is labeled as $2_{i,0}$. At this node, the attacker has to pick $d_i \in \{0, 1\}$ such that its expected utility is maximized.

The expected utility obtained by the attacker at node $2_{i,0}$ for choosing $d_i = 1$ and $d_i = 0$ are respectively given by

$$\tilde{U}(d_i = 1 | 2_{i,0}) = \tilde{U}(6_{i,0}) \quad \text{and} \quad \tilde{U}(d_i = 0 | 2_{i,0}) = 0 \tag{45}$$

With reasons similar to Lemma 10, note that $d_i = 1$ is the best response ransomware development strategy if $\tilde{U}(d_i = 1 | 2_{i,0}) \geq \tilde{U}(d_i = 0 | 2_{i,0})$, i.e. $\tilde{U}(6_{i,0}) > 0$. On the contrary, $d_i = 0$ is the best response decryption strategy if $\tilde{U}(6_{i,0}) < 0$.

### 4.5 Stage 1: Defender's Decision-Making on Data Backup and Deception

The defender's decision-making on Stage 1 for the $i^{th}$ subsystem considers the product of the backup decision i.e. $b_i$ and deception decision i.e. $h_i$ making it a bilinear problem. For the $i^{th}$ subsystem, the defender's utility is given by:

$$U_i(1) = \Big[(1 - b_i) \cdot (1 - h_i) \cdot U(2_{i,0}) + (1 - b_i) \cdot (h_i) \cdot U(2_{i,1}) + (b_i) \cdot (1 - h_i) \cdot U(2_{i,2}) + b_i \cdot h_i \cdot U(2_{i,3})\Big] \tag{46}$$

Combining the problem across all the $N$ subsystems, the defender wishes to

$$\max_{(b_1,h_1),\cdots,(b_N,h_N)} \sum_{i=1}^{N} \Big[\ (1 - b_i)(1 - h_i)U(2_{i,0}) + (1 - b_i)h_i U(2_{i,1}) + b_i(1 - h_i)U(2_{i,2}) + b_i h_i U(2_{i,3})\ \Big] \tag{47}$$

In vector notation, let $y = [b_1, h_1, \cdots, b_i, h_i, \cdots, b_N, h_N]^T$ denote the decision variable at node 1. Then, the aforementioned optimization problem can be rewritten as the following mixed-integer linear program

$$\max_{y} \ y^T \Theta y + \theta^T y + \delta, \tag{48}$$

where

$$\Theta = \begin{bmatrix} \Theta_1 & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & \Theta_N \end{bmatrix} \quad \text{is a block-diagonal matrix with}$$

$$\Theta_i = \begin{bmatrix} 0 & 0 \\ U(2_{i,0}) - U(2_{i,1}) - U(2_{i,2}) + U(2_{i,3}) & 0 \end{bmatrix}, \tag{49}$$

$$\theta = \begin{bmatrix} U(2_{1,2}) - U(2_{1,0}) \\ U(2_{1,1}) - U(2_{1,0}) \\ \vdots \\ U(2_{N,2}) - U(2_{N,0}) \\ U(2_{N,1}) - U(2_{N,0}) \end{bmatrix}, \quad \text{and} \quad \delta = \sum_{i=1}^{N} U(2_{i,0}).$$

Since the above problem is a binary quadratic program, the solution cannot be evaluated in closed-form (a finite number of operations, using a given set of functions and mathematical operations). Instead, the problem has to be computationally solved using a standard integer-programming algorithm to maximize and find the defender's best response for this stage.

## 5 Ransomware Defense in Healthcare: A Case Study

Ransomware attacks affect the healthcare industry leading to the slowing of critical processes to make them completely inoperable and making important information inaccessible [7]. It is important to analyze our work with respect to the behavior in the real world where ransomware has been highly rampant in the healthcare industry. However, it is difficult to get a well-formed dataset that provides information on different parameters such as ransomware amount demanded, value of data for the victim, cost of ransomware development, compromised amount, value of data privacy (breach of data), etc. To accurately portray this information we had to collect information from a variety of credible sources including news, blogs, and statistics. Our parameter settings for the experiment and numerical analysis thus include information collected from these sources and is depicted in the Table 3. On average the ransomware amount demanded by the cybercriminals in a healthcare sector compromise ranges from $0.25 million to $5 million with a mean of $2.63 million whereas a single attack can cost a healthcare provider about $112 million [10]. The average cost of a healthcare data breach has risen to $10.93 million [5] leading to data privacy compromise. In [9], it is discussed that one healthcare sector invested around $8 million towards cybersecurity. We set the parameters for cyber deception using the honeypot, honeytoken, honeyfiles based on their costs from discussed literature from Sect. 2 whereas the numbers are derived by considering $1 million of the overall cybersecurity investment towards cyber deception. The loss of the attacker after getting caught in fact can be considered in terms of the amount being recovered, and jail time, among others. For the experiment, it is considered to be twice as much of the ransomware amount requested as it is difficult to quantify aspects other than the amount recovered. For cybercrimes such as phishing and ransomware, only 5% of cybercriminals are apprehended for their crimes [4]. This goes to show how difficult it is to apprehend cyber criminals. By using deception techniques, we intend to engage attackers with the targeted systems enough to buy time for cybercriminals to be tracked. The probability of

**Table 3.** Parameters Settings in the Experiment for Healthcare Industry Ransomware Breach.

| Fig. | T | $C_D$ | F | $C_x$ | $V_i$ | $V_p$ | $R_i$ | $V_i'$ | $C_{ht}$ | $n_{ht}$ | $C_{hp}$ | $C_{hf}$ | $n_{hf}$ | $C_{nh}$ | $p_0$ | $p_1$ | $p_2$ | $p_3$ |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 4 | 3 m | – | – | 1000 | 112 m | 10.93 m | 2.63 m | 4.38 m | 0.05 | 0.67 m | 0.08 m | 0.08 m | 0.5 m | 0.1 m | 0.1 | 0.15 | 0.18 | 0.2 |
| 5 | 3 m | – | 5.25 m | 1000 | 112 m | 10.93 m | 2.63 m | 4.38 m | 0.05 | 0.67 m | – | 0.08 m | 0.5 m | 0.1 m | 0.1 | 0.15 | 0.18 | 0.2 |
| 6 | 3 m | 0.1 m | – | – | 112 m | 10.93 m | 2.63 m | 4.38 m | 0.05 | 0.67 m | 0.08 m | 0.08 m | 0.5 m | 0.1 m | 0.1 | 0.15 | 0.18 | 0.2 |
| 7 | 3 m | – | 5.25 m | 1000 | 112 m | 10.93 m | 2.63 m | 4.38 m | 0.05 | 0.67 m | – | 0.08 m | 0.5 m | – | 0.1 | 0.15 | 0.18 | 0.2 |
| 8 | 3 m | 0.1 m | 5.25 m | 1000 | 112 m | 10.93 m | 2.63 m | – | 0.05 | 0.67 m | – | 0.08 m | 0.5 m | 0.1 m | 0.1 | 0.15 | 0.18 | 0.2 |
| 9 | 3 m | 0.1 m | 5.25 m | 1000 | 112 m | 10.93 m | 2.63 m | – | 0.05 | 0.67 m | 0.08 m | – | 0.5 m | 0.1 m | 0.1 | 0.15 | 0.18 | 0.2 |

getting apprehended by law enforcers should rise given the deception methods are designed to slow down the attackers and/or stop them from attacking. We define the probability values based on this observation.

## 6 Numerical Results and Discussion

### 6.1 Game Experiment Result

The computation experiments are carried out on a single Intel Xeon CPU operating at 2.20 GHz equipped with 12 GB of RAM and a Tesla K80 accelerator. All the required programs for the experiments were developed in Python and executed on this specific configuration. The experiment codebase can be accessed from the GitHub Repository [2].

The observed trend in Fig. 4, where the attacker utility initially increases with the cost of ransomware development $C_D$ and then starts to decrease is attributed to the dynamic interplay between the increasing sophistication of the ransomware and the defender's strategic response. The observed relationship between $C_D$ and attacker utility is further nuanced by the loss of the attacker for being arrested $F$. A lower value of $F$ encourages a more risk-tolerant approach, contributing to the initial increase in attacker utility, while higher values of $F$ may prompt risk-averse behavior, leading to a subsequent decrease. As $C_D$ rises, the attacker may invest in more advanced ransomware, making it initially more potent and profitable. However, the defender's investment in deception techniques to bolster their cybersecurity defenses creates a threshold beyond which further increases in $C_D$ yield diminishing returns for the attacker.

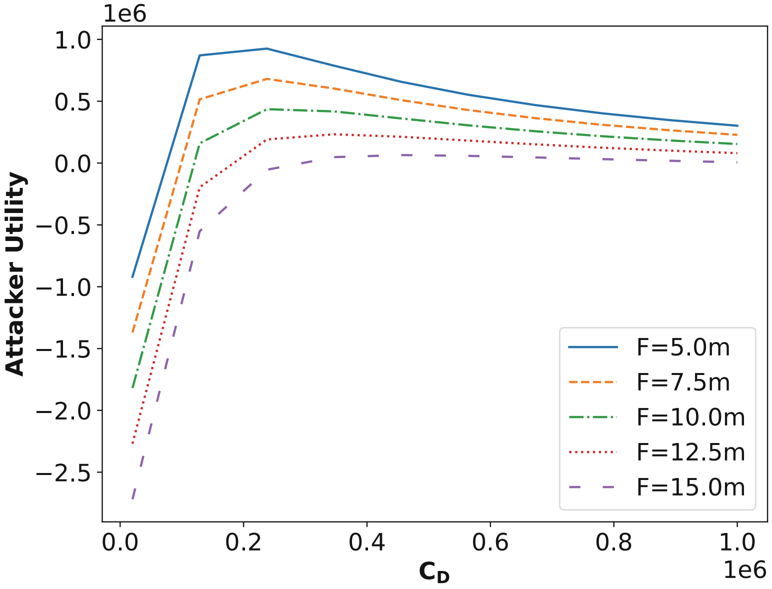

**Fig. 4.** Impact of attacker loss $F$ and cost of ransomware development $C_D$ on attacker utility.

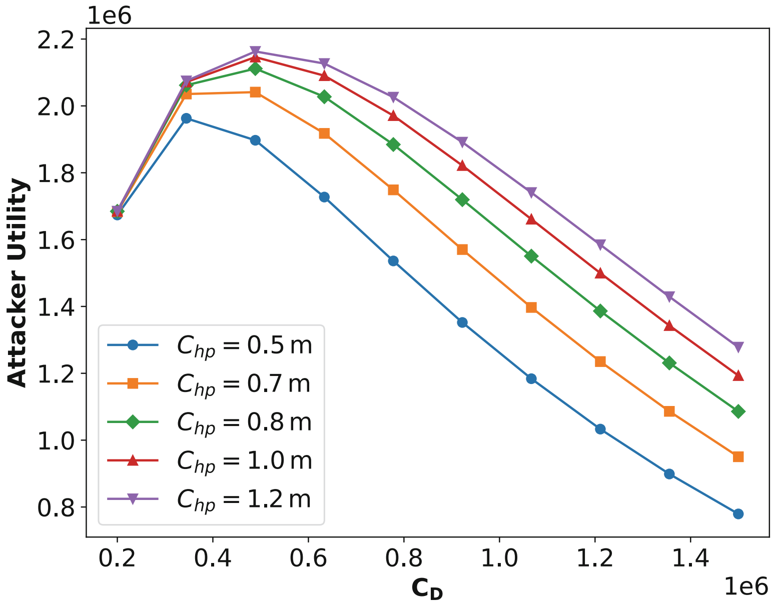

**Fig. 5.** Impact of cost of honeypot $C_{hp}$ and cost of ransomware development $C_D$ on attacker utility.

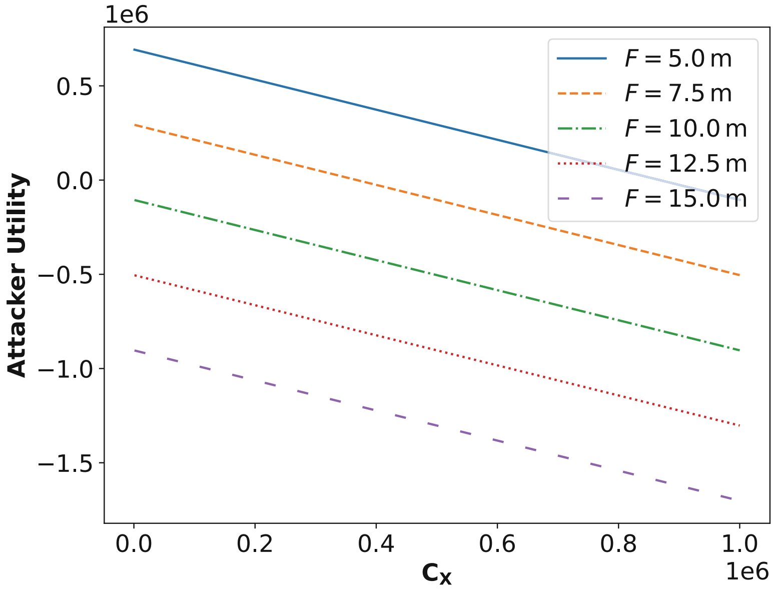

**Fig. 6.** Impact of cost of exfiltration $C_x$ and attacker loss $F$ on attacker utility.

In Fig. 5, the observed trend in the attacker's utility as $C_D$ increases aligns with the anticipation of deception techniques. Until reaching a threshold, the attacker finds it increasingly cost-effective to develop ransomware. However, beyond this threshold, there is a diminishing return which leads to a linear decline in utility. The effectiveness of this strategy is influenced by $C_{hp}$ with lower values making ransomware attacks more attractive early on, followed by a linear decrease in utility. This suggests that attackers strategically adapt to the evolving security landscape weighing the cost and benefits of different attack methods.

In Fig. 6, the linear decrease in attacker utility as $C_x$ increases indicates a direct and proportional relationship between $C_x$ and $F$. This trend suggests that the economic burden on the attacker rises linearly with the cost of exfiltration. The observed behavior is indicative of a deterrent effect, wherein higher exfiltration costs discourage attackers due to the linear impact on their overall utility.

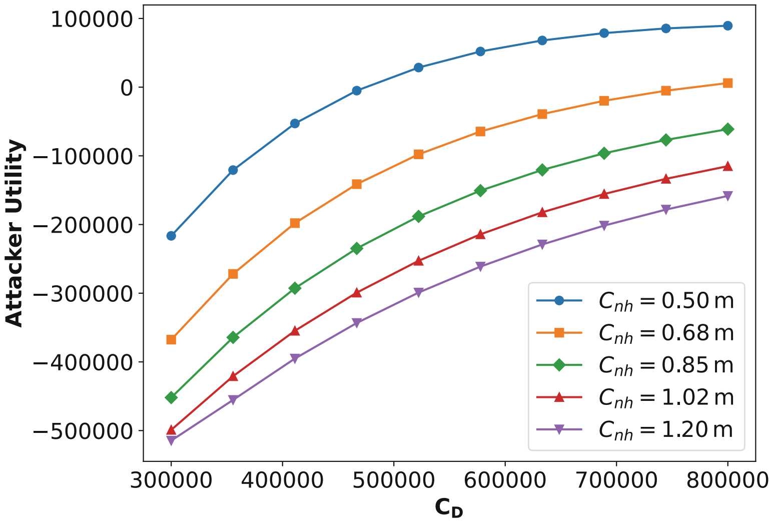

**Fig. 7.** Impact of cost of n/w honeypot $C_{nh}$ and cost of ransomware development $C_D$ on attacker utility.

In Fig. 7, the observed logarithmic increase in attacker utility with varying $C_D$ for different values of $C_{nh}$ suggests that the impact of increasing countermeasures may have diminishing returns for the attacker. At lower $C_{nh}$, the attacker finds it more profitable to invest in ransomware development which leads to a sharper increase in utility. However, for higher $C_{nh}$, the incremental gain in attacker utility diminishes. This reflects a balance between the defender's countermeasures and the attacker's risk tolerance.

The logarithmic curves observed in Fig. 8 and Fig. 9 indicate diminishing returns on defensive investments for different values of $V_i$. Both figures show that higher values of $V_i$ lead to greater utility for the defender. Notably, the utility is consistently higher for the Cost of Network Honeypot Deployment $C_{nh}$ compared to $C_{hp}$ for equivalent $V_i$ and investment in deception techniques. This implies that network honeypots are more effective in enhancing the defender's utility as they provide a stronger deterrence against attackers in the current game framework. The exponential relationship highlights the importance of optimizing resource allocation in ransomware attacks.

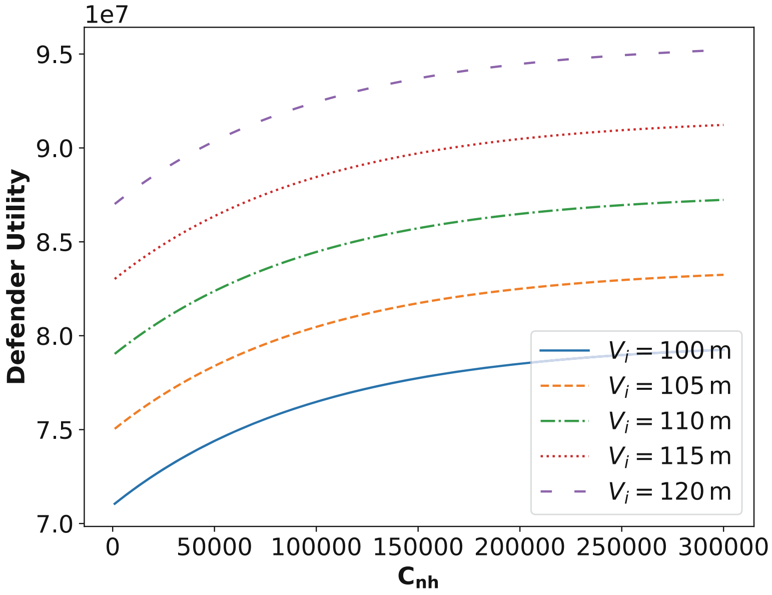

**Fig. 8.** Impact of cost of n/w honeypot $C_{nh}$ and defender's value of data $V_i$ on defender utility.

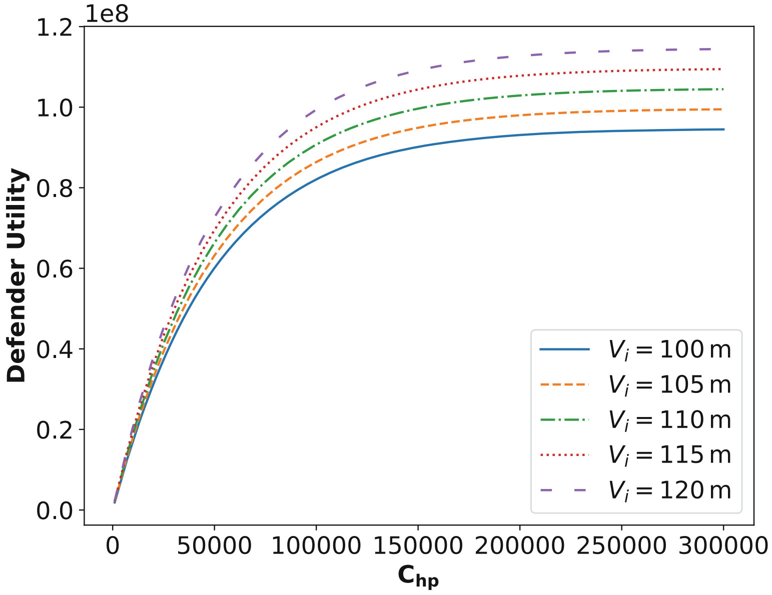

**Fig. 9.** Impact of cost of honeypot $C_{hp}$ and defender's value of data $V_i$ on defender utility.

**Table 4.** Parameters settings for comparing the game outcomes with and without the deception.

| Fig. | T | $C_D$ | F | $C_x$ | $V_i$ | $V_p$ | $R_i$ | $V_i'$ | $C_{ht}$ | $n_{ht}$ | $C_{hp}$ | $C_{hf}$ | $n_{hf}$ | $C_{nh}$ | $p_0$ | $p_1$ | $p_2$ | $p_3$ |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 10 | 1 m | 0.1 m | – | 10000 | 112 m | 10.93 m | – | 4.38 m | 0.05 | 0.67 m | 0.08 m | 0.08 m | 0.5 m | 0.08 m | 0.06 | 0.15 | 0.18 | 0.2 |

### 6.2 Comparison with Ransomware Game Model in the Literature

We compare our numerical results and game performance to the one presented in [42] to show how use of cyber deception achieved by using deception techniques discussed in the paper can be beneficial. The compared paper presents a theoretical approach that uses parameters that are simulated. We considered various common parameters in the two different approaches to show the impact of deception. The parameter settings for the comparison are presented in Table 4. Looking at Fig. 10, a notable trend emerges in the attacker's utility during comparison where $A$ is the utility from the compared paper, while $B$ is from the current game context. The attacker's utility exhibits a slower rate of increase shown by dotted line that is attributed to heightened uncertainties stemming from deceptive elements. This discrepancy in the rate of increase is particularly evident as we plot against $R$. This observation leads us to conclude that the incorporation of deception techniques serves as an effective measure in shaping the strategic landscape of the adversarial game, offering the defender a valuable tool to influence and mitigate the attacker's utility growth over different scenarios.

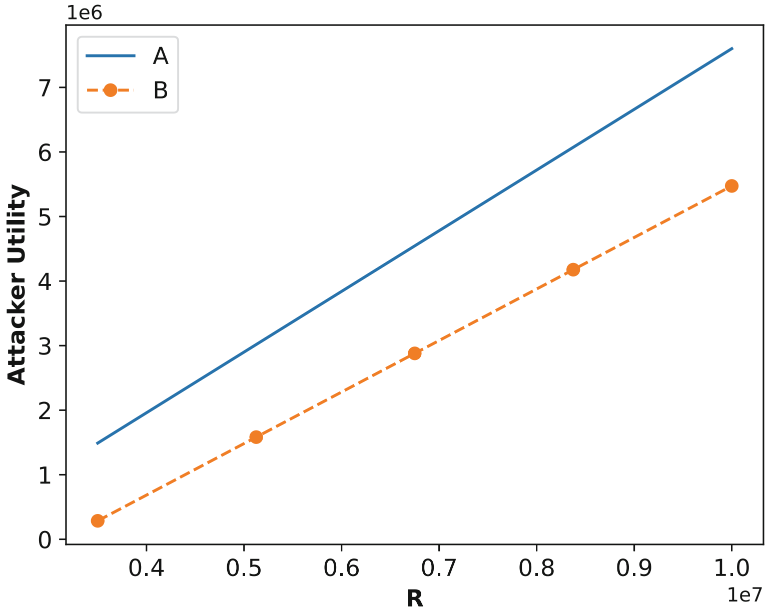

**Fig. 10.** Comparison of attacker utilities for ransomware amount $R$ in A [42] and B (this work).

## 7 Conclusion and Future Work

In this paper, we presented a multi-stage ransomware game considering state-of-the-art deception strategies in the form of honey-x deployable to different stages of the ransomware attack. We evaluated Subgame-Perfect Nash Equilibrium (SPNE) for the game in closed form using backward induction principles and standard integer programming. We performed a numerical analysis of the developed game to evaluate the strategies given a realistic game model using real-world data and statistics relating to the healthcare industry. It is seen that the use of deception technologies is favorable to the defender towards thwarting cyber-criminals with higher chances of getting caught. Our findings pave the way for future research and practical applications to strengthen the resilience of critical systems against ransomware threats. This work portrays a complete game as a baseline which can be extended to an exploration of a non-deterministic/incomplete game model for the given problem. Furthermore, it can be extended to experimentation under consideration of more than one subsystem as discussed in the game model.

**Acknowledgement.** This material is based upon work supported by the National Science Foundation under award number CNS-2243619, and the National Security Agency under award number H98230-21-1-0260. Any opinions, findings, and conclusions or recommendations expressed in this publication are those of the author(s) and do not necessarily reflect the views of the National Science Foundation or the National Security Agency.

## References

1. Canarytokens. https://canarytokens.org/generate. Accessed 16 Dec 2023
2. Deception-based Ransomware Defense. https://github.com/bhusalb/gt-ransomware-simulation. Accessed 20 May 2024
3. FBI Internet Crime Report 2022. https://www.ic3.gov/Media/PDF/AnnualReport/2022_IC3Report.pdf. Accessed 20 May 2023
4. How Do Hackers Get Caught and Exposed?. https://www.metacompliance.com/blog/phishing-and-ransomware/how-do-hackers-normally-get-caught. Accessed 20 Jan 2024
5. IBM: Average Cost of a Healthcare Data Breach Increases to Almost $11 Million. https://www.hipaajournal.com/2023-cost-healthcare-data-breach/. Accessed 20 Jan 2024
6. Kippo. https://github.com/desaster/kippo. Accessed 16 Dec 2023
7. Ransomware: In the Healthcare Sector. https://www.cisecurity.org/insights/blog/ransomware-in-the-healthcare-sector. Accessed 20 Jan 2024
8. Spacesiren: A honeytoken manager. https://github.com/spacesiren/spacesiren. Accessed 16 Dec 2023
9. The Cost of Cybersecurity in Healthcare. https://www.cdw.com/content/cdw/en/articles/security/the-cost-of-cybersecurity-in-healthcare.html. Accessed 20 Jan 2024
10. The Latest 2023 Ransomware Statistics (2024). https://aag-it.com/the-latest-ransomware-statistics/. Accessed 20 Jan 2024
11. Aumann, R.J.: Backward induction and common knowledge of rationality. Games Econom. Behav. **8**(1), 6–19 (1995)
12. Bercovitch, M., Renford, M., Hasson, L., Shabtai, A., Rokach, L., Elovici, Y.: HoneyGen: an automated honeytokens generator. In: Proceedings of 2011 IEEE

International Conference on Intelligence and Security Informatics, pp. 131–136. IEEE (2011)
13. Cartwright, A., Cartwright, E.: The economics of ransomware attacks on integrated supply chain networks. Digit. Threats: Res. Pract. (2023)
14. Cartwright, E., Hernandez Castro, J., Cartwright, A.: To pay or not: game theoretic models of ransomware. J. Cybersecur. **5**(1), tyz009 (2019)
15. Dameff, C., et al.: Ransomware attack associated with disruptions at adjacent emergency departments in the us. JAMA Netw. Open **6**(5), e2312270–e2312270 (2023)
16. Feng, Y., Liu, C., Liu, B.: Poster: a new approach to detecting ransomware with deception. In: 38th IEEE symposium on security and privacy (2017)
17. Ganfure, G.O., Wu, C.F., Chang, Y.H., Shih, W.K.: RTrap: trapping and containing ransomware with machine learning. IEEE Trans. Inf. Forensics Secur. **18**, 1433–1448 (2023)
18. Gómez-Hernández, J.A., Álvarez-González, L., García-Teodoro, P.: R-locker: thwarting ransomware action through a honeyfile-based approach. Comput. Secur. **73**, 389–398 (2018)
19. Keijzer, N.: The new generation of ransomware: an in depth study of Ransomware-as-a-service. Master's thesis, University of Twente (2020)
20. Kolodenker, E., Koch, W., Stringhini, G., Egele, M.: PayBreak: defense against cryptographic ransomware. In: Proceedings of the 2017 ACM on Asia Conference on Computer and Communications Security, pp. 599–611 (2017)
21. Lapan, H.E., Sandler, T.: To bargain or not to bargain: that is the question. Am. Econ. Rev. **78**(2), 16–21 (1988)
22. Li, Z., Liao, Q.: Game theory of data-selling ransomware. J. Cyber Secur. Mob. 65–96 (2021)
23. Liu, S., Chen, X.: Mitigating data exfiltration ransomware through advanced decoy file strategies (2023)
24. Min, D., Ko, Y., Walker, R., Lee, J., Kim, Y.: A content-based ransomware detection and backup solid-state drive for ransomware defense. IEEE Trans. Comput. Aided Des. Integr. Circuits Syst. **41**(7), 2038–2051 (2021)
25. Mokube, I., Adams, M.: Honeypots: concepts, approaches, and challenges. In: Proceedings of the 45th Annual Southeast Regional Conference, pp. 321–326 (2007)
26. Mphago, B., Bagwasi, O., Phofuetsile, B., Hlomani, H.: Deception in dynamic web application honeypots: case of Glastopf. In: Proceedings of the International Conference on Security and Management (SAM). p. 104. The Steering Committee of The World Congress in Computer Science, Computer ... (2015)
27. Müter, M., Freiling, F., Holz, T., Matthews, J.: A generic toolkit for converting web applications into high-interaction honeypots. Univ. Mannheim **280**, 6–1 (2008)
28. Patyal, M., Sampalli, S., Ye, Q., Rahman, M.: Multi-layered defense architecture against ransomware. Int. J. Bus. Cyber Secur. **1**(2) (2017)
29. Qin, X., Jiang, F., Cen, M., Doss, R.: Hybrid cyber defense strategies using honey-x: a survey. Comput. Netw. 109776 (2023)
30. Reeder, J.R., Hall, C.T.: Cybersecurity's pearl harbor moment: lessons learned from the colonial pipeline ransomware attack (2021)
31. Săndescu, C., Rughiniş, R., Grigorescu, O.: HUNT: using honeytokens to understand and influence the execution of an attack. eLearn. Softw. Educ. **1** (2017)
32. Selten, R., Selten, R.: A Simple Game Model of Kidnapping. Springer, Heidelberg (1988)

33. Shaukat, S.K., Ribeiro, V.J.: RansomWall: a layered defense system against cryptographic ransomware attacks using machine learning. In: 2018 10th International Conference on Communication Systems & Networks (COMSNETS), pp. 356–363. IEEE (2018)
34. Sheen, S., Asmitha, K., Venkatesan, S.: R-sentry: deception based ransomware detection using file access patterns. Comput. Electr. Eng. **103**, 108346 (2022)
35. Spitzner, L.: Honeypots: Tracking Hackers, vol. 1. Addison-Wesley Reading (2003)
36. Subedi, K.P., Budhathoki, D.R., Chen, B., Dasgupta, D.: RDS3: ransomware defense strategy by using stealthily spare space. In: 2017 IEEE Symposium Series on Computational Intelligence (SSCI), pp. 1–8. IEEE (2017)
37. Tandon, A., Nayyar, A.: A comprehensive survey on ransomware attack: a growing havoc cyberthreat. In: Data Management, Analytics and Innovation: Proceedings of ICDMAI 2018, vol. 2, pp. 403–420 (2019)
38. Wang, Z., Wu, X., Liu, C., Liu, Q., Zhang, J.: RansomTracer: exploiting cyber deception for ransomware tracing. In: 2018 IEEE Third International Conference on Data Science in Cyberspace (DSC), pp. 227–234. IEEE (2018)
39. Wilson, D., Avery, J.: Mitigating data exfiltration in storage-as-a-service clouds. arXiv preprint arXiv:1606.08378 (2016)
40. Yin, T., Sarabi, A., Liu, M.: Deterrence, backup, or insurance: a game-theoretic analysis of ransomware. In: The Annual Workshop on the Economics of Information Security (WEIS) (2021)
41. Yuill, J., Zappe, M., Denning, D., Feer, F.: HoneyFiles: deceptive files for intrusion detection. In: 2004 Proceedings from the Fifth Annual IEEE SMC Information Assurance Workshop, pp. 116–122. IEEE (2004)
42. Zhang, C., Luo, F., Ranzi, G.: Multistage game theoretical approach for ransomware attack and defense. IEEE Trans. Serv. Comput. (2022)
43. Zhao, Y., Ge, Y., Zhu, Q.: Combating ransomware in internet of things: a games-in-games approach for cross-layer cyber defense and security investment. In: Bošanský, B., Gonzalez, C., Rass, S., Sinha, A. (eds.) GameSec 2021. LNCS, vol. 13061, pp. 208–228. Springer, Cham (2021). https://doi.org/10.1007/978-3-030-90370-1_12