Auditing a Cloud Provider's Compliance with Data Backup Requirements: A Game Theoretical Analysis

Ziad Ismail, Christophe Kiennert, Jean Leneutre, and Lin Chen

***Abstract*—The new developments in cloud computing have introduced significant security challenges to guarantee the confidentiality, integrity, and availability of outsourced data. A Service Level Agreement (SLA) is usually signed between the cloud provider and the customer. For redundancy purposes, it is important to verify the cloud provider's compliance with data backup requirements in the SLA. There exists a number of security mechanisms to check the integrity and availability of outsourced data. This task can be performed by the customer or be delegated to an independent entity that we will refer to as the verifier. However, checking the availability of data introduces extra costs, which can discourage the customer of performing data verification too often. The interaction between the verifier and the cloud provider can be captured using game theory in order to find an optimal data verification strategy. In this paper, we formulate this problem as a two player non-cooperative game. We consider the case in which each type of data is replicated a number of times which can depend on a set of parameters including, among others, its size and sensitivity. We analyze the strategies of the cloud provider and the verifier at the Nash Equilibrium and derive the expected behavior of both players. Finally, we validate our model numerically on a case study and explain how we evaluate the parameters in the model.**

***Index Terms*—Cloud storage, SLA compliance, data replication auditing, game theory**

I. Introduction

Cloud computing is a model for enabling ubiquitous, convenient, on-demand network access to a shared pool of configurable computing resources (e.g., networks, servers, storage, applications, and services) that can be rapidly provisioned and released with minimal management effort or service provider interaction [1]. However, all the benefits brought by the Cloud, such as lower costs and ease of use, come with a tradeoff. In particular, users have to entrust their data to a cloud provider (CP), which can be viewed as a selfish entity aimed at maximizing profits. This could lead the CP to act in ways that are detrimental to users' interests. The new security issues introduced by cloud computing need to be addressed and are of interest to both industry and academia [2].

One aspect of cloud computing is the ability to buy or lease storage capacity, which introduces security problems related to data integrity and availability. The client often lacks full control over the manner his data is stored, entailing difficulties in ensuring that data stored in the Cloud are indeed left intact. A number of guarantees are given through the *Service Level Agreement* (SLA) which is a contract between the CP and the client that defines the expected level of the service offered by the CP. This includes in particular the overall availability rate, i.e. the expected downtime per year. In addition, an SLA can include other features such as the number of data backups, which may be physically stored at different geographical locations. However, in a worst-case scenario, a CP may not respect the requirements of the backup process of some of the entrusted data to save both money and storage space capacities. By behaving this way, the CP may not directly cause data losses for the client (as the original copy can be left intact), but raises the probability of accidental data loss happening (related to hazards), which impacts the overall data availability rate.

The client may be interested in checking the availability of all data backups using specific protocols such as proofs of data retrievability widely studied in the literature [3]. In these works, efforts have been made to design solutions that meet various requirements such as low time complexity, stateless verification, unbounded use of queries, and retrievability of data, etc. In particular, several protocols allow public verifiability from a Third Party Auditor (TPA), to which the client can delegate the verification task through an *Audit Level Agreement*. This assumption is more realistic, since in most cases, a lack of resources or expertise will prevent the client from personally performing these verifications. In this paper, we will consider that the TPA, which is an independent entity, will be the verifier of the client's data in the CP systems.

In spite of the numerous features of the verification schemes, choosing the efficient set of features to use remains a challenging task. For example, it would be a waste of both time and resources for the verifier to check the client's data all the time in the case of an honest CP. On the other hand, it would be risky if the data is not checked regularly when the CP is acting dishonestly. Therefore, in order to analyze the interactions between the CP and the verifier, and derive their expected behaviors to find the optimal verification strategy for the verifier, we model the data availability verification problem as a two player non-cooperative static game featuring the cloud provider and the TPA. We introduce a number of extensions to the basic model in [4] to take into account more realistic scenarios. In particular, we consider a model featuring multiple copies of each data stored by the CP and analyze the behavior

This research was supported in part by the joint research laboratory SEIDO (SEcurité et Internet Des Objets) between EDF R&D and Télécom ParisTech.

Z. Ismail and J. Leneutre are with the Department of Computer Science and Networks, LTCI, CNRS, Télécom ParisTech, Université Paris-Saclay, 75013, Paris, France (E-mails: {ziad, leneutre}@telecom-paristech.fr).

C. Kiennert is with the Department of Networks and Telecommunication Services, SAMOVAR, CNRS UMR 5157, Télécom SudParis, 9 Rue Charles Fourier, 91011 Evry, France (E-mail: christophe.kiennert@telecom-sudparis.eu).

L. Chen is with the Department of Computer Science and Networks, LRI (Laboratoire de Recherche en Informatique), University of Paris-Sud 11, 15 Rue Georges Clemenceau, 91400 Orsay, France (E-mail: lin.chen@lri.fr).

of both players under different types of strategies.

The remainder of the paper is organized as follows. In Section II, we describe the technical background and related work. In the next sections, we study the Nash equilibrium (NE) of the cloud storage game model while considering the existence of multiple copies of each data on the cloud provider's servers. In Section III, we analyze two types of one-shot games related to the dependency of players' strategies on a certain data on other data. Section IV presents a second formulation of the problem as a stackelberg game in which we have a leader and a follower in the game. Section V provides numerical results validating our analysis. Section VI explains how to evaluate the parameters in the model in a practical scenario, and illustrates it with a numerical example. Finally, we conclude the paper in Section VII.

II. Related Work

In untrusted cloud storage, it is important to verify the cloud provider's compliance with the security requirements in the SLA. For example, Popa et al. [5] designed a proof-based system to enable security guarantees in an SLA. In recent years, a significant amount of data integrity schemes were proposed by different researchers, and have been gradually adapted to specific use cases such as outsourced databases and cloud computing, for which works focusing on public verifiability issues, such as [6], were noticeably helpful and allowed clients to delegate the verification process to third parties. Among these schemes, the two main directions explored by researchers include the Provable Data Possession (PDP) for ensuring possession of data, and the Proof of Retrievability (POR) for data possession and retrievability. The main idea of PDP is that a data owner generates some metadata information for a data file to be used later for verification purposes. Many extensions of this scheme managed to decrease the communication cost and complexity [7], as well as to allow dynamic operations on data such as insertion, modification, or deletion [8]. Moreover, [9] and [10] proposed PDP schemes specific to cloud computing.

The POR scheme is considered as a complementary approach to PDP. [11] was among the first papers to consider formal models for POR schemes. In this scheme, disguised blocks (called sentinels) are embedded into the data before outsourcing. The verifier checks randomly picked sentinels, which would be influenced with a certain probability if the data is corrupted. An improved version of the POR approach was achieved with compact proofs of retrievability [12], with the design of a stateless protocol with unbounded audit interactions. [3] gives a detailed survey of the contributions of numerous extensions of the PDP and POR schemes. However, the schemes presented so far focus primarily on a single copy of a data file. Other schemes, such as [13], allow the verifier to check multiple copies of a data file on multiple cloud servers.

In the cloud domain, game theory has emerged in recent years as an important tool to analyze the interactions between multiple players with the same or conflicting interests. It has been used to study a number of problems including resource allocation and management [14] and cloud service negotiation [15], while some research papers addressed the problem of cloud security [16] [17]. To address cloud integrity issues, the authors in [16] proposed a model in which a client checks the correctness of calculations made on the data by the CP. In [17], Nix et al. study the case of querying one cloud provider, since checking data at multiple CPs is prohibitively expensive. [16] and [17] focused on checking whether the queries sent to the CP are being computed correctly, under the condition that the stored data is intact. On a side note, they did not mention which type of verification protocol (deterministic or probabilistic) they used. In addition to cloud-related problems, game theory has been used in multiple domains including network security [18] [19], intrusion detection [20], Botnet defense [21], among others. The work presented in this paper extends our previous work in [4].

III. Untrusted Cloud Storage Game with Multiple Data Copies

We consider a client outsourcing a set $\mathcal{D} = \{D_1, D_2, ..., D_N\}$ of N data to a cloud provider (CP). We consider the case in which the client delegates the data availability verification process to a Third Party Auditor (TPA).

We model the data availability problem as a non-cooperative static game with two players, the cloud provider and the TPA. We assume that both players are rational. The CP tries to gain storage space by not backing up correctly the client's data without being caught. On the other hand, the objective of the TPA is to distribute verification resources in order to detect partially of fully unbacked up data. Using the model defined in our previous work in [4] as a basis, we introduce an important extension related to the existence of multiple copies of the same data on the CP servers. This assumption has many important implications, as the CP has the possibility to dishonor his backup commitments in a more stealthy way without compromising the original version of the data. On the other hand, the verifier (TPA) will need to improve his verification strategy to check not only the existence of a data file, but the existence of the required number of backups to that file as well. This new extension to the model will allow us to analyze the behavior of both players from which we derive the optimal verification strategy. This scenario is closer to what we might expect to have in a real world setting. Although this game features interactions between only two players, several users may delegate the verification process to a same TPA. On the other hand, the case of a TPA verifying multiple cloud providers can be regarded as independent occurrences of the two-player game, except in the case where the cloud providers cooperate against the TPA, which leads to an entirely different scenario. Therefore, the proposed model covers a wide range of realistic situations, and can be applied to the case of multiple independent users relying on the same TPA, as well as to the case of auditing multiple independent cloud providers.

The reputation of a cloud provider will rely, among other factors, on the availability rate of clients' data. This can be achieved by keeping a number of copies of the data. This number can be a function of the importance of the data to the client in addition to its size. The data can be kept on the same server or distributed on multiple geographically dispersed servers. This will offer a higher availability rate and improve the resiliency against attacks, accidents, and hazards targeting specific locations. In this section, we extend the previous model in [4] with the assumption of the existence of multiple copies of the data on the CP servers. In this case, with the absence of a verification mechanism, the CP can remove the additional copies of a data file without impacting the client's access to that file. However, the CP takes a risk in the case if the remaining copy of the data became unavailable for some reason. The CP will try to weigh the risk of behaving in a malicious way with the possible benefit of increasing the storage space, which translates in practice to additional profits.

We associate to each data D_i the following parameters: the financial storage cost $S^i \geq 0$ of one copy of data D_i by the CP, which is proportional to data D_i's size; the financial value $F^i \geq 0$ of one copy of D_i quantifying how critical data D_i is to the client. The cost of processing the verification query for the TPA and the cost of executing the verification query by the cloud provider are supposed to be proportional to S^i and are given by C^tS^i and C^sS^i respectively, where $0 \leq C^s, C^t \leq 1$. In addition, let R^i refer to the number of backup copies of data $D_i \in \mathcal{D}$ that needs to be stored on the CP servers and ϵF^i the reward (e.g. in reputation) the CP gets if he acts honestly or passes the verification test undetected by the TPA otherwise, where $\epsilon > 0$.

In this paper, we make the following assumptions:

Assumption 1. *The costs related to network communications, both on the CP side and between the CP and TPA, are ignored.*

The model presented in this paper aims only at analyzing whether the CP will behave honestly or dishonestly, the possible storage flaws of an honest CP are out of scope of this work. Therefore, we make the following assumption:

Assumption 2. *The way the backup copies of the data are stored on the cloud provider servers is not taken into account.*

Assumption 3. *The probability of data corruption remaining undetected by the TPA after a check is neglected, even when using a probabilistic protocol.*

The approximation in Assumption 3 is justified by the fact that a dishonest CP will be likely to entirely omit one or more copies of the data, rather than keep parts of the copies stored on his servers. Nevertheless, such scenario was already taken into account in one of our models presented in [4]. While it is possible to take into account such scenario in the present work, the authors believe it will impact the clarity of the presentation of the model and increase the complexity of already complex equations.

TABLE I: Cloud Storage Game with Deterministic Verification for Data D_i

TPA CP	Check	Not check
Replicate	ϵF^i , $-C^tS^i - C^sS^i$	0 , 0
Not replicate	$-C^sS^i - S^i$, $-C^tS^i + F^i$	S^i , $-F^i$

Table I presents the payoffs for both players for data D_i in the case where $R^i = 1$. In this case, if the corrupted or unavailable data D_i is not checked, the CP gains a payoff S^i proportional to the size of data D_i while the TPA loses F^i. In addition to the cost of processing the verification query C^tS^i, we consider that the TPA should pay the cost of executing the verification query C^sS^i when he decides to verify D_i in the case where the CP respects the backup process of the data. However, when the CP chooses not to respect the backup process on D_i and the TPA chooses to verify, the TPA will gain F^i while paying for the verification cost C^tS^i, and the CP will lose S^i while paying for the cost of executing the verification query C^sS^i. Finally, neither players will achieve anything when the TPA decides not to verify D_i and the CP respects the backup process. In this paper, we focus on the number of backup copies of data D_i that can be checked by the TPA. We assume that an original version of the data is present on the CP servers and that version will not be targeted by the CP. Therefore, the TPA will be interested in verifying that the required number of backup copies R^i for each type of data D_i agreed on between the CP and the client is indeed present on the CP servers.

Let $\mathbb{1}$ represent the indicator function. We refer by p_0^m the probability that the CP respects the requirements of the backup process for data D_m. $\forall 1 \leq i \leq R^m$, let p_i^m denote the probability that the CP does not keep i copies of data D_m. Similarly for the verifier, we refer by q_0^m the probability that the TPA does not check the existence of any copy of data D_m, and $\forall 1 \leq j \leq R^m$, q_j^m the probability that the TPA verifies the existence of j copies of data D_m.

The utilities U_A and U_D of the cloud provider and the TPA are given in Equations 1 and 2 respectively.

The actions of both the CP and the TPA will determine their utilities. The actions of the CP that result in him getting a positive payoff are limited to the case where the number of copies j that has been checked by the verifier is less than the number of copies that remain on the CP servers after the CP has kept $R^m - i$ copies. In this case, the CP benefits

$$U_A(p,q) = \sum_{m=1}^{N} \Big\{ -\sum_{i=1}^{R^m}\sum_{j=1}^{R^m} p_i^m q_j^m (iS^m + jC^sS^m)\mathbb{1}_{i>R^m-j} + \sum_{i=0}^{R^m}\sum_{j=1}^{R^m} \epsilon p_i^m q_j^m (jF^m)\mathbb{1}_{i\leq R^m-j} + \sum_{i=1}^{R^m}\sum_{j=0}^{R^m} p_i^m q_j^m (iS^m)\mathbb{1}_{i\leq R^m-j} \Big\} \quad (1)$$

$$U_D(p,q) = \sum_{m=1}^{N} \Big\{ \sum_{i=1}^{R^m}\sum_{j=1}^{R^m} p_i^m q_j^m (iF^m)\mathbb{1}_{i>R^m-j} - \sum_{i=0}^{R^m}\sum_{j=1}^{R^m} p_i^m q_j^m (jC^sS^m)\mathbb{1}_{i\leq R^m-j} - \sum_{j=1}^{R^m} q_j^m C^tS^m j - \sum_{i=1}^{R^m}\sum_{j=0}^{R^m} p_i^m q_j^m (iF^m)\mathbb{1}_{i\leq R^m-j} \Big\} \quad (2)$$

from the value he gets from the additional storage space that has been freed up in addition to a reward for passing the verification test. The TPA, on the other hand, gets a negative payoff that includes the importance of the i copies that were not kept by the CP and that were undetected and the cost of processing the verification query. Otherwise ($i > R^m - j$), the CP gets a negative payoff that includes the cost of executing the verification query in addition to the value of the storage space that needs to be reallocated to the client's data. In this case, the TPA gets a positive payoff related to the importance of the copies of the data that were not kept by the CP and whose absence was detected by the TPA.

In this section, we investigate the case where both the CP and the TPA take their decisions at the same time while taking into account each other's strategies. This type of interactions falls under the one-shot game category [22]. In addition, we analyze the behavior of the CP and the TPA in two different game settings. In the first case, we suppose that the strategy of each player for a data item D_i is independent from the other data items D_j. In the second case, this condition is relaxed and we suppose that the strategies for data items D_i are interdependent.

A. Independent Strategies Game

We define an independent strategies game as follows:

Definition 1. *An Independent Strategies (IS) game is a game in which each player's strategy for each data D_i do not depend on other data D_j, $\forall j \neq i$.*

In this case, we have $\sum\limits_{i=0}^{R^m} p_i^m = 1$ and $\sum\limits_{j=0}^{R^m} q_j^m = 1$, $\forall m \in \{1, ..., N\}$ where m refers to data D_m.

Let $\theta_i^m = 2iF^m + (R^m - i)C^s S^m$ and $\phi_i^m = 2(R^m - i)S^m + i(C^s S^m + \epsilon F^m)$.

Theorem 1. *The NE of the IS game for the CP and the TPA is expressed as follows, $\forall m \in \{1, ..., N\}$:*

$$\begin{cases} p_0^{m*} = \dfrac{\dfrac{2R^m F^m - C^t S^m}{2R^m F^m + C^s S^m} - C^t S^m \sum\limits_{i=1}^{R^m - 1} \dfrac{1}{\theta_i^m} \prod\limits_{j=1}^{i-1} \left(1 + \dfrac{C^s S^m}{\theta_j^m}\right)}{1 + C^s S^m \sum\limits_{i=1}^{R^m - 1} \dfrac{1}{\theta_i^m} \prod\limits_{j=1}^{i-1} \left(1 + \dfrac{C^s S^m}{\theta_j^m}\right)} \\ p_i^{m*} = \dfrac{C^s S^m (p_0^m)^* + C^t S^m}{\theta_i^m} \prod\limits_{j=1}^{i-1} \left(1 + \dfrac{C^s S^m}{\theta_j^m}\right) \\ \qquad\qquad\qquad\qquad \forall i \in \{1, ..., R^m - 1\} \\ p_{R^m}^{m*} = \dfrac{C^t S^m + C^s S^m}{2R^m F^m + C^s S^m} \end{cases}$$

$$\begin{cases} q_0^{m*} = \dfrac{1 - \dfrac{S^m}{2S^m + \phi_{R^m}^m} + S^m \sum\limits_{i=1}^{R^m - 1} \dfrac{1}{\phi_i^m} \prod\limits_{j=1}^{i-1} \left(1 + \dfrac{2S^m}{\phi_j^m}\right)}{1 + 2S^m \sum\limits_{i=1}^{R^m - 1} \dfrac{1}{\phi_i^m} \prod\limits_{j=1}^{i-1} \left(1 + \dfrac{2S^m}{\phi_j^m}\right)} \\ q_i^{m*} = \dfrac{(2q_0^{m*} - 1)S^m}{\phi_i^m} \prod\limits_{j=1}^{i-1} \left(1 + \dfrac{2S^m}{\phi_j^m}\right) \ \forall i \in \{1, ..., R^m - 1\} \\ q_{R^m}^{m*} = \dfrac{S^m}{2S^m + \phi_{R^m}^m} \end{cases}$$

Proof. Refer to Appendix A. □

We can notice that $q_0^{m*} > 0.5$, $\forall m \in \{1, ..., N\}$. This result can be interpreted as the following. When the verifier wants to decide whether to check the existence of a data D_i, he has a choice between performing the verification of a number i of copies of the data or dropping his request. When the TPA prefers not to check over performing any checking ($q_0^m \geq \sum\limits_{j=1}^{R^m} q_j^m$), he allocates nevertheless some resources to execute verification queries. This will ensure that the CP will operate at the NE, and therefore cannot improve his utility by changing his strategy unilaterally.

With respect to the CP's strategy at the NE, we have the following Lemma:

Lemma 1. *In the case of an IS game, $\exists!\ x_0^m = F^m/S^m > 0$ s.t. $p_0^{m*}(x_0^m) = 0$.*

Proof. Refer to Appendix A. □

As a consequence of Lemma 1, the condition for the existence of the NE in this case is that $F^m/S^m > x_0^m$, $\forall m \in \{1, ..., N\}$. If that condition is not respected for data D_m, the CP is better off deleting at least one copy of the data. However, the TPA will respond by verifying the existence of the maximum number of copies as required in the backup process agreed on between the CP and the client. In this case, the TPA will make sure that he will always catch a dishonest CP and gets rewarded for his actions. Unfortunately, this scenario does not allow the emergence of a NE.

B. Correlated Strategies Game

In this case, the players' choices for their strategies for data D_i depend on their strategies for data D_j, $\forall j \neq i$. There are two possible scenarios. In the first scenario, we limit the actions of each player on one data item at each instance of the game. For example, at a given moment, the verifier will issue a query to verify only the existence of backups for data D_i. However, this scenario is limiting in practice, as sometimes it is more beneficial for the TPA to issue queries to verify the existence of backups for different types of data at once. In this case, we consider that at each instance of the game, each player can execute an action on each type of available data. For example, the CP can dishonor his backup commitments on a set of data items at once. Nevertheless, in the remaining of this section, we analyze the behavior of the CP and the TPA in both scenarios.

1) Single Targets:

We define a correlated strategies single targets game as follows:

Definition 2. *A Correlated Strategies Single Targets (CSST) game is a game in which each player can target one type of data and execute one action related to that data at each instance of the game.*

In practice, this translates to having $\sum\limits_{m=1}^{N} \sum\limits_{i=0}^{R^m} p_i^m = 1$ and $\sum\limits_{m=1}^{N} \sum\limits_{j=0}^{R^m} q_j^m = 1$. In this case, $\sum\limits_{i=0}^{R^m} p_i^m$ and $\sum\limits_{j=0}^{R^m} q_j^m$ refer to the probability of targeting data D_m for the CP and the TPA respectively.

Let parameters ψ_i^m, ω^m, τ^m, α^m, β^m, γ^m, δ^m, and η^m be defined as in Appendix B.

Theorem 2. *The NE of the CSST game for the CP and the TPA is expressed as follows, $\forall m \in \{1, ..., N\}$:*

$$\begin{cases} p_0^{m*} = \dfrac{\alpha^m}{\beta^m} \\ p_i^{m*} = \dfrac{C^s\alpha^m S^m + C^t\beta^m S^m}{\beta^m\theta_i^m}\psi_i^m \quad \forall i \in \{1, ..., R^m - 1\} \\ p_{R^m}^{m*} = \dfrac{(C^s\alpha^m S^m + C^t\beta^m S^m)(R^m - 2\omega^m)}{2\beta^m R^m F^m} \end{cases}$$

$$\begin{cases} q_0^{m*} = \dfrac{\eta^m}{\sum\limits_{i\in\mathcal{D}} \eta^i\big(1 + \delta^i + \gamma^i\phi_1^i \sum\limits_{j=1}^{R^i-1} \frac{1}{\phi_j^i}\big)} \\ q_i^{m*} = \dfrac{\gamma_m\eta^m\phi_1^m}{\phi_i^m \sum\limits_{i\in\mathcal{D}} \eta^i\big(1 + \delta^i + \gamma^i\phi_1^i \sum\limits_{j=1}^{R^i-1} \frac{1}{\phi_j^i}\big)} \\ \qquad\qquad \forall i \in \{1, ..., R^m - 1\} \\ q_{R^m}^{m*} = \dfrac{\delta_m\eta^m}{\sum\limits_{i\in\mathcal{D}} \eta^i\big(1 + \delta^i + \gamma^i\phi_1^i \sum\limits_{j=1}^{R^i-1} \frac{1}{\phi_j^i}\big)} \end{cases}$$

Proof. The result is found using a similar analysis as in the proof of Theorem 1. □

Lemma 2. *In the case of a CSST game, $\exists!\ S_1^m, S_2^m > 0$ s.t. $\forall S^m \in [S_1^m; S_2^m]$, we have $p_0^{m*} \in [0; 1]$.*

Proof. Refer to Appendix A. □

Given the result of Lemma 2, a necessary condition for the existence of the NE of the game in this case is that we have $S^m \in [S_1^m; S_2^m]$, $\forall m \in \{1, ..., N\}$ where S_1^m and S_2^m are the solutions of equations $p_0^{m*}(S_1^m) = 1$ and $p_0^{m*}(S_2^m) = 0$ respectively.

Lemma 3. *In the case of a CSST game, a necessary condition for the existence of a NE is that:*

$$\max_{i\in[\![1;N]\!]}(S^iR^i) < \frac{N + \dfrac{C^s}{C^t}}{\sum\limits_{m\in\mathcal{D}}\Big(\dfrac{1}{S^mR^m} + \dfrac{C^s}{R^m}\big(\tau^m + \dfrac{R^m - 2\omega^m}{2R^mF^m}\big)\Big)}$$

Proof. Follows directly from Lemma 2. □

2) Multiple Targets:

We define a correlated strategies multiple targets game as follows:

Definition 3. *A Correlated Strategies Multiple Targets (CSMT) game is a game in which each player can target multiple types of data at each instance of the game.*

In addition, in this case, we consider that the resources available to each player are limited. Therefore, we have $\sum\limits_{m=1}^{N}\sum\limits_{i=1}^{R^m} p_i^m = P$ and $\sum\limits_{m=1}^{N}\sum\limits_{j=1}^{R^m} q_j^m = Q$, where P and Q represent the resource constraints of the CP and the TPA respectively. We also have $\sum\limits_{i=0}^{R^m} p_i^m = 1$ and $\sum\limits_{j=0}^{R^m} q_j^m = 1$, $\forall m \in \{1, ..., N\}$.

Given the limited resources for the CP and the TPA, we can predict that they may be interested to take actions on a subset of the data stored on the CP servers. Let $\mathcal{T}_S$ denote such subset which we find using Algorithm 1. Let parameters E^m, G^m, H^m, W^m, ν and κ be defined as in Appendix B. We have the following theorem:

Algorithm 1: FindSensibleDataSet

```
Require: The set of data items $\mathcal{D}$
Output: The sensible target set $\mathcal{T}_S$
begin
    $S^{i'} \longleftarrow SortInDescendingOrder\Big(\dfrac{S^{\sigma(i)}(\phi_1^{\sigma(i)} + R^{\sigma(i)}C^sS^{\sigma(i)} + \epsilon R^{\sigma(i)}F^{\sigma(i)})}{\phi_1^{\sigma(i)} - 2S^{\sigma(i)}} \sum_{j=2}^{R^{\sigma(i)}-1} \frac{\phi_1^{\sigma(i)}}{\phi_j^{\sigma(i)}}\Big)$
    initialization: $n_S \longleftarrow N$
    while $n_S \geq 1$ do
        $z \longleftarrow \dfrac{n_S - Q - \sum_{i=1}^{n_S} \dfrac{(1 - W^i)(2S^i + \phi_{R^i}^i) - S^i}{H^i(2S^i + \phi_{R^i}^i)}}{\sum_{i=1}^{n_S} \dfrac{1}{H^i(2S^i + \phi_{R^i}^i)}}$
        if $(S^{n_S'} \leq z)$ then
            $n_S \longleftarrow n_S - 1$
        else
            break
        end
    end
    $\mathcal{T}_S = \{\sigma(i) \in \mathcal{D} : i \in [\![1, n_S]\!]\}$
end
```

Theorem 3. *If $\max\limits_{D_m\in\mathcal{D}\backslash\mathcal{T}_S} S^mR^m < \kappa$, the NE of the CSMT game for the CP and the TPA is expressed as follows, $\forall m \in \mathcal{T}_S$:*

$$\begin{cases} p_i^{m*} = \dfrac{-\nu C^sS^m + C^sS^mG^m + C^tS^mE^m}{E^m\theta_i^m}\psi_i^m \\ \qquad\qquad \forall i \in \{1, ..., R^m - 1\} \\ p_{R^m}^{m*} = 1 - C^tS^m\tau^m - (1 + C^sS^m\tau^m)(\dfrac{-\nu + G^m}{E^m}) \end{cases}$$

$$\begin{cases} q_i^{m*} = \dfrac{\phi_1^mS^m}{\phi_i^m(\phi_1^m - 2S^m)}\Big(\dfrac{2\kappa - 2S^m}{H^m(2S^m + \phi_{R^m}^m)} \\ \qquad + \dfrac{2(1 - W^m)}{H^m} - 1\Big) \quad \forall i \in \{1, ..., R^m - 1\} \\ q_{R^m}^{m*} = \dfrac{S^m - \kappa}{2S^m + \phi_{R^m}^m} \end{cases}$$

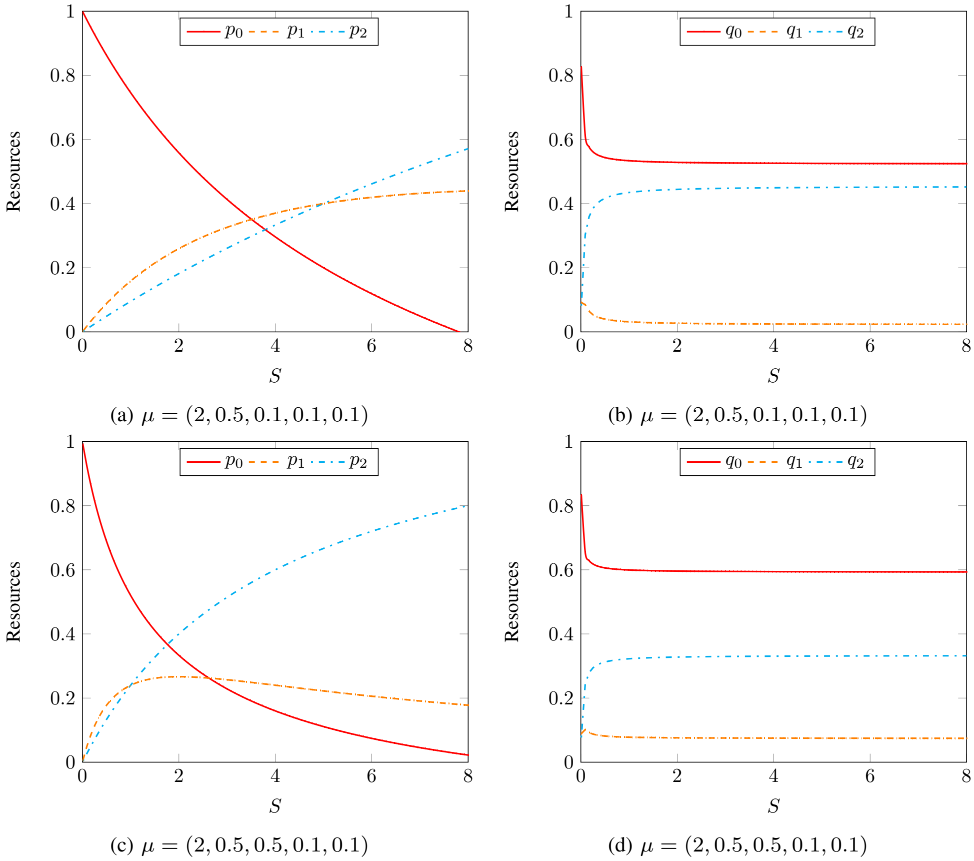
Proof. Refer to Appendix A. □

An immediate consequence of Theorem 3 is that the CP has no incentive to dishonor the agreement with the client for any data $D_j \in \mathcal{D}\backslash\mathcal{T}_S$ under the condition that $\max\limits_{D_m\in\mathcal{D}\backslash\mathcal{T}_S} S^mR^m < \kappa$.

IV. Untrusted Cloud Storage Stackelberg Game

In this section, we consider multiple backup copies for each data and analyze the case where the TPA will choose his strategy first. Then, the CP, informed by the TPA's choice, chooses his strategy. This type of interactions between the two players falls under the Stackelberg game category [22].

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TIFS.2016.2549002, IEEE Transactions on Information Forensics and Security



(a) $\mu = (2, 0.5, 0.1, 0.1, 0.1)$

(b) $\mu = (2, 0.5, 0.1, 0.1, 0.1)$

(c) $\mu = (2, 0.5, 0.5, 0.1, 0.1)$

(d) $\mu = (2, 0.5, 0.5, 0.1, 0.1)$

Fig. 1: IS game

In this type of games, we have a leader and a follower. The objective of the leader in the game is to anticipate the follower's response and to choose his strategy accordingly.

We will the study the interactions between the CP and the TPA when the choice of a strategy for data D_i is independent of the strategy for data D_j, $\forall j \neq i$.

The utility of the CP can be written as follows:

$$\begin{aligned} U_A(p,q) = & \sum_{m=1}^{N} \sum_{i=1}^{R^m} p_i^m \Big(- \sum_{j=1}^{R^m} q_j^m (iS^m + jC^s S^m) \mathbb{1}_{i > R^m - j} \\ & + \epsilon F^m \sum_{j=1}^{R^m} q_j^m (j) \mathbb{1}_{i \leq R^m - j} + \sum_{j=0}^{R^m} q_j^m (iS^m) \mathbb{1}_{i \leq R^m - j} \Big) \\ & + \epsilon \sum_{m=1}^{N} p_0^m \sum_{j=1}^{R^m} q_j^m (jF^m) \end{aligned}$$

The TPA will try to choose a strategy that will deter the CP from dishonoring his backup commitments on any data D_m. This translates to having $p_i^m = 0$, $\forall m \in \{1, ..., N\}$ $\forall i \in \{1, ..., R^m\}$.

In this case, analyzing the CP's utility function, we should have:

$$\begin{aligned} - \sum_{j=1}^{R^m} q_j^m (iS^m + jC^s S^m) \mathbb{1}_{i > R^m - j} + \epsilon F^m \sum_{j=1}^{R^m} q_j^m (j) \mathbb{1}_{i \leq R^m - j} \\ + \sum_{j=0}^{R^m} q_j^m (iS^m) \mathbb{1}_{i \leq R^m - j} \leq 0 \quad (3) \end{aligned}$$

In fact, we can relax the inequality to only require that Equation 3 equals 0. Therefore, the NE for the TPA and the CP can be expressed as follows, $\forall m \in \{1, ..., N\}$:

$$\begin{cases} p_0^{m*} = 1 \\ p_i^{m*} = 0 \quad \forall i \in \{1, ..., R^m\} \end{cases}$$

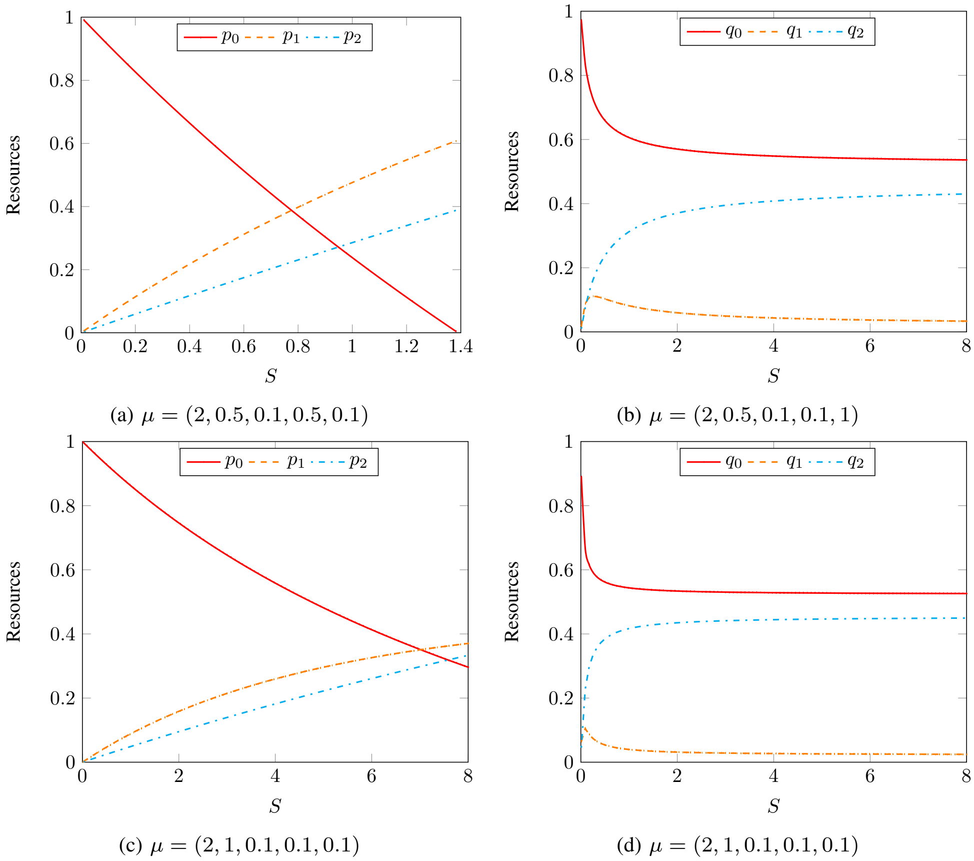
$$\begin{cases} q_0^{m*} = \dfrac{1}{2} \\ + \dfrac{C^s R^m}{4 + 2C^s R^m + 4 \sum_{i=1}^{R^m - 1} \frac{(C^s R^m S^m + 2S^m + i\epsilon F^m)}{\phi_i^m} \prod_{j=1}^{i-1} \big(1 + \frac{2S^m}{\phi_j^m}\big)} \\ q_i^{m*} = \dfrac{(2q_0^{m*} - 1)S^m}{\phi_i^m} \displaystyle\prod_{j=1}^{i-1} \big(1 + \dfrac{2S^m}{\phi_j^m}\big) \ \forall i \in \{1, ..., R^m - 1\} \\ q_{R^m}^{m*} = \dfrac{1}{1 + C^s R^m} \Big(q_0^{m*} \\ \qquad + \displaystyle\sum_{i=1}^{R^m - 1} \frac{(2q_0^{m*} - 1)(S^m + i\epsilon F^m)}{\phi_i^m} \prod_{j=1}^{i-1} \big(1 + \frac{2S^m}{\phi_j^m}\big) \Big) \end{cases}$$

We notice that $q_0^{m*} > 0.5$, $\forall m \in \{1, ..., N\}$. Therefore, in order to achieve his objective, the TPA needs the CP to believe that he will more probably not attempt to check the existence of any copy of the data. This can be interpreted as if the TPA will trust the CP to respect the requirements of the backup process for data D_m. However, the TPA does not take the option of checking the existence of at least one copy of the data off the table, even though the probability of such event is lower than the probability of not checking the existence of any copy at all.

V. Numerical Study

In this section, unless stated otherwise, we consider the baseline parameters $C^s = 0.1$, $C^t = 0.1$, and $\epsilon = 0.1$ and

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TIFS.2016.2549002, IEEE Transactions on Information Forensics and Security



(a) $\mu = (2, 0.5, 0.1, 0.5, 0.1)$

(b) $\mu = (2, 0.5, 0.1, 0.1, 1)$

(c) $\mu = (2, 1, 0.1, 0.1, 0.1)$

(d) $\mu = (2, 1, 0.1, 0.1, 0.1)$

Fig. 2: IS game

we analyze players' strategies w.r.t. increasing values of S. Let $\mu = (R, F, C^s, C^t, \epsilon)$.

A. Independent Strategies Game

The strategies of the CP and the TPA for data D_j do not depend on their strategies for data D_i, $\forall i \neq j$. Since we focus on each data D_m independently, we will drop the index m in this section. Let $D \in \mathcal{D}$ s.t. $F = 0.5$ and $R = 2$. We will study the impact of the CP storage cost S of the data D on both players' strategies.

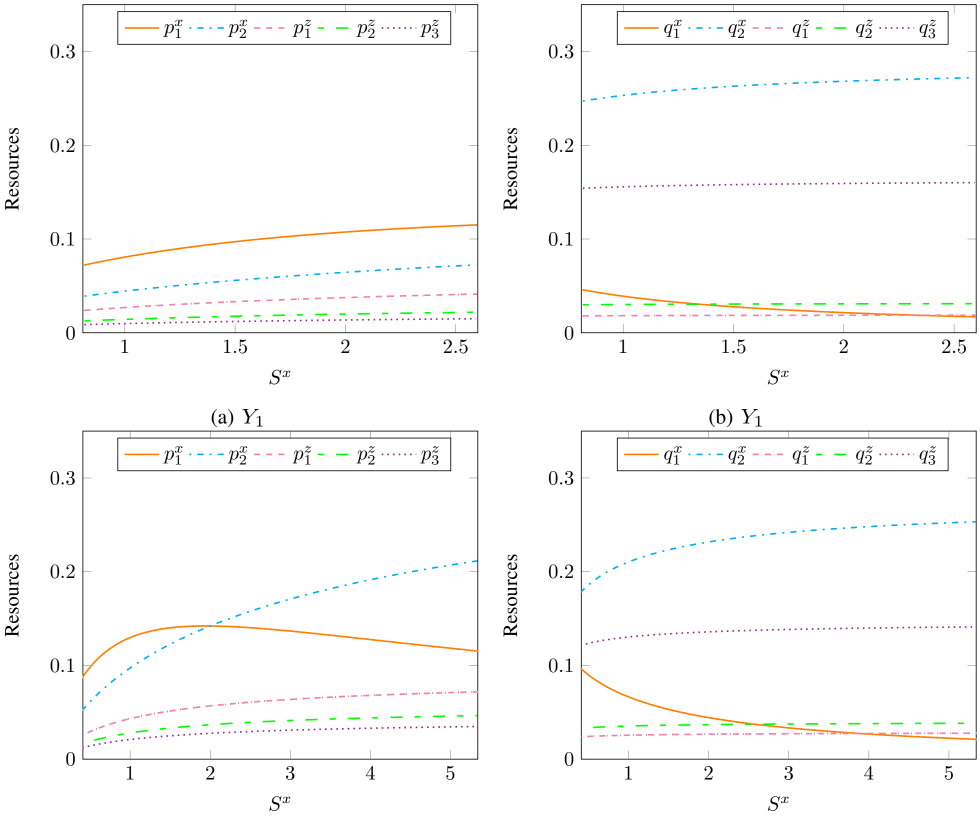
In Fig. 1a, the CP's strategy p_0^* decreases w.r.t. increasing values of S whereas p_2^* increases. We note that there exists a value of S under which $p_1^* > p_2^*$. From Fig. 1b, when the storage cost S of data D is small, the TPA will privilege not to check the existence of any backup copies. When S increases, the TPA's strategy q_0^* quickly decreases before stabilizing on a value greater than 0.5. On the other hand, q_2^* increases quickly before stabilizing. For small values of S, we observe a peak for q_1^* before decreasing and eventually stabilizing on a value less than 0.5. For small values of C^s and ϵ, when S increases, the values of q_0^* and q_2^* stabilize around 0.5. In this case, it is as if the choice of the TPA is restricted to whether to check all backup copies or none at all. The TPA does not have any interest in checking the existence of a number of backup copies less than the number required in the contract between the TPA and the client. In this case, the cost $C^s S$ paid by the TPA is relatively small when the CP passes the verification test. As a result, the TPA prefers to check the existence of all backup copies stored on the CP servers.

Impact of C^s. From Fig. 1d, an increase in the cost of executing the verification query $C^s S$ will have no significant impact on the pattern of change of the TPA's NE strategy w.r.t. to S. However, the stable values of q^* for large values of S change. In particular, they increase for q_0^* and q_1^* and decrease for q_2^*. The TPA increases the frequency of checking one copy instead of two copies, since checking either an honest CP or a CP that passes the verification test will entail a higher cost $C^s S$ for the TPA.

Impact of C^t. The TPA's strategy at the NE is independent of C^t. In Fig. 2a, as with greater values of C^s, a similar change is observed in the CP's NE strategy when increasing the cost of processing the verification query for the TPA $C^t S$. However, in this case, the CP's strategy changes more quickly w.r.t. S.

Impact of ϵ. In Fig. 2b, when ϵ increases, the TPA's NE strategy rate of change decreases. For large values of S, we notice an increase of the stable values for q_0^* and q_1^* and a decrease for q_2^*. The TPA's NE reflects his belief that the CP will more likely behave honestly given the increased incentive given to him when behaving as such. However, this incentive is given to the CP when the TPA fails to detect a malicious act by the CP and therefore, it does not completely prevent such scenario.

Impact of F. In Fig. 2c, when F increases, the rate of change of the CP's NE strategy decreases. For small values



(a) Y_1

(b) Y_1

(c) Y_2

(d) Y_2

Fig. 3: CSST game

of S/F, the CP has no interest in deleting any copies of the data since such action will entail a small payoff and exposes the CP to the risk of being detected by the TPA.

B. Correlated Strategies Game

For presentation reasons, we consider only two data items D_x and D_z and plot the strategies of the TPA and the CP when targeting at least one backup copy of the data. Let $R_x = 2$, $F_x = 1$, $R_z = 3$, $F_z = 2$, and $S^z = 1$. We will analyze the strategies of the TPA and the CP w.r.t. the importance S^x of the data D_x. Table II exhibits the different values of parameters used in this section.

TABLE II: Values of Parameters

	C^s	C^t	ϵ
$\boldsymbol{Y_1}$	0.1	0.1	0.1
$\boldsymbol{Y_2}$	0.5	0.1	0.1
$\boldsymbol{Y_3}$	0.1	0.1	1

1) Single Targets:

In a CSST game, each player can target one type of data and execute one action related to that data at each instance of the game.

In Fig. 3a, w.r.t. increasing values of S^x, we notice that p_1^{x*} and p_2^{x*} increase. As the importance of the data to the CP increases, he will be more tempted not to respect data backup requirements to free additional space on his servers. For data D_z (Fig. 3a), p_i^{z*} increases, $\forall i \in \{1, ..., 3\}$. On the other hand, we notice that there exists a value $S' \approx 1.6$ s.t. $\forall S > S'$, the CP will focus more on data D_z even though this does not necessarily translate in removing any backup copy of D_z at each instance of the game.

For the TPA (Fig. 3b), we notice that q_1^{x*} decreases and q_2^{x*} increases. In this case, the TPA privileges checking the maximum number of backup copies given the high value of the data S^x to the CP, which is correlated with an increase in the likelihood that the CP dishonors the backup agreement for D_x.

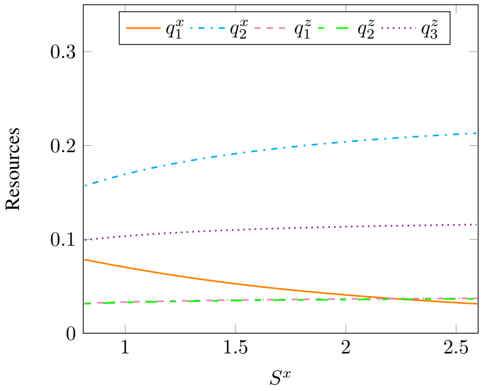
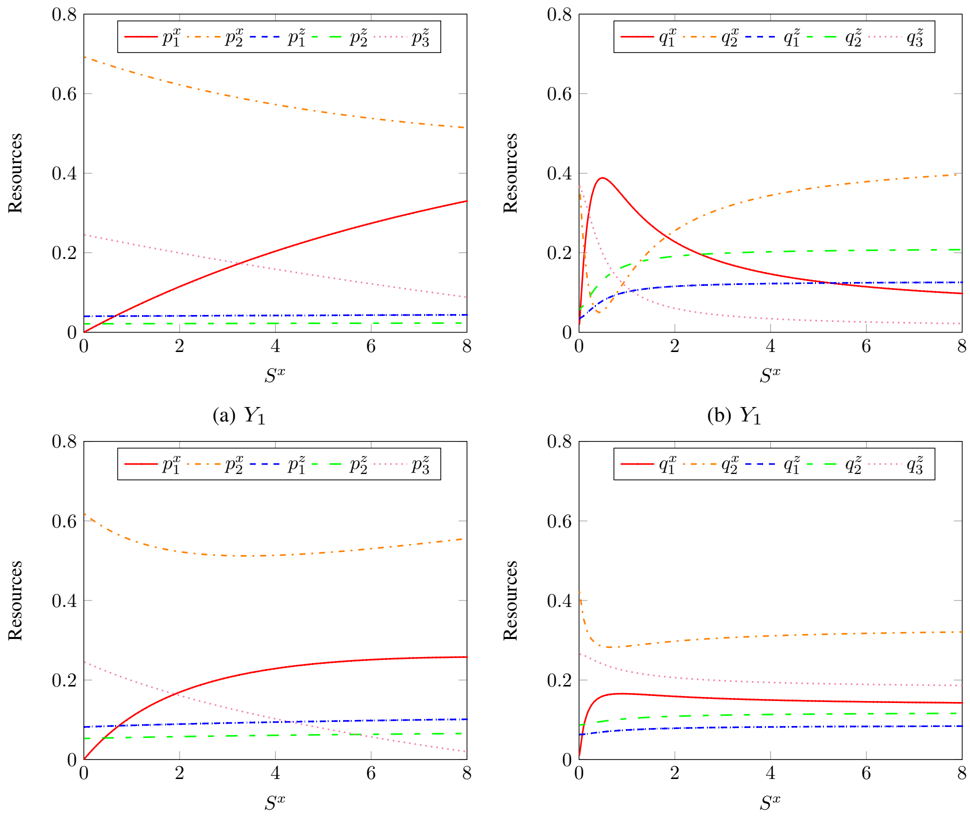


Fig. 4: CSST game: Y_3

Comparing Fig. 3b and Fig. 4, we notice that higher values of ϵ do not affect the pattern of change of the TPA's NE



(a) Y_1

(b) Y_1

(c) Y_2

(d) Y_2

Fig. 5: CSMT game

strategy for data D_x.

Finally, for greater values of C^s, the interval of values of S^x in which a NE exists widens (Fig. 3c). In addition, when S increases, the CP will allocate more resources to remove all backup copies of D_x.

2) Multiple Targets:

In a CSMT game, each player can target multiple types of data at each instance of the game. We suppose the resource constraints $P = 1$ and $Q = 0.85$. In this case, we find that the attractive set $T_S = \{D_x, D_z\}$.

In Fig. 5a, interestingly, when S_x increases, p_1^{x*} increases while p_2^{x*} and p_3^{z*} decrease. For higher values of S_x, the CP strategically manages his resources in order to increase his payoff by keeping only one copy of the data D_x, while at the same time reducing the risk of being caught by the TPA. On the other hand, the TPA responds by allocating more resources to verify the existence of two backup copies of D_x.

When C^s increases (Fig. 5c and Fig. 5d), the rate of change of both players' strategies increases. The TPA's NE strategy quickly stabilizes to certain values. Compared to Fig. 5b, the TPA reduces the allocated resources to verify the existence of all the backup copies of data D_x, since the cost of verification if the CP was acting honestly is higher. However, the TPA increases the resources to verify the existence of one copy of D_x and at least one copy of D_z.

While increasing the value of ϵ does not impact the CP's strategy at the NE, it directly affects the TPA's NE strategy (Fig. 6). In particular, we notice that the TPA will focus on verifying the existence of all the backup copies of D_x and D_z.

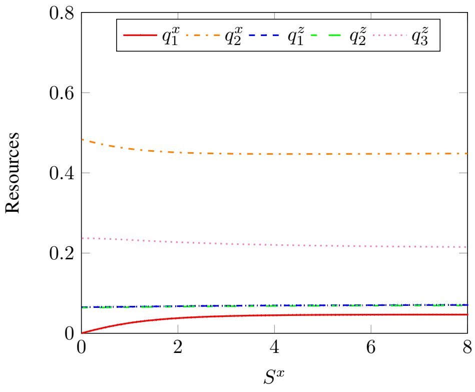
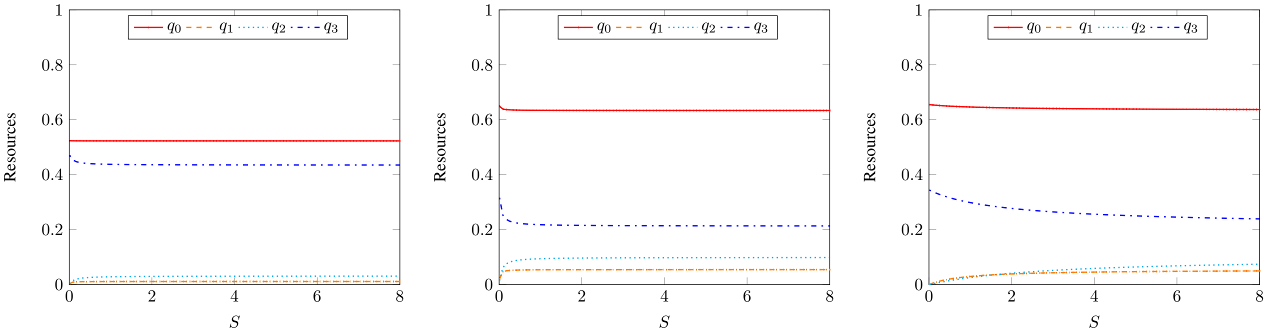


Fig. 6: CSMT game: Y_3

C. Stackelberg Game

In this section, the TPA chooses his strategy first. Then, informed by the TPA's choice, the CP chooses his strategy. We consider the case where the strategy for a data D_i is independent of the strategies for other data D_j, $\forall j \neq i$. In Section IV, we proved that in this case, a NE of the game exists. The TPA's strategy at the NE discourages the CP of dishonoring the data backup agreement with the client.

From Fig. 7a, w.r.t. increasing values of S, we find that q_0^* and q_R^* decrease while q_i^* increases, $\forall i \in \{1, ..., R\}$. However,



(a) $\mu = (3, 1, 0.1, 0.1, 0.1)$ (b) $\mu = (3, 1, 0.9, 0.1, 0.1)$ (c) $\mu = (3, 1, 0.9, 0.1, 5)$

Fig. 7: Stackelberg game

the TPA's strategy quickly stabilizes afterwards. We note that we always have $q_0^* > 0.5$. When C^s increases (Fig. 7b), for large values of S, the stable values of q_0^* increases, q_R^* decreases, and q_i^* increases, $\forall i \in \{1, ..., R\}$. Given the higher cost $C^s S$, the TPA focuses more on checking the existence of $k < R$ copies instead of R copies. When the incentive ϵ increases (Fig. 7c), the rate of change of the TPA's strategy decreases and it stabilizes slower w.r.t. S. In this case, for smaller values of ϵ, the TPA's incentive is not sufficient to deter the CP from dishonoring the requirements of the backup process agreed on with the client, which forces the TPA to adopt an aggressive strategy even for small values of S.

VI. Practical Scenario

In this section, we present a practical application of our model through a concrete scenario based on storage Cloud.

A. Parameter Evaluation

Applying a theoretical model in a realistic scenario means being able to evaluate each of the parameters in the model. In this section, we provide a guideline of how to evaluate the model parameters by the different players in the game. We introduce additional intermediate parameters, which will be used to deduce the parameters used in the model. $\forall m \in \{1, ..N\}$, let T^m represent the size of data D_m, measured in bytes. Given a data integrity verification protocol, let b be the size ratio of the data being checked (e.g. $b = 0.1$ if 10% is the proportion of data D that is checked by the TPA), and t_{CP} and t_{TPA} the execution times of the protocol on the CP and the TPA sides respectively.

The input parameters of the model are the number of data items N, the data set $\mathcal{D}$, and for each data D_m, its size T^m and the number of backup copies R^m. The value of R^m is assumed to be known as it can be part of the SLA with the cloud provider. From these values, we will first evaluate S^m, which is the financial value corresponding to the storage of one copy of data D_m by the CP. Based on [23], the storage costs can be precisely deduced from T^m. If we denote α the storage cost per bit in a given fixed period of time, the value of which can be obtained from [23], then we get $S^m = \alpha T^m$. Note that S^m could also take into account the estimated financial loss due to the amount of money that the CP may have earned with the omission of one copy of D_m.

The next step consists of evaluating F^m, which is the financial value corresponding to the importance of one copy of data D_m from the client's perspective. In general, F^m is correlated to R^m. In fact, the client is more likely to ask for additional backups copies of his most important data, as it could happen in the context of Cloud archiving, for instance. The very nature of this parameter makes risk assessment methods, such as EBIOS [24], one of the relevant methods to obtain the necessary information allowing the client to evaluate it. Financial cost due to data loss may be deduced from business knowledge, relying on criteria such as the loss of competitive advantage, or the difficulty to reproduce the data. Based on this assessment, the value F^m of one copy can be considered to be equal to the estimated value of the data divided by the number of copies R^m.

The verification cost and the cost of executing the verification query are given by $C^t S^i$ and $C^s S^i$ respectively. For a given data D_m of size T^m, t_{CP} and t_{TPA} can be measured from an implementation of the data integrity verification protocol, and the size ratio b to be checked in order to obtain a reliable proof, which can be deduced from the reference paper describing the protocol. From the values of t_{CP} and t_{TPA}, the number of CPU cycles can be estimated given the host characteristics. From [23], knowing the number of CPU cycles, the execution costs per bit for the CP and the TPA C_{bit}^{CP} and C_{bit}^{TPA} respectively can be deduced. From the available information so far, we can write the following equation $C^s S^m = bT_m C_{bit}^{CP}$. Since $S^m = \alpha T^m$, we have $C^s = \frac{bC_{bit}^{CP}}{\alpha}$.

In the case study, it is more interesting to assess the impact of the parameter ϵ on the players' strategies rather than define a method to evaluate it. ϵF^m is used as a reward for the CP for acting honestly. For example, it may refer to gains in terms of the reputation of the CP that is being highlighted by the TPA for the good behavior. The objective for the TPA would therefore be to find the optimal value of ϵ that decreases the probability that the CP acts dishonestly at the NE.

B. Numerical Example

We consider the case where $N = 3$ data items are outsourced. The characteristics of data D_1, D_2, D_3 are defined as in Table III.

TABLE III: Data Characteristics

	$\boldsymbol{T^m}$ (in GB)	$\boldsymbol{R^m}$	$\boldsymbol{F^m}$ (in $)
$\boldsymbol{D_1}$	0.01	4	200
$\boldsymbol{D_2}$	5	5	300
$\boldsymbol{D_3}$	200	2	130

First, we compute the value of S^m for each data D_m. From [23], we know that the storage cost for a CP can be estimated between about 100 picocent/bit and 300 picocent/bit per year (1 picocent = 10^{-14} $). In this case study, we consider a time period of one year, and an average storage value of 200 picocent/bit. Therefore, we find $S^1 = 0.000016$ $, $S^2 = 0.08$ $, and $S^3 = 3.2$ $.

For the evaluation of C^s and C^t, the verification scheme we implemented is based on the open source proof of retrievability project by Zachary Peterson, and corresponds to the basic POR scheme from [11], where the number of checked sentinel blocks represent $b = 1\%$ of the data file size. We used a Linux Virtual Machine running on a laptop with an i7 Intel Core processor, with 2.3 GHz clock frequency, and 8 GB of RAM. For the biggest data D_3, we measured $t_{TPA} = 1.51s$ and $t_{CP} = 0.27s$, the difference being due to the fact that there is no specific processing on the CP side besides giving the correct sentinel blocks in this scheme. Based on the values in [23], the CP cycle cost can be estimated at 2 picocent/cycle, while the TPA cycle cost should rather be around 20 picocent/cycle. Therefore $C^s S^3 = 1.24.10^{-5}$ $, and $C^t S^3 = 7.0.10^{-4}$ $, which gives us $C^s = 3.88.10^{-6}$ and $C^t = 2.19.10^{-4}$.

We assess the impact of ϵ on the Nash Equilibrium strategies in the case of the CSMT game, where $P = 1$ and $Q = 0.75$. Since the primary objective consists of finding the optimal checking strategy, we focus in this section on the behavior of the TPA. Table IV depicts the probability of checking the existence of at least one backup copy of each data for different values of ϵ. We notice that when the value of ϵ increases, the TPA will spend less resources on checking the existence of at least one backup copy of data D_2. This can be explained by the fact that D_2 has the highest value F^2. For the CP, a higher ϵ means receiving a substantial reward when he acts honestly. In this case, the TPA will therefore not waste too many resources on checking the existence of backup copies of this data, as the incentive is assumed to be high enough for the CP to behave honestly. However, if we multiply the size of data D_2 by 10 (therefore $S^2 = 0.8$ $), for $\epsilon = 0.01$, we notice that the TPA will spend twice as much resources to check the existence of at least one backup copy of D_2 w.r.t. the results for $\epsilon = 0.01$ in Table IV. In this case, for data D_2, the TPA anticipates that the reward will be less effective in preventing the CP from acting dishonestly.

TABLE IV: Probability of checking at least one backup copy of the data at the NE

	$\epsilon = 0.01$	$\epsilon = 0.1$
$\boldsymbol{D_1}$	0.137	0.161
$\boldsymbol{D_2}$	0.117	0.091
$\boldsymbol{D_3}$	0.496	0.498

An example of finding an optimal value for ϵ would be to find the minimum incentive that guarantees that the TPA will not need to use more than 10% of his resources for checking the existence of at least one backup copy of data D_2 at the NE. Running the optimization in this case study, we find the value of 0.028 for ϵ.

It is worth mentioning that the difference between the values F of the data items and the other parameters in this case study is substantial, which could raise concerns about the influence of the values of F over the other parameters. However, if we multiply the size of data D_3 by 10, bringing S^3 to 32 $, which is not negligible compared to F^3, we do not notice a big difference in the TPA's NE strategy w.r.t. the results in Table IV.

VII. Conclusion

In this paper, we analyzed the problem of verifying data availability in the case of data outsourced to a cloud provider. We formulated the problem between the CP and the TPA as a non-cooperative game. The TPA's objective is to detect any deviation from the agreement signed between the CP and the client by checking the existence of the required number of backup copies of each type of data on the CP's servers. On the other hand, the CP's objective is to increase the storage capacity on his servers, which translates in practice in the existence of a number of copies less than the required number included in the contract with the client. We performed an in-depth analysis of multiple extensions of the simple model in [4] taking into account the existence of multiple backup copies of each data. In each proposed extension, we identified the optimal verification strategy for the TPA. Finally, we validated our analytical results on a case study.

One of the interesting results that we found relates to the stackelberg game in which we have a leader (the TPA) and a follower (the CP) in the game. This type of games reflects realistic scenarios that we can encounter in real life. Interestingly, our results show that a NE of the game exists and when it is achieved, the CP cannot improve his utility by acting dishonestly. At the NE, it is as if the trust of the TPA in the CP's actions outweigh any belief of a potential misconduct.

The results in this paper rely on the basic assumption of the rationality of the CP and the TPA, which is a reasonable hypothesis in this case. However, one may argue about the relevance of the different types of parameters introduced in the model and the cost allocations (who will need to pay what). For instance, the signed agreement between the CP and the client can specify that the CP must always take charge of the cost of executing the verification query. While this is a realistic assumption, always taking the burden of this cost by the CP may result in an abusive verification behavior by the TPA. Therefore, in this paper, we distinguished which player needs to pay that cost according to the detection of a malicious act by the CP. With ϵF, these parameters play the role of incentives

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TIFS.2016.2549002, IEEE Transactions on Information Forensics and Security

and punishments for the CP and allows us to analyze their subsequent effects on his behavior. In addition, the analysis of the different types of games can be used not only to study the behavior of players, but can also be leveraged to help adjust these incentives and punishments to be aligned with the client's interests when negotiating an SLA with a CP.

The model presented in this paper can be adapted to verify the existence of the required number of backup copies in specific geographical locations as is sometimes specified in an SLA. As future work, we plan to investigate the case where interactions between the CP and the TPA can occur on multiple occasions over time. This type of interactions is particularly interesting if we consider a repeated game setting where we have a number of TPAs, on behalf of multiple clients, verifying the CP's compliance with the signed agreements with the clients. In this case, the result of the interactions between the CP and a client is not limited to that particular client, but extends to impact the behavior of all the other players in the game. For example, we can study how the discovery of an improper act by the CP can affect his reputation and therefore his future payoffs, as clients will be more inclined to change provider. In this case, players' behaviors may change after it has been made public that a CP breached his agreement with a client. Therefore, each short-term gain of the CP must be weighted against the enduring long-term impact on his reputation, which automatically affects his future profits. The public exposure of the behavior of the CP is an important dimension that needs to be taken into account, which can play a decisive role of deterrence to force the CP to fully respect the backup agreements signed with the clients.

Appendix A

***Proof of Theorem 1**:*

In this case, considering the data independence hypothesis, we solve the game by focusing on any fixed data D_m independently from the other data. Considering that $p_0^m = 1 - \sum_{i=1}^{R^m} p_i^m$, and $q_0^m = 1 - \sum_{i=1}^{R^m} q_i^m$ and integrating these constraints in the payoff functions, at the optimum we have: $\frac{\partial U_A(p,q)}{\partial p_i^m} = 0$ and $\frac{\partial U_D(p,q)}{\partial q_i^m} = 0$, $\forall i \in \{1, ... R^m\}$.

We have $\forall j \in \{1, ..., R^m\}$:

$$\frac{\partial U_D(p,q)}{\partial q_j^m} = \sum_{i=1}^{R^m} p_i^m (iF^m) \mathbb{1}_{i > R^m - j} - jC^t S^m + \sum_{i=1}^{R^m} p_i^m (iF^m) - \sum_{i=0}^{R^m} p_i^m (jC^s S^m) \mathbb{1}_{i \leq R^m - j} - \sum_{i=1}^{R^m} p_i^m (iF^m) \mathbb{1}_{i \leq R^m - j}$$

We have $\frac{\partial U_D(p,q)}{\partial q_j^m} - \frac{\partial U_D(p,q)}{\partial q_{j-1}^m} = 0$, $\forall j \geq 2$. Therefore, $\forall i \in \{1, ..., R^m\}$, we have:

$$(p_i^m)^* \theta_i^m = C^t S^m + C^s S^m \sum_{j=0}^{i-1} (p_j^m)^* \tag{4}$$

From Equation 4, we prove by induction the following result, $\forall j \in \{1, ..., R^m - 1\}$:

$$(p_j^m)^* = \frac{C^s S^m (p_0^m)^* + C^t S^m}{\theta_j^m} \prod_{j=1}^{i-1} \Big(1 + \frac{C^s S^m}{\theta_j^m}\Big)$$

Moreover, solving $\frac{\partial U_D(p,q)}{\partial q_1^m} = 0$ gives $(p_{R^m}^m)^* = \dfrac{C^t S^m + C^s S^m}{2R^m F^m + C^s S^m}$

Given the constraint $\sum_{i=0}^{R^m} p_i^m = 1$, we find $(p_0^m)^*$.

Similarly, we find the TPA strategy at the NE $(q_i^m)^*$.

***Proof of Lemma 1**:*

Let $x = F^m / S^m$. In this case, p_0^{m*} can be written as $p_0^{m*} = \dfrac{1 - \dfrac{C^t + C^s}{2R^m x + C^s} - C^t \Delta}{1 + C^s \Delta}$, where $\Delta = \sum_{i=1}^{R^m - 1} \dfrac{1}{2ix + (R^m - i)C^s} \prod_{j=1}^{i-1} \Big(1 + \dfrac{C^s}{2jx + (R^m - j)C^s}\Big)$.

Therefore, $\dfrac{\partial p_0^{m*}}{\partial x} = \dfrac{\dfrac{2R^m (C^t + C^s)}{2R^m x + C^s} \Big(\dfrac{1}{2R^m x + C^s} - x \dfrac{\partial \Delta}{\partial x}\Big)}{(1 + C^s \Delta)^2}$.

$$\frac{\partial \Delta}{\partial x} = \sum_{i=1}^{R^m - 1} \frac{-2i}{(2ix + (R^m - i)C^s)^2} \prod_{j=1}^{i-1} \Big(1 + \frac{C^s}{2jx + (R^m - j)C^s}\Big) + \sum_{i=1}^{R^m - 1} \frac{1}{2ix + (R^m - i)C^s} \sum_{k=1}^{i-1} \frac{-2kC^s x}{(2ks + (R^m - k)C^s)^2} \prod_{j=1, j \neq k}^{i-1} \Big(1 + \frac{C^s}{2jx + (R^m - j)C^s}\Big) < 0.$$

As a result, $\dfrac{\partial p_0^{m*}}{\partial x} > 0$ and p_0^{m*} is a strictly increasing function with respect to F^m / S^m.

p_0^{m*} is a continuous function in $[0; +\infty[$. We have for $F^m = 0$, $p_0^{m*} < 0$. For $F^m / S^m \to +\infty$, $p_0^{m*} \to 1$, and as a result $\exists y > 0$ s.t. $\forall F^m / S^m > y$, $p_0^{m*} > 0$. Therefore, by the Intermediate Value Theorem and the fact that p_0^{m*} is a strictly increasing function w.r.t. F^m / S^m, there exists only one value $x_0^m = F^m / S^m$ s.t. $p_0^{m*}(x_0^m) = 0$.

***Proof of Lemma 2**:*

$p_0^{m*} = \dfrac{\alpha^m}{\beta^m} \Rightarrow \dfrac{\partial p_0^{m*}}{\partial S^m} = \dfrac{\beta^m \frac{\partial \alpha^m}{\partial S^m} - \alpha^m \frac{\partial \beta^m}{\partial S^m}}{(\beta^m)^2}$. Let $\Delta^i = \sum_{j=1}^{R^i - 1} \dfrac{S^i \psi_j^i}{\theta_j^i}$ and $B^i = \sum_{j=1}^{R^i - 1} \dfrac{jS^i}{\theta_j^i} \psi_j^i$, $\forall i \in \{1, ..., N\}$.

We have:

$$\beta^m \frac{\partial \alpha^m}{\partial S^m} - \alpha^m \frac{\partial \beta^m}{\partial S^m} = \Big(- \frac{\partial \Delta^m}{\partial S^m} - \frac{1}{2F^m} + \frac{1}{R^m} \frac{\partial B^m}{\partial S^m} - R^m \sum_{i=1, i \neq m}^{N} \frac{1}{S^i R^i} (\Delta^i + \frac{1}{C^s} + \frac{S^i (R^i - 2\omega^i)}{2R^i F^i}) \Big) \Big(C^s \alpha^m + C^t \beta^m \Big)$$

However, $C^s \alpha^m + C^t \beta^m = C^s + NC^t > 0$, $\Delta^i - \dfrac{S^i \omega^i}{R^i F^i} > 0$ $\forall i \in \{1, ..., N\}$, and $-\dfrac{\partial \Delta^m}{\partial S^m} + \dfrac{1}{R^m} \dfrac{\partial B^m}{\partial S^m} < 0$. Therefore, we have $\dfrac{\partial p_0^{m*}}{\partial S^m} < 0$ and p_0^{m*} is a strictly decreasing function with respect to S^m.

p_0^{m*} is a continuous function in $[0; +\infty[$. We have for $S^m = 0$, $p_0^{m*} = 1 + (N-1)\dfrac{C^t}{C^s} > 1$. For $S^m \to +\infty$, $p_0^{m*} \to -\dfrac{C^t}{C^s} < 0$, and as a result $\exists y > 0$ s.t. $\forall S^m > y$, we have $p_0^{m*} < 0$. Therefore, by the Intermediate Value Theorem, there exists only one value S_2^m s.t. $p_0^{m*}(S_2^m) = 0$. In addition, given

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TIFS.2016.2549002, IEEE Transactions on Information Forensics and Security

the fact that p_0^{m*} is a strictly decreasing function w.r.t. S^m, and that $p_0^{m*}(0) > 1$ and $p_0^{m*}(S_2^m) = 0$, there exists only one value $S_1^m > 0$ s.t. $p_0^{m*}(S_1^m) = 1$.

Proof of Theorem 3:

Let us suppose that TPA and the CP focus on the attractive set $\mathcal{T}_S$. Therefore, we have $\sum\limits_{m\in\mathcal{T}_S}\sum\limits_{i=1}^{R^m} p_i^m = P$, $\sum\limits_{m\in\mathcal{T}_S}\sum\limits_{j=1}^{R^m} q_j^m = Q$, $\sum\limits_{i=0}^{R^m} p_i^m = 1$ and $\sum\limits_{j=0}^{R^m} q_j^m = 1$ $\forall m \in \mathcal{T}_S$ where m refers to data D_m.

We have $\frac{\partial U_D(p,q)}{\partial q_i^m} = \nu$, $\forall m \in \mathcal{T}_S$ $\forall i \in \{1,...,R^m-1\}$ where $\nu > 0$.

Similarly to the proof of the Theorem 1, we find that $\forall j \in \{1,...,R^m-1\}$, we have:

$$p_j^m = \frac{C^s S^m p_0^m + C^t S^m}{\theta_j^m}\psi_j^m \tag{5}$$

$$\frac{\partial U_D(p,q)}{\partial q_{R^m}^m} = 2\sum_{i=1}^{R^m} p_i^m(iF^m) - R^m C^t S^m - R^m C^s S^m p_0^m = \nu$$

$$\Rightarrow p_{R^m}^m = \frac{(C^t S^m + C^s S^m p_0^m)(R^m - 2\omega^m) + \nu}{2R^m F^m} \tag{6}$$

$$\forall j \in \{1,...,R^m-1\},\ p_j^m = \frac{C^s S^m p_0^m + C^t S^m}{\theta_j^m}\psi_j^m$$

$$\Rightarrow \sum_{i=1}^{R^m-1} p_i^m = 1 - p_0^m - p_{R^m}^m = (C^s S^m p_0^m + C^t S^m)\sum_{i=1}^{R^m-1}\frac{\psi_i^m}{\theta_i^m} \tag{7}$$

Let E^m and G^m be defined as in Appendix B.

From Equations 6 and 7, we find $p_0^m = \dfrac{-\nu + G^m}{E^m}$.

We have $\sum\limits_{m\in\mathcal{T}_S}\sum\limits_{i=1}^{R^m} p_i^m = P \Rightarrow \sum\limits_{m\in\mathcal{T}_S} p_0^m = |\mathcal{T}_S| - P$

$$\Rightarrow \nu = \frac{\sum\limits_{m\in\mathcal{T}_S}\frac{G^m}{E^m} - (|\mathcal{T}_S| - P)}{\sum\limits_{m\in\mathcal{T}_S}\frac{1}{E^m}}$$

We find the strategy p^m of the CP by replacing ν in Equations 5 and 6.

We have $\frac{\partial U_A(p,q)}{\partial p_i^m} = \kappa$, $\forall m \in \mathcal{T}_S$ $\forall i \in \{1,...,R^m-1\}$ where $\kappa > 0$.

Solving the system of equations, we find that $\forall j \in \{2,...,R^m-1\}$, we have:

$$q_j^m = \frac{\phi_1^m}{\phi_j^m} q_1^m \tag{8}$$

$$\frac{\partial U_A(p,q)}{\partial p_1^m} = \kappa \Rightarrow q_{R^m}^m = \frac{S^m - \kappa}{2S^m + \phi_{R^m}^m}$$

We have $\frac{\partial U_A(p,q)}{\partial p_{R^m}^m} - \frac{\partial U_A(p,q)}{\partial p_{R^m-1}^m} = 0$

$$\Rightarrow q_1^m(\phi_1^m - S^m) = S^m q_0^m - S^m \sum_{j=2}^{R^m} q_j^m$$

$$\Rightarrow q_1^m = \frac{(2q_0^m - 1)S^m}{\phi_1^m - 2S^m}$$

Let H^m and I^m be defined as in Appendix B.

We have $q_i^m = \dfrac{\phi_1^m}{\phi_i^m} q_1^m$, $\forall i \in \{2,...,R^m-1\}$.

$$\Rightarrow \sum_{i=2}^{R^m-1} q_i^m = 1 - q_0^m - q_1^m - q_{R^m}^m = \phi_1^m q_1^m \sum_{i=2}^{R^m-1}\frac{1}{\phi_i^m}$$

$$\Rightarrow q_{R^m}^m = 1 - H^m q_0^m - W^m \tag{9}$$

$$\sum_{m\in\mathcal{T}_S}\sum_{i=1}^{R^m} q_i^m = Q$$

$$\Rightarrow \sum_{m\in\mathcal{T}_S} q_0^m = |\mathcal{T}_S| - Q$$

$$\Rightarrow \kappa = \frac{|\mathcal{T}_S| - Q - \sum\limits_{m\in\mathcal{T}_S}\dfrac{(1-W^m)(2S^m + \phi_{R^m}^m) - S^m}{H^m(2S^m + \phi_{R^m}^m)}}{\sum\limits_{m\in\mathcal{T}_S}\dfrac{1}{H^m(2S^m + \phi_{R^m}^m)}}$$

We find the strategy q^m of the TPA by replacing κ in Equations 8 and 9.

Let us suppose that the TPA focuses on the attractive set $\mathcal{T}_S$. We want to find whether the CP will only be interested in targeting data in $\mathcal{T}_S$ or if he will attempt to target any data $D_i \in \mathcal{D}\backslash\mathcal{T}_S$.

We consider a strategy vector p for the CP s.t. $\sum\limits_{m\in\mathcal{D}\backslash\mathcal{T}_S}\sum\limits_{i=1}^{R^m} p_i^m > 0$.

Let $x \in \mathcal{T}_S$ and $r \in \{1,...,R^x\}$. We define a vector p' based on p as follows:

$$p_i^{m'} = \begin{cases} p_i^m & m \in \mathcal{T}_S\ and\ m \neq x,\ i \in \{1,...,R^m\} \\ p_i^x & m = x,\ i \in \{1,..,r-1,r+1,...,R^x\} \\ p_r^x + \sum\limits_{j\in\mathcal{D}\backslash\mathcal{T}_S}\sum\limits_{i=1}^{R^j} p_i^j & m = x\ and\ i = r \\ 0 & m \in \mathcal{D}\backslash\mathcal{T}_S \end{cases}$$

Therefore,

$$\begin{aligned} U_A(p,q^*) - U(p',q^*) &= \sum_{m\in\mathcal{D}\backslash\mathcal{T}_S}\big(\sum_{i=1}^{R^m} p_i^m\big)(iS^m) \\ &-\big(\sum_{m\in\mathcal{D}\backslash\mathcal{T}_S}\sum_{i=1}^{R^m} p_i^m\big)\big(-\sum_{j=1}^{R^x} q_j^x(rS^x + jC^sS^x)\mathbb{1}_{r>R^x-j} \\ &+\epsilon F^x\sum_{j=1}^{R^x} q_j^x(j)\mathbb{1}_{r\leq R^x-j} + \sum_{j=0}^{R^x} q_j^x(rS^x)\mathbb{1}_{r\leq R^x-j} \\ &-\epsilon F^x\sum_{j=1}^{R^x} q_j^x(j)\big) \\ &= \sum_{m\in\mathcal{D}\backslash\mathcal{T}_S}\big(\sum_{i=1}^{R^m} p_i^m\big)\big(iS^m - \kappa\big) \\ &\leq \sum_{m\in\mathcal{D}\backslash\mathcal{T}_S}\big(\sum_{i=1}^{R^m} p_i^m\big)\big(\max_{D_j\in\mathcal{D}\backslash\mathcal{T}_S}(S^jR^j) - \kappa\big) < 0 \end{aligned}$$

Therefore, when $\max\limits_{D_j\in\mathcal{D}\backslash\mathcal{T}_S}(S^jR^j) < \kappa$ and the TPA chooses to focus on $\mathcal{T}_S$, the CP is better off focusing on this set too.

Finally, the necessary conditions for the solution to be a NE are:

$$\begin{cases} |\mathcal{T}_S| - \sum\limits_{m\in\mathcal{T}_S}\frac{G^m}{E^m} + \max\limits_i\big(G^i - \frac{(1-C^tS^i\tau^i)E^i}{1+C^sS^i\tau^i}\big)\sum\limits_{m\in\mathcal{T}_S}\frac{1}{E^m} \\ \qquad < P < |\mathcal{T}_S| - \sum\limits_{m\in\mathcal{T}_S}\frac{G^m}{E^m} + \min\limits_i G^i\sum\limits_{m\in\mathcal{T}_S}\frac{1}{E^m} \\ |\mathcal{T}_S| - \min\limits_i S^i\sum\limits_{m\in\mathcal{T}_S}\frac{1}{H^m(2S^m+\phi_{R^m}^m)} \\ \qquad - \sum\limits_{m\in\mathcal{T}_S}\frac{(1-W^m)(2S^m+\phi_{R^m}^m)-S^m}{H^m(2S^m+\phi_{R^m}^m)} \\ \qquad < Q < |\mathcal{T}_S| + \min\limits_i(S^i + \phi_{R^i}^i)\sum\limits_{m\in\mathcal{T}_S}\frac{1}{H^m(2S^m+\phi_{R^m}^m)} \\ \qquad - \sum\limits_{m\in\mathcal{T}_S}\frac{(1-W^m)(2S^m+\phi_{R^m}^m)-S^m}{H^m(2S^m+\phi_{R^m}^m)} \end{cases}$$

Appendix B

$$
\begin{aligned}
\theta_i^m &= 2iF^m + (R^m - i)C^s S^m \\
\phi_i^m &= 2(R^m - i)S^m + i(C^s S^m + \epsilon F^m) \\
\psi_i^m &= \prod_{j=1}^{i-1} \Big(1 + \frac{C^s S^m}{\theta_j^m}\Big) \\
\omega^m &= \sum_{i=1}^{R^m-1} \frac{iF^m}{\theta_i^m} \psi_i^m \\
\tau^m &= \sum_{i=1}^{R^m-1} \frac{\psi_i^m}{\theta_i^m} \\
\alpha^m &= 1 + \frac{NC^t}{C^s} - \frac{C^t S^m R^m}{C^s} \sum_{i \in \mathcal{D}} \Big(\frac{1}{S^i R^i} \\
&\qquad\qquad\qquad + \frac{C^s}{R^i}(\tau^i + \frac{R^i - 2\omega^i}{2R^i F^i})\Big) \\
\beta^m &= S^m R^m \sum_{i \in \mathcal{D}} \Big(\frac{1}{S^i R^i} + \frac{C^s}{R^i}(\tau^i + \frac{R^i - 2\omega^i}{2R^i F^i})\Big) \\
\gamma^m &= \frac{1}{\Big(1 + \frac{S^m}{\phi_{R^m}^m}\Big)\Big(\phi_1^m(1 - \frac{S^m}{\phi_{R^m-1}^m} + \frac{(S^m)^2}{\phi_{R^m-1}^m} + ((S^m)^2 - 1)\sum_{j=2}^{R^m-2} \frac{1}{\phi_j^m}) + (S^m)^2\Big)} \\
\delta^m &= \frac{S^m}{S^m + R^m(C^s S^m + \epsilon F^m)} + \gamma_m S^m \Big(1 + \phi_1^m \sum_{j=2}^{R^m-1} \frac{1}{\phi_j^m}\Big) \\
\eta^m &= \frac{F^m}{\delta^m R^m F^m + \gamma^m \phi_1^m \omega^m} \\
E^m &= 2R^m F^m + C^s S^m (R^m - 2\omega^m) + 2R^m F^m C^s S^m \sum_{i=1}^{R^m-1} \frac{\psi_i^m}{\theta_i^m} \\
G^m &= 2R^m F^m - C^t S^m (R^m - 2\omega^m) - 2R^m F^m C^t S^m \sum_{i=1}^{R^m-1} \frac{\psi_i^m}{\theta_i^m} \\
\nu &= \frac{\sum_{m \in \mathcal{T}_S} \frac{G^m}{E^m} - (|\mathcal{T}_S| - P)}{\sum_{m \in \mathcal{T}_S} \frac{1}{E^m}} \\
H^m &= 1 + \frac{2S^m \sum_{i=1}^{R^m-1} \frac{\phi_1^m}{\phi_i^m}}{\phi_1^m - 2S^m} \\
W^m &= \frac{-S^m \sum_{i=1}^{R^m-1} \frac{\phi_1^m}{\phi_i^m}}{\phi_1^m - 2S^m} \\
\kappa &= \frac{|\mathcal{T}_S| - Q - \sum_{m \in \mathcal{T}_S} \frac{(1 - W^m)(2S^m + \phi_{R^m}^m) - S^m}{H^m(2S^m + \phi_{R^m}^m)}}{\sum_{m \in \mathcal{T}_S} \frac{1}{H^m(2S^m + \phi_{R^m}^m)}}
\end{aligned}
$$

References

[1] P. Mell and T. Grance, "The NIST definition of cloud computing (draft)," *NIST special publication*, vol. 800, no. 145, p. 7, 2011.

[2] M. Armbrust, A. Fox, R. Griffith, A. Joseph, R. Katz, A. Konwinski, G. Lee, D. Patterson, A. Rabkin, I. Stoica *et al.*, "A view of cloud computing," *Communications of the ACM*, vol. 53, no. 4, pp. 50–58, 2010.

[3] A. Kochumol and M. Win, "Proving possession and retrievability within a cloud environment: A comparative survey," *International Journal of Computer Science and Information Technologies*, vol. 5, no. 1, pp. 478–485, 2014.

[4] B. Djebaili, C. Kiennert, J. Leneutre, and L. Chen, "Data integrity and availability verification game in untrusted cloud storage," in *Proceedings of the 5th International Conference on Decision and Game Theory for Security (GameSec)*, 2014, pp. 287–306.

[5] R. Popa, J. Lorch, D. Molnar, H. Wang, and L. Zhuang, "Enabling security in cloud storage SLAs with CloudProof," in *Proceedings of the 2011 USENIX Conference on USENIX Annual Technical Conference*, ser. USENIXATC'11, 2011, pp. 31–31.

[6] C. Wang, Q. Wang, K. Ren, and W. Lou, "Privacy-preserving public auditing for data storage security in cloud computing," in *INFOCOM*, 2010, pp. 1–9.

[7] G. Ateniese, R. Di Pietro, L. Mancini, and G. Tsudik, "Scalable and efficient provable data possession," in *Proceedings of the 4th international conference on Security and privacy in communication networks*, 2008.

[8] C. Erway, A. Küpçü, C. Papamanthou, and R. Tamassia, "Dynamic provable data possession," in *Proceedings of the 16th ACM conference on Computer and communications security*, 2009, pp. 213–222.

[9] Y. Zhu, H. Wang, Z. Hu, G.-J. Ahn, H. Hu, and S. Yau, "Efficient provable data possession for hybrid clouds," in *Proceedings of the 17th ACM Conference on Computer and Communications Security*, ser. CCS '10, 2010, pp. 756–758.

[10] J. Yang, H. Wang, J. Wang, C. Tan, and D. Yu, "Provable data possession of resource-constrained mobile devices in cloud computing," *JNW*, vol. 6, no. 7, pp. 1033–1040, 2011.

[11] A. Juels and B. Kaliski, "PORs: Proofs of retrievability for large files," in *Proceedings of the 14th ACM conference on Computer and communications security*, 2007, pp. 584–597.

[12] H. Shacham and B. Waters, "Compact proofs of retrievability," in *Proceedings of the 14th International Conference on the Theory and Application of Cryptology and Information Security: Advances in Cryptology (ASIACRYPT)*, 2008, pp. 90–107.

[13] R. Curtmola, O. Khan, R. Burns, and G. Ateniese, "Mr-pdp: Multiple-replica provable data possession," in *Proceedings of the 28th International Conference on Distributed Computing Systems (ICDCS'08)*, 2008, pp. 411–420.

[14] M. Hassan, B. Song, and E.-N. Huh, "Distributed resource allocation games in horizontal dynamic cloud federation platform," in *Proceedings of the 13th International Conference onHigh Performance Computing and Communications (HPCC)*, 2011, pp. 822–827.

[15] X. Zheng, P. Martin, W. Powley, and K. Brohman, "Applying bargaining game theory to web services negotiation," in *2010 IEEE International Conference on Services Computing (SCC)*, 2010, pp. 218–225.

[16] R. Nix and M. Kantarcioglu, "Contractual agreement design for enforcing honesty in cloud outsourcing," in *Decision and Game Theory for Security*, 2012, pp. 296–308.

[17] ——, "Efficient query verification on outsourced data: A game-theoretic approach," *arXiv preprint arXiv:1202.1567*, 2012.

[18] A. Gueye and V. Marbukh, "A game-theoretic framework for network security vulnerability assessment and mitigation," in *Decision and Game Theory for Security*, 2012, pp. 186–200.

[19] T. Alpcan and T. Basar, *Network Security: A Decision and Game-Theoretic Approach*. Cambridge University Press, 2010.

[20] L. Chen and J. Leneutre, "A game theoretical framework on intrusion detection in heterogeneous networks," *IEEE Transactions on Information Forensics and Security*, vol. 4, no. 2, pp. 165–178, 2009.

[21] A. Bensoussan, M. Kantarcioglu, and S. Hoe, "A game-theoretical approach for finding optimal strategies in a botnet defense model," in *Decision and Game Theory for Security*, 2010, pp. 135–148.

[22] M. J. Osborne and A. Rubinstein, *A course in game theory*. MIT Press, 1994.

[23] Y. Chen and R. Sion, "To cloud or not to cloud? musings on costs and viability," in *Proceedings of the 2nd ACM Symposium on Cloud Computing (SOCC)*, 2011.

[24] ANSSI. EBIOS (Expression of Needs and Identification of Security Objectives) Risk Management Method, 2010, URL: http://www.ssi.gouv.fr/IMG/pdf/EBIOS-1-GuideMethodologique-2010-01-25.pdf.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TIFS.2016.2549002, IEEE Transactions on Information Forensics and Security

Ziad Ismail is currently a postdoctoral research fellow in the Department of Computer Science and Networks, LTCI, CNRS, Télécom ParisTech, Université Paris-Saclay, Paris. He received his B.E. degree in Telecommunication Engineering from ESIB, Lebanon in 2009 and the Engineer Diploma, PhD from Télécom ParisTech, Paris in 2012 and 2016, respectively. His main research interests include security policies optimization and enforcement, modelling of security strategies, and game theory.

Christophe Kiennert is currently conducting postdoctoral research at Télécom SudParis, Institut Mines-Télécom, France. He graduated from Télécom ParisTech in 2008, and received his PhD degree in Network Security in 2012. His research interests cover security and privacy of identity management in networks, security of Cloud Computing, as well as game theory and optimization models for decision making in network security.

Jean Leneutre received his PhD in Computer Science from Télécom ParisTech, Paris, in 1998. He is currently an associate professor in the Department of Computer Science and Networks, LTCI, CNRS, Télécom ParisTech, Université Paris-Saclay, and is co-leader of the Network and Information Security team. His main research interests include the definition of security models and design of security mechanisms for complex systems and networks.

Lin Chen received his B.E. degree in Radio Engineering from Southeast University, China in 2002 and the Engineer Diploma, PhD from Télécom ParisTech, Paris in 2005 and 2008, respectively. He also holds a M.S. degree of Networking from the University of Paris 6. He currently works as associate professor in the department of computer science of the University of Paris-Sud XI. His main research interests include modeling and control for wireless networks, security and cooperation enforcement in wireless networks and game theory.