





Article

A Game-Theoretic Approach to Information-Flow Control via Protocol Composition

Mário S. Alvim¹ , Konstantinos Chatzikokolakis^{2,3} , Yusuke Kawamoto⁴  and Catuscia Palamidessi^{5,2} 

¹ Universidade Federal de Minas Gerais, Brazil;

² École Polytechnique, France;

³ CNRS, France;

⁴ AIST, Japan;

⁵ INRIA, France;

Version May 18, 2018 accepted to Entropy

Abstract: In the inference attacks studied in Quantitative Information Flow (QIF), the attacker typically tries to interfere with the system in the attempt to increase its leakage of secret information. The defender, on the other hand, typically tries to decrease leakage by introducing some controlled noise. This noise introduction can be modeled as a type of protocol composition, i.e., a probabilistic choice among different protocols, and its effect on the amount of leakage depends heavily on whether or not this choice is visible to the attacker. In this work, we consider operators for modeling visible and hidden choice in protocol composition, and we study their algebraic properties. We then formalize the interplay between defender and attacker in a game-theoretic framework adapted to the specific issues of QIF, where the payoff is information leakage. We consider various kinds of leakage games, depending on whether players act simultaneously or sequentially, and on whether or not the choices of the defender are visible to the attacker. In the case of sequential games, the choice of the second player is generally a function of the choice of the first player, and his/her probabilistic choice can be either over the possible functions (mixed strategy) or it can be on the result of the function (behavioral strategy). We show that when the attacker moves first in a sequential game with a hidden choice, then behavioral strategies are more advantageous for the defender than mixed strategies. This contrasts with the standard game theory, where the two types of strategies are equivalent. Finally, we establish a hierarchy of these games in terms of their information leakage and provide methods for finding optimal strategies (at the points of equilibrium) for both attacker and defender in the various cases.

Keywords: information leakage; quantitative information flow; game theory; algebraic properties

1. Introduction

A fundamental problem in computer security is the leakage of sensitive information due to *correlation* of secret values with *observables*—i.e., any information accessible to the attacker, such as, for instance, the system's outputs or execution time. The typical defense consists in reducing this correlation, which can be done in, essentially, two ways. The first, applicable when the correspondence secret-observable is deterministic, consists in coarsening the equivalence classes of secrets that give rise to the same observables. This can be achieved with post-processing, i.e., sequentially composing the original system with a program that removes information from observables. For example, a typical attack on encrypted web traffic consists in the analysis of the packets' length, and a typical defense consists in padding extra bits so to diminish the length variety [1].

The second kind of defense, on which we focus in this work, consists in adding controlled noise to the observables produced by the system. This can be usually seen as a composition of different protocols via probabilistic choice.

Example 1 (Differential privacy). Consider a counting query f , namely a function that, applied to a dataset x , returns the number of individuals in x that satisfy a given property. A way to implement differential privacy [2] is to add geometrical noise to the result of f , so to obtain a probability distribution P on integers of the form $P(z) = c e^{-|z-f(x)|}$, where c is a normalization factor. The resulting mechanism can be interpreted as a probabilistic choice on protocols of the form $f(x), f(x) + 1, f(x) + 2, \dots, f(x) - 1, f(x) - 2, \dots$, where the probability assigned to $f(x) + n$ and to $f(x) - n$ decreases exponentially with n .

Example 2 (Dining cryptographers). Consider two agents running the dining cryptographers protocol [3], which consists in tossing a fair binary coin and then declaring the exclusive or \oplus of their secret value x and the result of the coin. The protocol can be thought as the fair probabilistic choice of two protocols, one consisting simply of declaring x , and the other declaring $x \oplus 1$.

Most of the work in the literature of quantitative information flow (QIF) considers passive attacks, in which the attacker only observes the system. Notable exceptions are the works [4–6], which consider attackers who interact with and influence the system, possibly in an adaptive way, with the purpose of maximizing the leakage of information.

Example 3 (CRIME attack). Compression Ratio Info-leak Made Easy (CRIME) [7] is a security exploit against secret web cookies over connections using the HTTPS and SPDY protocols and data compression. The idea is that the attacker can inject some content a in the communication of the secret x from the target site to the server. The server then compresses and encrypts the data, including both a and x , and sends back the result. By observing the length of the result, the attacker can then infer information about x . To mitigate the leakage, one possible defense would consist in transmitting, along with x , also an encryption method f selected randomly from a set F . Again, the resulting protocol can be seen as a composition, using probabilistic choice, of the protocols in the set F .

Example 4 (Timing side-channels). Consider a password-checker, or any similar system in which the user authenticates himself by entering a secret that is checked by the system. An adversary does not know the real secret, of course, but a timing side-channel could reveal the part (eg. which bit) of the secret in which the adversary's input fails. By repeating the process with different inputs, the adversary might be able to fully retrieve the secret. A possible counter measure is to make the side channel noisy, by randomizing the order in which the secret's bit are checked against the user input. This example is studied in detail in Section 7.

In all examples above the main use of the probabilistic choice is to obfuscate the relation between secrets and observables, thus reducing their correlation—and, hence, the information leakage. To achieve this goal, it is essential that the attacker never comes to know the result of the choice. In the CRIME example, however, if f and a are chosen independently, then (in general) it is still better to choose f probabilistically, even if the attacker will come to know, afterwards, the choice of f . In fact, this is true also for the attacker: his best strategies (in general) are to choose a according to some probability distribution. Indeed, suppose that $F = \{f_1, f_2\}$ are the defender's choices and $A = \{a_1, a_2\}$ are the attacker's, and that $f_1(\cdot, a_1)$ leaks more than $f_1(\cdot, a_2)$, while $f_2(\cdot, a_1)$ leaks less than $f_2(\cdot, a_2)$. This is a scenario like *the matching pennies* in game theory: if one player selects an action deterministically, the other player may exploit this choice and get an advantage. For each player the optimal strategy is to play probabilistically, using a distribution that maximizes his own gain for all possible actions of the attacker. In zero-sum games, in which the gain of one player coincides with the loss of the other, the optimal pair of distributions always exists, and it is called *saddle point*. It also coincides with the *Nash equilibrium*, which is defined as the point in which neither of the two players gets any advantage in changing unilaterally his strategy.

Motivated by these examples, this paper investigates the two kinds of choice, visible and hidden (to the attacker), in a game-theoretic setting. Looking at them as language operators, we study their algebraic properties, which will help reason about their behavior in games. We consider zero-sum games, in which the gain (for the attacker) is represented by the leakage. While for visible choice it is appropriate to use the “classic” game-theoretic framework, for hidden choice we need to adopt the more general framework of the *information leakage games* proposed in [6]. This happens because, in contrast with standard game theory, in games with hidden choice the payoff of a mixed strategy is a convex function of the distribution on the defender’s pure actions, rather than simply the expected value of their utilities. We will consider both simultaneous games—in which each player chooses independently—and sequential games—in which one player chooses his action first. We aim at comparing all these situations, and at identifying the precise advantage of the hidden choice over the visible one.

To measure leakage we use the well-known information-theoretic model. A central notion in this model is that of *entropy*, but here we use its converse, *vulnerability*, which represents the magnitude of the threat. In order to derive results as general as possible, we adopt the very comprehensive notion of vulnerability as any convex and continuous function, as used in [4] and [8]. This notion has been shown [8] to be, in a precise sense, the most general information measures w.r.t. a set of fundamental information-theoretic axioms. Our results, hence, apply to all information measures that respect such fundamental principles, including the widely adopted measures of *Bayes vulnerability* (aka min-vulnerability, aka (the converse of) Bayes risk) [9,10], *Shannon entropy* [11], *guessing entropy* [12], and *g-vulnerability* [13].

The main contributions of this paper are:

- We present a general framework for reasoning about information leakage in a game-theoretic setting, extending the notion of information leakage games proposed in [6] to both simultaneous and sequential games, with either hidden or visible choice.
- We present a rigorous compositional way, using visible and hidden choice operators, for representing attacker’s and defender’s actions in information leakage games. In particular, we study the algebraic properties of visible and hidden choice on channels, and compare the two kinds of choice with respect to the capability of reducing leakage, in presence of an adaptive attacker.
- We provide a taxonomy of the various scenarios (simultaneous and sequential) showing when randomization is necessary, for either attacker or defender, to achieve optimality. Although it is well-known in information flow that the defender’s best strategy is usually randomized, only recently it has been shown that when defender and attacker act simultaneously, the attacker’s optimal strategy also requires randomization [6].
- We compare the vulnerability of the leakage games for these various scenarios and establish a hierarchy of leakage games based on the order between the value of the leakage in the Nash equilibrium. Furthermore, we show that when the attacker moves first in a sequential game with hidden choice, the behavioral strategies (where the defender chooses his probabilistic distribution after he has seen the choice of the attacker) are more advantageous for the defender than the mixed strategies (where the defender chooses the probabilistic distribution over his possible functional dependency on the choice of the attacker). This contrast with the standard game theory, where the two types of strategies are equivalent. Another difference is that in our attacker-first sequential games there may not exist Nash equilibria with deterministic strategies for the defender (although the defender has full visibility of the attacker’s moves).
- We use our framework in a detailed case study of a password-checking protocol. A naive program, which checks the password bit by bit and stops when it finds a mismatch, is clearly very insecure, because it reveals at each attempt (via a timing side-channel), the maximum correct prefix. On the other hand, if we continue checking until the end of the string (time padding), the

program becomes very inefficient. We show that, by using probabilistic choice instead, we can obtain a good trade-off between security and efficiency.

Plan of the paper.

The remaining of the paper is organized as follows. In Section 2 we review some basic notions of game theory and quantitative information flow. In Section 3 we introduce our running example. In Section 4 we define the visible and hidden choice operators and demonstrate their algebraic properties. In Section 5, the core of the paper, we examine various scenarios for leakage games. In Section 6 we compare the vulnerability of the various leakage games, and establish a hierarchy among those games. In Section 7 we show an application of our framework to a password checker. In Section 8 we discuss related work and, finally, in Section 9 we conclude.

A preliminary version of this paper appeared in [14]. One difference with respect to [14] is that in the present paper we consider both behavioral and mixed strategies in the sequential games, while in [14] we only considered the latter. We also show that the two kinds of strategies are not equivalent in our context (Example 10: the optimal strategy profile yields a different payoff depending on whether the defender adopts mixed strategies or behavioral ones). In light of this difference, we provide new results that concern behavioral strategies, and in particular:

- Theorem 3, which concerns the defender's behavioral strategies in the defender-first game with visible choice (Game II),
- the second half of Theorem 6, which deals with the adversary's behavioral strategies in the attacker-first game with hidden choice (Game VI).

Furthermore, in this paper we define formally all concepts, and provide all the proofs. In particular, we provide a precise formulation of the comparison among games with visible/hidden choices (Propositions 4 and 5, Corollaries 3, 4, and 5) in Section 6. Finally, in Section 7 we provide a new result, expressed by Theorem 7, regarding the optimal strategies for the defender in the presence of a uniform prior on passwords.

2. Preliminaries

In this section we review some basic notions from game theory and quantitative information flow. We use the following notation: Given a set \mathcal{I} , we denote by $\mathbb{D}\mathcal{I}$ the set of all probability distributions over \mathcal{I} . Given $\mu \in \mathbb{D}\mathcal{I}$, its support $\text{supp}(\mu) \stackrel{\text{def}}{=} \{i \in \mathcal{I} : \mu(i) > 0\}$ is the set of its elements with positive probability. We use $i \leftarrow \mu$ to indicate that a value $i \in \mathcal{I}$ is sampled from a distribution μ on \mathcal{I} . A set $\mathcal{S} \subseteq \mathbb{R}^n$ is *convex* if $ts_0 + (1-t)s_1 \in \mathcal{S}$ for all $s_0, s_1 \in \mathcal{S}$ and $t \in [0, 1]$. For such a set, a function $f : \mathcal{S} \rightarrow \mathbb{R}$ is *convex* if $f(ts_0 + (1-t)s_1) \leq tf(s_0) + (1-t)f(s_1)$ for all $s_0, s_1 \in \mathcal{S}, t \in [0, 1]$, and *concave* if $-f$ is convex.

2.1. Basic concepts from game theory

2.1.1. Two-player games

Two-player games are a model for reasoning about the behavior of two players. In a game, each player has at its disposal a set of *actions* that he can perform, and he obtains some gain or loss depending on the actions chosen by both players. Gains and losses are defined using a real-valued *payoff function*. Each player is assumed to be *rational*, i.e., his choice is driven by the attempt to maximize his own expected payoff. We also assume that the set of possible actions and the payoff functions of both players are *common knowledge*.

In this paper we only consider *finite games*, in which the set of actions available to the players are finite, and which are also *zero-sum games*, so the payoff of one player is the loss of the other. Next we introduce an important distinction between *simultaneous* and *sequential* games. In the following, we will call the two players *defender* and *attacker*.

2.1.2. Simultaneous games

In a simultaneous game, each player chooses his action without knowing the action chosen by the other. The term “simultaneous” here does not mean that the players’ actions are chosen at the same time, but only that they are chosen independently. Formally, such a game is defined as a tuple¹ $(\mathcal{D}, \mathcal{A}, u_d, u_a)$, where \mathcal{D} is a nonempty set of *defender’s actions*, \mathcal{A} is a nonempty set of *attacker’s actions*, $u_d : \mathcal{D} \times \mathcal{A} \rightarrow \mathbb{R}$ is the *defender’s payoff function*, and $u_a : \mathcal{D} \times \mathcal{A} \rightarrow \mathbb{R}$ is the *attacker’s payoff function*.

Each player may choose an action deterministically or probabilistically. A *pure strategy* of the defender (resp. attacker) is a deterministic choice of an action, i.e., an element $d \in \mathcal{D}$ (resp. $a \in \mathcal{A}$). A pair (d, a) is called *pure strategy profile*, and $u_d(d, a)$, $u_a(d, a)$ represent the defender’s and the attacker’s payoffs, respectively. A *mixed strategy* of the defender (resp. attacker) is a probabilistic choice of an action, defined as a probability distribution $\delta \in \mathbb{D}\mathcal{D}$ (resp. $\alpha \in \mathbb{D}\mathcal{A}$). A pair (δ, α) is called *mixed strategy profile*. The defender’s and the attacker’s *expected payoff functions* for mixed strategies are defined, respectively, as:

$$U_d(\delta, \alpha) \stackrel{\text{def}}{=} \mathbb{E}_{\substack{d \leftarrow \delta \\ a \leftarrow \alpha}} u_d(d, a) = \sum_{\substack{d \in \mathcal{D} \\ a \in \mathcal{A}}} \delta(d)\alpha(a)u_d(d, a) \text{ and}$$

$$U_a(\delta, \alpha) \stackrel{\text{def}}{=} \mathbb{E}_{\substack{d \leftarrow \delta \\ a \leftarrow \alpha}} u_a(d, a) = \sum_{\substack{d \in \mathcal{D} \\ a \in \mathcal{A}}} \delta(d)\alpha(a)u_a(d, a).$$

A defender’s mixed strategy $\delta \in \mathbb{D}\mathcal{D}$ is a *best response* to an attacker’s mixed strategy $\alpha \in \mathbb{D}\mathcal{A}$ if $U_d(\delta, \alpha) = \max_{\delta' \in \mathbb{D}\mathcal{D}} U_d(\delta', \alpha)$. Symmetrically, $\alpha \in \mathbb{D}\mathcal{A}$ is a *best response* to $\delta \in \mathbb{D}\mathcal{D}$ if $U_a(\delta, \alpha) = \max_{\alpha' \in \mathbb{D}\mathcal{A}} U_a(\delta, \alpha')$. A *mixed-strategy Nash equilibrium* is a profile (δ^*, α^*) such that δ^* is a best response to α^* and vice versa. This means that in a Nash equilibrium, no unilateral deviation by any single player provides better payoff to that player. If δ^* and α^* are point distributions concentrated on some $d^* \in \mathcal{D}$ and $a^* \in \mathcal{A}$ respectively, then (δ^*, α^*) is a *pure-strategy Nash equilibrium*, and will be denoted by (d^*, a^*) . While not all games have a pure strategy Nash equilibrium, every finite game has a mixed strategy Nash equilibrium.

2.1.3. Sequential games

In a sequential game players may take turns in choosing their actions. In this paper, we only consider the case in which each player moves only once, in such a way that one of the players (*the leader*) chooses his action first, and commits to it, before the other player (*the follower*) makes his choice. The follower may have total knowledge of the choice made by the leader, or only partial. We refer to the two scenarios by the terms *perfect* and *imperfect information*, respectively. Another distinction is the kind of randomization used by the players, namely whether the follower chooses probabilistically his action after he knows (partially or totally) the move of the leader, or whether he chooses at the beginning of the game a probabilistic distribution on (deterministic) strategies that depend on the (partial or total) knowledge of the move of the leader. In the first case the strategies are called *behavioral*, in the second case *mixed*.

We now give the precise definitions assuming that the leader is the defender. The definitions for the case in which the leader is the attacker are analogous.

A *defender-first sequential game with perfect information* is a tuple $(\mathcal{D}, \mathcal{D} \rightarrow \mathcal{A}, u_d, u_a)$ where \mathcal{D} , \mathcal{A} , u_d and u_a are defined as in simultaneous games: The choice of an action $d \in \mathcal{D}$ represents a pure strategy of the defender. As for the attacker, his choice $a \in \mathcal{A}$ depends functionally on the prior choice d of the defender, and for this reason the pure strategies of the attacker are functions $s_a : \mathcal{D} \rightarrow \mathcal{A}$. As for the

¹ Following the convention of *security games*, we set the first player to be the defender.

probabilistic strategies, those of the defender are defined as in simultaneous games: namely they are distributions $\delta \in \mathbb{D}\mathcal{D}$. On the other hand, the attacker's probabilistic strategies can be defined in two different ways: In the *behavioral* case, an attacker's probabilistic strategy is a function $\phi_a : \mathcal{D} \rightarrow \mathbb{D}(\mathcal{A})$. Namely, the attacker chooses a distribution on its actions after he sees the move of the defender. In the *mixed* case, an attacker's probabilistic strategy is a probability distribution $\sigma_a \in \mathbb{D}(\mathcal{D} \rightarrow \mathcal{A})$. Namely, the attacker chooses *a priori* a distribution on pure strategies. The defender's and the attacker's *expected payoff functions* for mixed strategies are defined, respectively, as

$$\begin{aligned} \text{Behavioral case} & \left\{ \begin{aligned} U_d(\delta, \phi_a) & \stackrel{\text{def}}{=} \mathbb{E}_{d \leftarrow \delta} \mathbb{E}_{a \leftarrow \phi_a(d)} u_d(d, a) = \sum_{d \in \mathcal{D}} \delta(d) \sum_{a \in \mathcal{A}} \phi_a(d)(a) u_d(d, a) \\ U_a(\delta, \phi_a) & \stackrel{\text{def}}{=} \mathbb{E}_{d \leftarrow \delta} \mathbb{E}_{a \leftarrow \phi_a(d)} u_a(d, a) = \sum_{d \in \mathcal{D}} \delta(d) \sum_{a \in \mathcal{A}} \phi_a(d)(a) u_a(d, a) \end{aligned} \right. \\ \\ \text{Mixed case} & \left\{ \begin{aligned} U_d(\delta, \sigma_a) & \stackrel{\text{def}}{=} \mathbb{E}_{d \leftarrow \delta} \mathbb{E}_{s_a \leftarrow \sigma_a} u_d(d, s_a(d)) = \sum_{\substack{d \in \mathcal{D} \\ s_a : \mathcal{D} \rightarrow \mathcal{A}}} \delta(d) \sigma_a(s_a) u_d(d, s_a(d)) \\ U_a(\delta, \sigma_a) & \stackrel{\text{def}}{=} \mathbb{E}_{d \leftarrow \delta} \mathbb{E}_{s_a \leftarrow \sigma_a} u_a(d, s_a(d)) = \sum_{\substack{d \in \mathcal{D} \\ s_a : \mathcal{D} \rightarrow \mathcal{A}}} \delta(d) \sigma_a(s_a) u_a(d, s_a(d)) \end{aligned} \right. \end{aligned}$$

The case of imperfect information is typically formalized by assuming an *indistinguishability (equivalence) relation* over the actions chosen by the leader, representing a scenario in which the follower cannot distinguish between the actions belonging to the same equivalence class. The pure strategies of the followers, therefore, are functions from the set of the equivalence classes on the actions of the leader to his own actions. Formally, a *defender-first sequential game with imperfect information* is a tuple $(\mathcal{D}, K_a \rightarrow \mathcal{A}, u_d, u_a)$ where \mathcal{D} , \mathcal{A} , u_d and u_a are defined as in simultaneous games, and K_a is a partition of \mathcal{D} . The *expected payoff functions* are defined as before, except that now the argument of ϕ_a and s_a is the equivalence class of d . Note that in the case in which all defender's actions are indistinguishable from each other at the eyes of the attacker (*totally imperfect information*), we have $K_a = \{\mathcal{D}\}$ and the expected payoff functions coincide with those of the simultaneous games. In contrast, in the games in which all defender's actions are distinguishable from the view of the attacker (*perfect information*), we have $K_a = \{\{d\} \mid d \in \mathcal{D}\}$.

In the standard game theory, under the assumption of *perfect recall* (i.e., the players never forget what they have learned), behavioral and mixed strategies are equivalent, in the sense that for any behavioral strategy there is a mixed strategy that yields the same payoff, and vice versa. This is true for both cases of perfect and imperfect information, see [15], Chapter 11.4. In our leakage games, however, this equivalence does not hold anymore, as it will be shown in Sections 5 and 6.

2.1.4. Zero-sum games and Minimax Theorem

A game $(\mathcal{D}, \mathcal{A}, u_d, u_a)$ is *zero-sum* if for any $d \in \mathcal{D}$ and any $a \in \mathcal{A}$, the defender's loss is equivalent to the attacker's gain, i.e., $u_d(d, a) = -u_a(d, a)$. For brevity, in zero-sum games we denote by u the attacker's payoff function u_a , and by U the attacker's expected payoff U_a .² Consequently, the goal of the defender is to minimize U , and the goal of the attacker is to maximize it.

In simultaneous zero-sum games the Nash equilibrium corresponds to the solution of the *minimax* problem (or equivalently, the *maximin* problem), namely, the strategy profile (δ^*, α^*) such that $U(\delta^*, \alpha^*) = \min_{\delta} \max_{\alpha} U(\delta, \alpha)$. The von Neumann's minimax theorem, in fact, ensures that such solution (which always exists) is stable.

² Conventionally in game theory the payoff u is set to be that of the first player, but we prefer to look at the payoff from the point of view of the attacker to be in line with the definition of payoff as vulnerability.

Theorem 1 (von Neumann's minimax theorem). Let $\mathcal{X} \subset \mathbb{R}^m$ and $\mathcal{Y} \subset \mathbb{R}^n$ be compact convex sets, and $U : \mathcal{X} \times \mathcal{Y} \rightarrow \mathbb{R}$ be a continuous function such that $U(x, y)$ is a convex function in $x \in \mathcal{X}$ and a concave function in $y \in \mathcal{Y}$. Then

$$\min_{x \in \mathcal{X}} \max_{y \in \mathcal{Y}} U(x, y) = \max_{y \in \mathcal{Y}} \min_{x \in \mathcal{X}} U(x, y).$$

A related property is that, under the conditions of Theorem 1, there exists a *saddle point* (x^*, y^*) s.t., for all $x \in \mathcal{X}$ and $y \in \mathcal{Y}$: $U(x^*, y) \leq U(x^*, y^*) \leq U(x, y^*)$.

The solution of the minimax problem can be obtained by using convex optimization techniques. In case $U(x, y)$ is affine in x and in y , we can also use linear optimization.

In case \mathcal{D} and \mathcal{A} contain two elements each, there is a closed form for the solution. Let $\mathcal{D} = \{d_0, d_1\}$ and $\mathcal{A} = \{a_0, a_1\}$ respectively. Let u_{ij} be the payoff of the defender on d_i, a_j . Then the Nash equilibrium (δ^*, α^*) is given by:

$$\delta^*(d_0) = \frac{u_{11} - u_{10}}{u_{00} - u_{01} - u_{10} + u_{11}} \quad \alpha^*(a_0) = \frac{u_{11} - u_{01}}{u_{00} - u_{01} - u_{10} + u_{11}} \quad (1)$$

if these values are in $[0, 1]$. Note that, since there are only two elements, the strategy δ^* is completely specified by its value in d_0 , and analogously for α^* .

2.2. Quantitative information flow

Finally, we briefly review the standard framework of quantitative information flow, which is concerned with measuring the amount of information leakage in a (computational) system.

2.2.1. Secrets and vulnerability

A *secret* is some piece of sensitive information the defender wants to protect, such as a user's password, social security number, or current location. The attacker usually only has some partial knowledge about the value of a secret, represented as a probability distribution on secrets called a *prior*. We denote by \mathcal{X} the set of possible secrets, and we typically use π to denote a prior belonging to the set $\mathbb{D}\mathcal{X}$ of probability distributions over \mathcal{X} .

The *vulnerability* of a secret is a measure of the payoff that it represents for the attacker. In this paper we consider a very general notion of vulnerability, following [8], and we define a vulnerability \mathbb{V} to be any continuous and convex function of type $\mathbb{D}\mathcal{X} \rightarrow \mathbb{R}$. It has been shown in [8] that these functions coincide with the set of g -vulnerabilities, and are, in a precise sense, the most general information measures w.r.t. a set of fundamental information-theoretic axioms.³ This notion, hence, subsumes all information measures that respect such fundamental principles, including the widely adopted measures of *Bayes vulnerability* (aka min-vulnerability, aka (the converse of) Bayes risk) [9,10], *Shannon entropy* [11], *guessing entropy* [12], and g -*vulnerability* [13].

2.2.2. Channels, posterior vulnerability, and leakage

Computational systems can be modeled as information theoretic channels. A *channel* $C : \mathcal{X} \times \mathcal{Y} \rightarrow \mathbb{R}$ is a function in which \mathcal{X} is a set of *input values*, \mathcal{Y} is a set of *output values*, and $C(x, y)$ represents the conditional probability of the channel producing output $y \in \mathcal{Y}$ when input $x \in \mathcal{X}$ is provided. Every channel C satisfies $0 \leq C(x, y) \leq 1$ for all $x \in \mathcal{X}$ and $y \in \mathcal{Y}$, and $\sum_{y \in \mathcal{Y}} C(x, y) = 1$ for all $x \in \mathcal{X}$.

A distribution $\pi \in \mathbb{D}\mathcal{X}$ and a channel C with inputs \mathcal{X} and outputs \mathcal{Y} induce a joint distribution $p(x, y) = \pi(x)C(x, y)$ on $\mathcal{X} \times \mathcal{Y}$, producing joint random variables X, Y with marginal probabilities $p(x) = \sum_y p(x, y)$ and $p(y) = \sum_x p(x, y)$, and conditional probabilities $p(x|y) = p(x, y)/p(y)$ if $p(y) \neq 0$.

³ More precisely, if posterior vulnerability is defined as the expectation of the vulnerability of posterior distributions, the measure respects the fundamental information-theoretic properties of *data-processing inequality* (i.e., that post-processing can never increase information, but only destroy it), and of *non-negativity of leakage* (i.e., that by observing the output of a channel an actor cannot, in average, lose information) if, and only if, vulnerability is convex.

<p><u>Program 0</u></p> <p>High Input: $x \in \{0, 1\}$ Low Input: $a \in \{0, 1\}$ Output: $y \in \{0, 1\}$ $y = x \cdot a$ return y</p>	<p><u>Program 1</u></p> <p>High Input: $x \in \{0, 1\}$ Low Input: $a \in \{0, 1\}$ Output: $y \in \{0, 1\}$ $c \leftarrow$ flip coin with bias $a/3$ if $c = \text{heads}$ $\{y = x\}$ else $\{y = \bar{x}\}$ return y</p>
---	---

Figure 1. Alternative programs for the running example.

For a given y (s.t. $p(y) \neq 0$), the conditional probabilities $p(x|y)$ for each $x \in \mathcal{X}$ form the *posterior distribution* $p_{X|y}$.

A channel C in which \mathcal{X} is a set of secret values and \mathcal{Y} is a set of observable values produced by a system can be used to model computations on secrets. Assuming the attacker has prior knowledge π about the secret value, knows how a channel C works, and can observe the channel's outputs, the effect of the channel is to update the attacker's knowledge from π to a collection of posteriors $p_{X|y}$, each occurring with probability $p(y)$.

Given a vulnerability \mathbb{V} , a prior π , and a channel C , the *posterior vulnerability* $\mathbb{V}[\pi, C]$ is the vulnerability of the secret after the attacker has observed the output of the channel C . Formally: $\mathbb{V}[\pi, C] \stackrel{\text{def}}{=} \sum_{y \in \mathcal{Y}} p(y) \mathbb{V}[p_{X|y}]$.

Consider, for instance, the example of the password-checker with a timing side-channel from the introduction (Example 4, also discussed in detail in Section 7). Here the set of secrets \mathcal{X} consists of all possible passwords (say, all strings of n bits), and a natural vulnerability function is Bayes-vulnerability, given by $\mathbb{V}(\pi) = \max_{x \in \mathcal{X}} \pi(x)$. This function expresses the adversary's probability of *guessing correctly* the password in one try; assuming that the passwords are chosen uniformly, i.e. π is uniform, any guess would be correct with probability 2^{-n} , giving $\mathbb{V}(\pi) = 2^{-n}$. Now, imagine that the timing side-channel reveals that the adversary's input failed *on the first bit*. The adversary now knows the first bit of the password (say 0), hence the posterior $p_{X|y}$ assigns probability 0 to all passwords with first bit 1, and probability $2^{-(n-1)}$ to all passwords with first bit 0. This happens for all possible posteriors, giving posterior vulnerability $\mathbb{V}[\pi, C] = 2^{-(n-1)}$ (2 times greater than the prior \mathbb{V}).

It is known from the literature [8] that the posterior vulnerability is a convex function of π . Namely, for any channel C , any family of distributions $\{\pi_i\}$, and any set of convex coefficients $\{c_i\}$, we have:

$$\mathbb{V}\left[\sum_i c_i \pi_i, C\right] \leq \sum_i c_i \mathbb{V}[\pi_i, C]$$

The (*information*) *leakage* of channel C under prior π is a comparison between the vulnerability of the secret before the system was run—called *prior vulnerability*—and the posterior vulnerability of the secret. Leakage reflects by how much the observation of the system's outputs increases the attacker's information about the secret. It can be defined either *additively* ($\mathbb{V}[\pi, C] - \mathbb{V}[\pi]$), or *multiplicatively* ($\mathbb{V}[\pi, C]/\mathbb{V}[\pi]$). In the password-checker example, the additive leakage is $2^{-(n-1)} - 2^{-n} = 2^{-n}$ and the multiplicative leakage is $2^{-(n-1)}/2^{-n} = 2$.

3. An illustrative example

We introduce an example which will serve as running example through the paper. Although admittedly contrived, this example is simple and yet produces different leakage measures for all different combinations of visible/hidden choice and simultaneous/sequential games, thus providing a way to compare all different scenarios we are interested in.

Table 1. Channel matrices and payoff table for running example.

(a) The four channels C_{da} for $d, a \in \{0, 1\}$

C_{00}	$y = 0$	$y = 1$
$x = 0$	1	0
$x = 1$	1	0

C_{01}	$y = 0$	$y = 1$
$x = 0$	1	0
$x = 1$	0	1

C_{10}	$y = 0$	$y = 1$
$x = 0$	0	1
$x = 1$	1	0

C_{11}	$y = 0$	$y = 1$
$x = 0$	$1/3$	$2/3$
$x = 1$	$2/3$	$1/3$

(b) Bayes vulnerability of each channel C_{da} .

\mathbb{V}	$a = 0$	$a = 1$
$d = 0$	$1/2$	1
$d = 1$	1	$2/3$

Consider that a binary secret must be processed by a program. As usual, a defender wants to protect the secret value, whereas an attacker wants to infer it by observing the system's output. Assume the defender can choose which among two alternative versions of the program to run. Both programs take the secret value x as high input, and a binary low input a whose value is chosen by the attacker. They both return the output in a low variable y .⁴ Program 0 returns the binary product of x and a , whereas Program 1 flips a coin with bias $a/3$ (i.e., a coin which returns heads with probability $a/3$) and returns x if the result is heads, and the complement \bar{x} of x otherwise. The two programs are represented in Figure 1.

The combined choices of the defender's and of the attacker's determine how the system behaves. Let $\mathcal{D} = \{0, 1\}$ represent the set of the defender's choices—i.e., the index of the program to use—, and $\mathcal{A} = \{0, 1\}$ represent the set of the attacker's choices—i.e., the value of the low input a . We shall refer to the elements of \mathcal{D} and \mathcal{A} as *actions*. For each possible combination of actions $d \in \mathcal{D}$ and $a \in \mathcal{A}$, we can construct a channel C_{da} modeling how the resulting system behaves. Each channel C_{da} is a function of type $\mathcal{X} \times \mathcal{Y} \rightarrow \mathbb{R}$, where $\mathcal{X} = \{0, 1\}$ is the set of possible high input values for the system, and $\mathcal{Y} = \{0, 1\}$ is the set of possible output values from the system. Intuitively, each channel provides the probability that the system (which was fixed by the defender) produces output $y \in \mathcal{Y}$ given that the high input is $x \in \mathcal{X}$ (and that the low input was fixed by the attacker). The four possible channels are depicted in Table 1a.

Note that channel C_{00} does not leak any information about the input x (i.e., it is *non-interferent*), whereas channels C_{01} and C_{10} completely reveal x . Channel C_{11} is an intermediate case: it leaks some information about x , but not all.

We want to investigate how the defender's and the attacker's choices influence the leakage of the system. For that we can just consider the (simpler) notion of posterior vulnerability, since in order to make the comparison fair we need to assume that the prior is always the same in the various scenarios, and this implies that the leakage is in a one-to-one correspondence with the posterior vulnerability (this happens for both additive and multiplicative leakage).

For this example, assume we are interested in Bayes vulnerability [9,10], defined as $\mathbb{V}(\pi) = \max_x \pi(x)$ for every $\pi \in \mathbb{D}\mathcal{X}$. Assume for simplicity that the prior is the uniform prior π_u . In this case we know from [16] that the posterior Bayes vulnerability of a channel is the sum of the greatest elements of each column, divided by the total number of inputs. Table 1b provides the Bayes vulnerability $\mathbb{V}_{da} \stackrel{\text{def}}{=} \mathbb{V}[\pi_u, C_{da}]$ of each channel considered above.

Naturally, the attacker aims at maximizing the vulnerability of the system, while the defender tries to minimize it. The resulting vulnerability will depend on various factors, in particular on whether the two players make their choice *simultaneously* (i.e., without knowing the choice of the opponent) or *sequentially*. Clearly, if the choice of a player who moves first is known by an opponent who moves second, the opponent will be in advantage. In the above example, for instance, if the defender knows

⁴ We adopt the usual convention in QIF of referring to secret variables, inputs and outputs in programs as *high*, and to their observable counterparts as *low*.

the choice a of the attacker, the most convenient choice for him is to set $d = a$, and the vulnerability will be at most $2/3$. Vice versa, if the attacker knows the choice d of the defender, the most convenient choice for him is to set $a \neq d$. The vulnerability in this case will be 1.

Things become more complicated when players make choices simultaneously. None of the pure choices of d and a are the best for the corresponding player, because the vulnerability of the system depends also on the (unknown) choice of the other player. Yet there is a strategy leading to the best possible situation for both players (the *Nash equilibrium*), but it is mixed (i.e., probabilistic), in that the players randomize their choices according to some precise distribution.

Another factor that affects vulnerability is whether or not the defender's choice is known to the attacker at the moment in which he observes the output of the channel. Obviously, this corresponds to whether or not the attacker knows what channel he is observing. Both cases are plausible: naturally the defender has all the interest in keeping his choice (and, hence, the channel used) secret, since then the attack will be less effective (i.e., leakage will be smaller). On the other hand, the attacker may be able to identify the channel used anyway, for instance because the two programs have different running times. We will call these two cases *hidden* and *visible* choice, respectively.

It is possible to model players' strategies, as well as hidden and visible choices, as operations on channels. This means that we can look at the whole system as if it were a single channel, which will turn out to be useful for some proofs of our technical results. Next section is dedicated to the definition of these operators. We will calculate the exact values for our example in Section 5.

4. Choice operators for protocol composition

In this section we define the operators of visible and hidden choice for protocol composition. These operators are formally defined on the channel matrices of the protocols, and, since channels are a particular kind of matrix, we use these matrix operations to define the operations of visible and hidden choice among channels, and to prove important properties of these channel operations.

4.1. Matrices, and their basic operators

Given two sets \mathcal{X} and \mathcal{Y} , a *matrix* is a total function of type $\mathcal{X} \times \mathcal{Y} \rightarrow \mathbb{R}$. Two matrices $M_1 : \mathcal{X}_1 \times \mathcal{Y}_1 \rightarrow \mathbb{R}$ and $M_2 : \mathcal{X}_2 \times \mathcal{Y}_2 \rightarrow \mathbb{R}$ are said to be *compatible* if $\mathcal{X}_1 = \mathcal{X}_2$. If it is also the case that $\mathcal{Y}_1 = \mathcal{Y}_2$, we say that the matrices *have the same type*. The *scalar multiplication* $r \cdot M$ between a scalar r and a matrix M is defined as usual, and so is the *summation* $(\sum_{i \in \mathcal{I}} M_i)(x, y) = M_{i_1}(x, y) + \dots + M_{i_n}(x, y)$ of a family $\{M_i\}_{i \in \mathcal{I}}$ of matrices all of a same type.

Given a family $\{M_i\}_{i \in \mathcal{I}}$ of compatible matrices s.t. each M_i has type $\mathcal{X} \times \mathcal{Y}_i \rightarrow \mathbb{R}$, their *concatenation* $\diamond_{i \in \mathcal{I}}$ is the matrix having all columns of every matrix in the family, in such a way that every column is tagged with the matrix it came from. Formally, $(\diamond_{i \in \mathcal{I}} M_i)(x, (y, j)) = M_j(x, y)$, if $y \in \mathcal{Y}_j$, and the resulting matrix has type $\mathcal{X} \times (\bigsqcup_{i \in \mathcal{I}} \mathcal{Y}_i) \rightarrow \mathbb{R}$.⁵ When the family $\{M_i\}$ has only two elements we may use the *binary* version \diamond of the concatenation operator. The following depicts the concatenation of two matrices M_1 and M_2 in tabular form.

$$\begin{array}{|c|c|c|} \hline M_1 & y_1 & y_2 \\ \hline x_1 & 1 & 2 \\ \hline x_2 & 3 & 4 \\ \hline \end{array} \diamond \begin{array}{|c|c|c|c|} \hline M_2 & y_1 & y_2 & y_3 \\ \hline x_1 & 5 & 6 & 7 \\ \hline x_2 & 8 & 9 & 10 \\ \hline \end{array} = \begin{array}{|c|c|c|c|c|c|} \hline M_1 \diamond M_2 & (y_1, 1) & (y_2, 1) & (y_1, 2) & (y_2, 2) & (y_3, 2) \\ \hline x_1 & 1 & 2 & 5 & 6 & 7 \\ \hline x_2 & 3 & 4 & 8 & 9 & 10 \\ \hline \end{array}$$

⁵ $\bigsqcup_{i \in \mathcal{I}} \mathcal{Y}_i = \mathcal{Y}_{i_1} \sqcup \mathcal{Y}_{i_2} \sqcup \dots \sqcup \mathcal{Y}_{i_n}$ denotes the *disjoint union* $\{(y, i) \mid y \in \mathcal{Y}_i, i \in \mathcal{I}\}$ of the sets $\mathcal{Y}_{i_1}, \mathcal{Y}_{i_2}, \dots, \mathcal{Y}_{i_n}$.

may use the *binary* version ${}_p\sqcup$ of the visible choice operator: $C_{i_1} {}_p\sqcup C_{i_2} = p C_{i_1} \diamond \bar{p} C_{i_2}$. The following depicts the visible choice between channels C_1 and C_3 , with probability $p=1/3$.

$$\begin{array}{|c|c|c|} \hline C_1 & y_1 & y_2 \\ \hline x_1 & 1/2 & 1/2 \\ \hline x_2 & 1/3 & 2/3 \\ \hline \end{array} \quad {}_{1/3}\sqcup \quad \begin{array}{|c|c|c|} \hline C_3 & y_1 & y_3 \\ \hline x_1 & 1/3 & 2/3 \\ \hline x_2 & 1/2 & 1/2 \\ \hline \end{array} = \begin{array}{|c|c|c|c|c|} \hline C_1 \quad {}_{1/3}\sqcup C_3 & (y_1, 1) & (y_2, 1) & (y_1, 3) & (y_3, 3) \\ \hline x_1 & 1/6 & 1/6 & 2/9 & 4/9 \\ \hline x_2 & 1/9 & 2/9 & 1/3 & 1/3 \\ \hline \end{array}$$

4.3. Properties of hidden and visible choice operators

We now prove algebraic properties of channel operators. These properties will be useful when we model a (more complex) protocol as the composition of smaller channels via hidden or visible choice.

Whereas the properties of hidden choice hold generally with equality, those of visible choice are subtler. For instance, visible choice is not idempotent, since in general $C {}_p\sqcup C \neq C$. (In fact if C has type $\mathcal{X} \times \mathcal{Y} \rightarrow \mathbb{R}$, $C {}_p\sqcup C$ has type $\mathcal{X} \times (\mathcal{Y} \sqcup \mathcal{Y}) \rightarrow \mathbb{R}$.) However, idempotency and other properties involving visible choice hold if we replace the notion of equality with the more relaxed notion of “equivalence” between channels. Intuitively, two channels are equivalent if they have the same input space and yield the same value of vulnerability for every prior and every vulnerability function.

Definition 1 (Equivalence of channels). *Two compatible channels C_1 and C_2 with domain \mathcal{X} are equivalent, denoted by $C_1 \approx C_2$, if for every prior $\pi \in \mathbb{D}\mathcal{X}$ and every posterior vulnerability \mathbb{V} we have $\mathbb{V}[\pi, C_1] = \mathbb{V}[\pi, C_2]$.*

Two equivalent channels are indistinguishable from the point of view of information leakage, and in most cases we can just identify them. Indeed, nowadays there is a tendency to use *abstract channels* [8,17], which capture exactly the important behavior with respect to any form of leakage. In this paper, however, we cannot use abstract channels because the hidden choice operator needs a concrete representation in order to be defined unambiguously.

The first properties we prove regard idempotency of operators, which can be used to simplify the representation of some protocols.

Proposition 4.3 (Idempotency). *Given a family $\{C_i\}_{i \in \mathcal{I}}$ of channels s.t. $C_i = C$ for all $i \in \mathcal{I}$, and a distribution μ on \mathcal{I} , then: (a) $\sum_{i \leftarrow \mu} C_i = C$; and (b) $\bigsqcup_{i \leftarrow \mu} C_i \approx C$.*

See Appendix A for the proof.

The following properties regard the reorganization of operators, and they will be essential in some technical results in which we invert the order in which hidden and visible choice are applied in a protocol.

Proposition 4.4 (“Reorganization of operators”). *Given a family $\{C_{ij}\}_{i \in \mathcal{I}, j \in \mathcal{J}}$ of channels indexed by sets \mathcal{I} and \mathcal{J} , a distribution μ on \mathcal{I} , and a distribution η on \mathcal{J} :*

- (a) $\sum_{i \leftarrow \mu} \sum_{j \leftarrow \eta} C_{ij} = \sum_{j \leftarrow \eta} \sum_{i \leftarrow \mu} C_{ij}$, if all C_i 's have the same type;
- (b) $\bigsqcup_{i \leftarrow \mu} \bigsqcup_{j \leftarrow \eta} C_{ij} \approx \bigsqcup_{j \leftarrow \eta} \bigsqcup_{i \leftarrow \mu} C_{ij}$, if all C_i 's are compatible; and
- (c) $\sum_{i \leftarrow \mu} \bigsqcup_{j \leftarrow \eta} C_{ij} \approx \bigsqcup_{j \leftarrow \eta} \sum_{i \leftarrow \mu} C_{ij}$, if, for each i , all C_{ij} 's have same type $\mathcal{X} \times \mathcal{Y}_j \rightarrow \mathbb{R}$.

See Appendix A for the proof.

Finally, analogous properties of the binary operators are shown in Appendix B.

4.4. Properties of vulnerability w.r.t. channel operators

We now derive some relevant properties of vulnerability w.r.t. our channel operators, which will be later used to obtain the Nash equilibria in information leakage games with different choice operations.

The first result states that posterior vulnerability is convex w.r.t. hidden choice (this result was already presented in [6]), and linear w.r.t. to visible choice.

Theorem 2 (Convexity/linearity of posterior vulnerability w.r.t. choices). *Let $\{C_i\}_{i \in \mathcal{I}}$ be a family of channels, and μ be a distribution on \mathcal{I} . Then, for every distribution π on \mathcal{X} , and every vulnerability \mathbb{V} :*

1. *posterior vulnerability is convex w.r.t. to hidden choice: $\mathbb{V} \left[\pi, \sum_{i \leftarrow \mu} C_i \right] \leq \sum_{i \in \mathcal{I}} \mu(i) \mathbb{V} [\pi, C_i]$ if all C_i 's have the same type.*
2. *posterior vulnerability is linear w.r.t. to visible choice: $\mathbb{V} \left[\pi, \sqcup_{i \leftarrow \mu} C_i \right] = \sum_{i \in \mathcal{I}} \mu(i) \mathbb{V} [\pi, C_i]$ if all C_i 's are compatible.*

Proof. 1. Let us call $\mathcal{X} \times \mathcal{Y} \rightarrow \mathbb{R}$ the type of each channel C_i in the family $\{C_i\}$. Then:

$$\begin{aligned}
 \mathbb{V} \left[\pi, \sum_{i \leftarrow \mu} C_i \right] &= \mathbb{V} \left[\pi, \sum_i \mu(i) C_i \right] && \text{(by definition of hidden choice)} \\
 &= \sum_{y \in \mathcal{Y}} p(y) \cdot \mathbb{V} \left[\frac{\pi(\cdot) \sum_i \mu(i) C_i(\cdot, y)}{p(y)} \right] && \text{(by definition of posterior } \mathbb{V}) \\
 &= \sum_{y \in \mathcal{Y}} p(y) \cdot \mathbb{V} \left[\sum_i \mu(i) \frac{\pi(\cdot) C_i(\cdot, y)}{p(y)} \right] \\
 &\leq \sum_{y \in \mathcal{Y}} p(y) \cdot \sum_i \mu(i) \mathbb{V} \left[\frac{\pi(\cdot) C_i(\cdot, y)}{p(y)} \right] && \text{(by convexity of } \mathbb{V}) \\
 &= \sum_i \mu(i) \sum_{y \in \mathcal{Y}} p(y) \mathbb{V} \left[\frac{\pi(\cdot) C_i(\cdot, y)}{p(y)} \right] \\
 &= \sum_i \mu(i) \mathbb{V} [\pi, C_i]
 \end{aligned}$$

where $p(y) = \sum_{x \in \mathcal{X}} \pi(x) \sum_i \mu(i) C_i(x, y)$.

2. Let us call $\mathcal{X} \times \mathcal{Y}_i \rightarrow \mathbb{R}$ the type of each channel C_i in the family $\{C_i\}$. Then:

$$\begin{aligned}
 \mathbb{V} \left[\pi, \sqcup_{i \leftarrow \mu} C_i \right] &= \mathbb{V} [\pi, \diamond_i \mu(i) C_i] && \text{(by definition of visible choice)} \\
 &= \sum_{y \in \mathcal{Y}} p(y) \cdot \mathbb{V} \left[\frac{\pi(\cdot) \diamond_i \mu(i) C_i(\cdot, y)}{p(y)} \right] && \text{(by definition of posterior } \mathbb{V}) \\
 &= \sum_{y \in \mathcal{Y}} p(y) \cdot \mathbb{V} \left[\diamond_i \mu(i) \frac{\pi(\cdot) C_i(\cdot, y)}{p(y)} \right] \\
 &= \sum_{y \in \mathcal{Y}} p(y) \cdot \sum_i \mu(i) \mathbb{V} \left[\frac{\pi(\cdot) C_i(\cdot, y)}{p(y)} \right] && \text{(see (*) below)} \\
 &= \sum_i \mu(i) \sum_{y \in \mathcal{Y}} p(y) \mathbb{V} \left[\frac{\pi(\cdot) C_i(\cdot, y)}{p(y)} \right] \\
 &= \sum_i \mu(i) \mathbb{V} [\pi, C_i]
 \end{aligned}$$

where $p(y) = \sum_{x \in \mathcal{X}} \pi(x) \sum_i \mu(i) C_i(x, y)$, and step (*) holds because in the vulnerability of a concatenation of matrices every column will contribute to the vulnerability in proportion to its

weight in the concatenation, and hence it is possible to break the vulnerability of a concatenated matrix as the weighted sum of the vulnerabilities of its sub-matrices.

□

The next result is concerned with posterior vulnerability under the composition of channels using both operators.

Corollary 1 (Convex-linear payoff function). *Let $\{C_{ij}\}_{i \in \mathcal{I}, j \in \mathcal{J}}$ be a family of channels, all with domain \mathcal{X} and with the same type, and let $\pi \in \mathbb{D}\mathcal{X}$, and \mathbb{V} be any vulnerability. Define $U : \mathbb{D}\mathcal{I} \times \mathbb{D}\mathcal{J} \rightarrow \mathbb{R}$ as follows: $U(\mu, \eta) \stackrel{\text{def}}{=} \mathbb{V}\left[\pi, \sum_{i \leftarrow \mu} \lfloor \cdot \rfloor_{j \leftarrow \eta} C_{ij}\right]$. Then U is convex on μ and linear on η .*

Proof. To see that $U(\mu, \eta)$ is convex on μ , note that:

$$\begin{aligned} U(\mu, \eta) &= \mathbb{V}\left[\pi, \sum_{i \leftarrow \mu} \lfloor \cdot \rfloor_{j \leftarrow \eta} C_{ij}\right] && \text{(by definition)} \\ &\leq \sum_i \mu(i) \mathbb{V}\left[\pi, \lfloor \cdot \rfloor_{j \leftarrow \eta} C_{ij}\right] && \text{(by Theorem 2)} \end{aligned}$$

To see that $U(\mu, \eta)$ is linear on η , note that:

$$\begin{aligned} U(\mu, \eta) &= \mathbb{V}\left[\pi, \sum_{i \leftarrow \mu} \lfloor \cdot \rfloor_{j \leftarrow \eta} C_{ij}\right] && \text{(by definition)} \\ &= \mathbb{V}\left[\pi, \lfloor \cdot \rfloor_{j \leftarrow \eta} \sum_{i \leftarrow \mu} C_{ij}\right] && \text{(by Prop. 4.4)} \\ &= \sum_j \eta(j) \mathbb{V}\left[\pi, \sum_{i \leftarrow \mu} C_{ij}\right] && \text{(by Theorem 2)} \end{aligned}$$

□

5. Information leakage games

In this section we present our framework for reasoning about information leakage, extending the notion of *information leakage games* proposed in [6] from only simultaneous games with hidden choice to both simultaneous and sequential games, with either hidden or visible choice.

In an information leakage game the defender tries to minimize the leakage of information from the system, while the attacker tries to maximize it. In this basic scenario, their goals are just opposite (zero-sum). Both of them can influence the execution and the observable behavior of the system via a specific set of actions. We assume players to be rational (i.e., they are able to figure out what is the best strategy to maximize their expected payoff), and that the set of actions and the payoff function are common knowledge.

Players choose their own strategy, which in general may be probabilistic (i.e., behavioral or mixed) and choose their action by a random draw according to that strategy. After both players have performed their actions, the system runs and produces some output value which is visible to the attacker and may leak some information about the secret. The amount of leakage constitutes the attacker's gain, and the defender's loss.

To quantify the leakage we model the system as an information-theoretic channel (cf. Section 2.2). We recall that leakage is defined as the difference (additive leakage) or the ratio (multiplicative leakage) between posterior and prior vulnerability. Since we are only interested in comparing the leakage of

different channels for a given prior, *we will define the payoff just as the posterior vulnerability*, as the value of prior vulnerability will be the same for every channel.

5.1. Defining information leakage games

An (*information*) *leakage game* consists of:

- (1) two nonempty sets \mathcal{D} , \mathcal{A} of defender's and attacker's actions, respectively,
- (2) a function $C : \mathcal{D} \times \mathcal{A} \rightarrow (\mathcal{X} \times \mathcal{Y} \rightarrow \mathbb{R})$ that associates to each pair of actions $(d, a) \in \mathcal{D} \times \mathcal{A}$ a channel $C_{da} : \mathcal{X} \times \mathcal{Y} \rightarrow \mathbb{R}$,
- (3) a prior $\pi \in \mathbb{D}\mathcal{X}$ on secrets, and
- (4) a vulnerability measure \mathbb{V} , used to define the payoff function $u : \mathcal{D} \times \mathcal{A} \rightarrow \mathbb{R}$ for pure strategies as $u(d, a) \stackrel{\text{def}}{=} \mathbb{V}[\pi, C_{da}]$. We have only one payoff function because the game is zero-sum.

Like in traditional game theory, the order of actions and the extent by which a player knows the move performed by the opponent play a critical role in deciding strategies and determining the payoff. In security, however, knowledge of the opponent's move affects the game in yet another way: the effectiveness of the attack, i.e., the amount of leakage, depends crucially on whether or not the attacker knows what channel is being used. It is therefore convenient to distinguish two phases in the leakage game:

Phase 1 - Determination of players' strategies, and the subsequent choice of their actions: Each player determines the most convenient strategy (which in general is probabilistic) for himself, and draws his action accordingly. One of the players may commit first to his action, and his choice may or may not be revealed to the follower. In general, knowledge of the leader's action may help the follower choose a more advantageous strategy.

Phase 2 - Observation of the resulting channel's output, and payoff computation: The attacker observes the output of the selected channel C_{da} and performs his attack on the secret. In case he knows the defender's action, he is able to determine the exact channel C_{da} being used (since, of course, the attacker knows his own action), and his payoff will be the posterior vulnerability $\mathbb{V}[\pi, C_{da}]$. However, if the attacker does not know exactly which channel has been used, then his payoff will be smaller.

Note that the issues raised in Phase 2 are typical of leakage games; they do not have a correspondence (to the best of our knowledge) in traditional game theory. Indeed, in traditional game theory the resulting payoff is a deterministic function of all players' actions. On the other hand, the extra level of randomization provided by the channel is central to security, as it reflects the principle of preventing the attacker from inferring the secret by obfuscating the link between secret and observables.

Following the above discussion, we consider various possible scenarios for games, along two lines of classification. The first classification concerns Phase 1 of the game, in which strategies are selected and actions are drawn, and consists in three possible orders for the two players' actions.

Simultaneous: The players choose (draw) their actions in parallel, each without knowing the choice of the other.

Sequential, defender-first: The defender draws an action, and commits to it, before the attacker does.

Sequential, attacker-first: The attacker draws an action, and commits to it, before the defender does.

Note that these sequential games may present imperfect information (i.e., the follower may not know the leader's action), and that we have to further specify whether we use behavioral or mixed strategies.

The second classification concerns Phase 2 of the game, in which some leakage occurs in consequence of the attacker's observation of the channel's output, and consists in two kinds of knowledge the attacker may have at this point about the channel that was used.

Table 2. Kinds of games we consider. Sequential games have perfect information, except for Game V.

		Order of action		
		simultaneous	defender first	attacker first
Defender's choice	visible $\lfloor \cdot \rfloor$	Game I	Game II	Game III
	hidden $\overline{\Sigma}$	Game IV	Game V	Game VI

Visible choice: The attacker knows the defender's action when he observes the output of the channel, and therefore he knows which channel is being used. Visible choice is modeled by the operator $\lfloor \cdot \rfloor$.

Hidden choice: The attacker does not know the defender's action when he observes the output of the channel, and therefore in general he does not exactly know which channel is used (although in some special cases he may infer it from the output). Hidden choice is modeled by the operator $\overline{\Sigma}$.

Note that the distinction between sequential and simultaneous games is orthogonal to that between visible and hidden choice. Sequential and simultaneous games model whether or not, respectively, the follower's choice can be affected by knowledge of the leader's action. This dichotomy captures how knowledge about the other player's actions can *help a player choose his own action*, and it concerns how Phase 1 of the game occurs. On the other hand, visible and hidden choice capture whether or not, respectively, the attacker is able to fully determine the channel representing the system, *once the defender and attacker's actions have already been fixed*. This dichotomy reflects the different *amounts of information leaked* by the system as viewed by the attacker, and it concerns how Phase 2 of the game occurs. For instance, in a simultaneous game neither player can choose his action based on the choice of the other. However, depending on whether or not the defender's choice is visible, the attacker will or will not, respectively, be able to completely recover the channel used, which will affect the amount of leakage.

If we consider also the subdivision of sequential games into perfect and imperfect information, there are 10 possible different combinations. Some, however, make little sense. For instance, the defender-first sequential game with perfect information (by the attacker) does not combine naturally with hidden choice $\overline{\Sigma}$, since that would mean that the attacker knows the action of the defender and chooses his strategy accordingly, but forgets it at the moment of computing the channel and its vulnerability. (We assume *perfect recall*, i.e., the players never forget what they have learned.) Yet other combinations are not interesting, such as the attacker-first sequential game with (totally) imperfect information (by the defender), since it coincides with the simultaneous-game case. Note that the attacker and defender are not symmetric with respect to hiding/revealing their actions a and d , since the knowledge of a affects the game only in the usual sense of game theory (in Phase 1), while the knowledge of d also affects the computation of the payoff (in Phase 2). Note that the attacker and defender are not symmetric with respect to hiding/revealing their actions a and d , since the knowledge of a affects the game only in the usual sense of game theory, while the knowledge of d also affects the computation of the payoff (cf. "Phase 2" above). Other possible combinations would come from the distinction between behavioral and mixed strategies, but, as we will see, they are always equivalent except in one scenario, so for the sake of conciseness we prefer to treat it as a case apart.

Table 2 lists the meaningful and interesting combinations. In Game V we assume imperfect information: the attacker does not know the action chosen by the defender. In all the other sequential games we assume that the follower has perfect information. In the remaining of this section, we discuss each game individually, using the example of Section 3 as a running example.

5.1.1. Game I (simultaneous with visible choice)

This simultaneous game can be represented by a tuple $(\mathcal{D}, \mathcal{A}, u)$. As in all games with visible choice $\lfloor \cdot \rfloor$, the expected payoff U of a mixed strategy profile (δ, α) is defined to be the expected value of u , as in traditional game theory:

$$U(\delta, \alpha) \stackrel{\text{def}}{=} \mathbb{E}_{\substack{d \leftarrow \delta \\ a \leftarrow \alpha}} u(d, a) = \sum_{\substack{d \in \mathcal{D} \\ a \in \mathcal{A}}} \delta(d) \alpha(a) u(d, a),$$

where we recall that $u(d, a) = \mathbb{V}[\pi, C_{da}]$.

From Theorem 2(2) we derive that $U(\delta, \alpha) = \mathbb{V}\left[\pi, \lfloor \cdot \rfloor_{\substack{d \leftarrow \delta \\ a \leftarrow \alpha}} C_{da}\right]$, and hence the whole system can be equivalently regarded as the channel $\lfloor \cdot \rfloor_{\substack{d \leftarrow \delta \\ a \leftarrow \alpha}} C_{da}$. Still from Theorem 2(2) we can derive that $U(\delta, \alpha)$ is linear in δ and α . Therefore the Nash equilibrium can be computed using the standard method (cf. Section 2.1).

Example 5. Consider the example of Section 3 in the setting of Game I, with uniform prior. The Nash equilibrium (δ^*, α^*) can be obtained using the closed formula from Section 2.1, and it is given by $\delta^*(0) = \alpha^*(0) = (2/3-1)/(1/2-1-1+2/3) = 2/5$. The corresponding payoff is $U(\delta^*, \alpha^*) = 2/5 \cdot 2/5 \cdot 1/2 + 2/5 \cdot 3/5 + 3/5 \cdot 2/5 + 3/5 \cdot 3/5 \cdot 2/3 = 4/5$.

5.1.2. Game II (defender-first with visible choice)

This defender-first sequential game can be represented by a tuple $(\mathcal{D}, \mathcal{D} \rightarrow \mathcal{A}, u)$. We will first consider mixed strategies for the follower (which in this case is the attacker), namely strategies of type $\mathbb{D}(\mathcal{D} \rightarrow \mathcal{A})$. Hence a (mixed) strategy profile is of the form (δ, σ_a) , with $\delta \in \mathbb{D}\mathcal{D}$ and $\sigma_a \in \mathbb{D}(\mathcal{D} \rightarrow \mathcal{A})$, and the corresponding payoff is

$$U(\delta, \sigma_a) \stackrel{\text{def}}{=} \mathbb{E}_{\substack{d \leftarrow \delta \\ s_a \leftarrow \sigma_a}} u(d, s_a(d)) = \sum_{\substack{d \in \mathcal{D} \\ s_a: \mathcal{D} \rightarrow \mathcal{A}}} \delta(d) \sigma_a(s_a) u(d, s_a(d)),$$

where $u(d, s_a(d)) = \mathbb{V}[\pi, C_{ds_a(d)}]$.

Again, from Theorem 2(2) we derive: $U(\delta, \sigma_a) = \mathbb{V}\left[\pi, \lfloor \cdot \rfloor_{\substack{d \leftarrow \delta \\ s_a \leftarrow \sigma_a}} C_{ds_a(d)}\right]$ and hence the system can be expressed as a channel $\lfloor \cdot \rfloor_{\substack{d \leftarrow \delta \\ s_a \leftarrow \sigma_a}} C_{ds_a(d)}$. From the same theorem we also derive that $U(\delta, \sigma_a)$ is linear in δ and σ_a , so the mutually optimal strategies can be obtained again by solving the minimax problem. In this case, however, the solution is particularly simple, because there are always deterministic optimal strategy profiles. We first consider the case of attacker's strategies of type $\mathbb{D}(\mathcal{D} \rightarrow \mathcal{A})$.

Theorem 3 (Pure-strategy Nash equilibrium in Game II - strategies of type $\mathbb{D}(\mathcal{D} \rightarrow \mathcal{A})$). Consider a defender-first sequential game with visible choice, and attacker's strategies of type $\mathbb{D}(\mathcal{D} \rightarrow \mathcal{A})$. Let $d^* \stackrel{\text{def}}{=} \operatorname{argmin}_d \max_a u(d, a)$ and let $s_a^* : \mathcal{D} \rightarrow \mathcal{A}$ be defined as $s_a^*(d) \stackrel{\text{def}}{=} \operatorname{argmax}_a u(d, a)$ (if there is more than one that maximizes $u(d, a)$, we select one of them arbitrarily). Then for every $\delta \in \mathbb{D}\mathcal{D}$ and $\sigma_a \in \mathbb{D}(\mathcal{D} \rightarrow \mathcal{A})$ we have $U(d^*, \sigma_a) \leq u(d^*, s_a^*(d^*)) \leq U(d^*, s_a^*)$.

Proof. Let δ and σ_a be arbitrary elements of $\mathbb{D}\mathcal{D}$ and $\mathbb{D}(\mathcal{D} \rightarrow \mathcal{A})$, respectively. Then:

$$\begin{aligned} U(d^*, \sigma_a) &= \sum_{s_a: \mathcal{D} \rightarrow \mathcal{A}} \sigma_a(s_a) u(d^*, s_a(d^*)) \\ &\leq \sum_{s_a: \mathcal{D} \rightarrow \mathcal{A}} \sigma_a(s_a) u(d^*, s_a^*(d^*)) && \text{(by definition of } s_a^*) \\ &= u(d^*, s_a^*(d^*)) && \text{(since } \sigma_a \text{ is a distribution)} \end{aligned}$$

$$\begin{aligned}
&= \sum_{d \in \mathcal{D}} \delta(d) u(d^*, s_a^*(d^*)) && \text{(since } \delta \text{ is a distribution)} \\
&\leq \sum_{d \in \mathcal{D}} \delta(d) u(d, s_a^*(d)) && \text{(by definition of } d^*) \\
&= U(\delta, s_a^*)
\end{aligned}$$

□

Hence to find the optimal strategy it is sufficient for the defender to find the action d^* which minimizes $\max_a u(d^*, a)$, while the attacker's optimal choice is the pure strategy s_a^* such that $s_a^*(d) = \operatorname{argmax}_a u(d, a)$, where d is the (visible) move by the defender.

Example 6. Consider the example of Section 3 in the setting of Game II, with uniform prior. If the defender chooses 0 then the attacker chooses 1. If the defender chooses 1 then the attacker chooses 0. In both cases, the payoff is 1. The game has therefore two solutions, (δ_1^*, α_1^*) and (δ_2^*, α_2^*) , with $\delta_1^*(0) = 1, \alpha_1^*(0) = 0$ and $\delta_2^*(0) = 0, \alpha_2^*(1) = 1$.

Consider now the case of behavioral strategies. Following the same line of reasoning as before, we can see that under the strategy profile (δ, ϕ_a) the system can be expressed as the channel

$$\begin{array}{c} \boxed{\cdot} \\ \downarrow \\ d \leftarrow \delta \end{array} \begin{array}{c} \boxed{\cdot} \\ \downarrow \\ a \leftarrow \phi_a(d) \end{array} C_{da}.$$

Also in this case there are deterministic optimal strategy profiles. An optimal strategy for the follower (in this case the attacker) consists in looking at the action d chosen by the leader and then selecting with probability 1 the action a that maximizes $u(d, a)$.

Theorem 4 (Pure-strategy Nash equilibrium in Game II - strategies of type $\mathcal{D} \rightarrow \mathbb{D}(\mathcal{A})$). Consider a defender-first sequential game with visible choice, and attacker's strategies of type $\mathcal{D} \rightarrow \mathbb{D}(\mathcal{A})$. Let $d^* \stackrel{\text{def}}{=} \operatorname{argmin}_d \max_a u(d, a)$ and let $\phi_a^* : \mathcal{D} \rightarrow \mathbb{D}(\mathcal{A})$ be defined as $\phi_a^*(d)(a) \stackrel{\text{def}}{=} 1$ if $a = \operatorname{argmax}_{a'} u(d, a')$ (if there is more than one such a , we select one of them arbitrarily), and $\phi_a^*(d)(a) \stackrel{\text{def}}{=} 0$ otherwise. Then for every $\delta \in \mathbb{D}\mathcal{D}$ and $\phi_a : \mathcal{D} \rightarrow \mathbb{D}(\mathcal{A})$ we have: $U(d^*, \phi_a(d^*)) \leq U(d^*, \phi_a^*(d^*)) \leq U(\delta, \phi_a^*)$.

Proof. Let a^* be the action selected by $\phi_a^*(d^*)$, i.e., $\phi_a^*(d^*)(a^*) \stackrel{\text{def}}{=} 1$. Then $u(d^*, a^*) = \max_a u(d^*, a)$. Let δ and ϕ_a be arbitrary elements of $\mathbb{D}\mathcal{D}$ and $\mathcal{D} \rightarrow \mathbb{D}(\mathcal{A})$, respectively. Then:

$$\begin{aligned}
U(d^*, \phi_a(d^*)) &= \sum_{a \in \mathcal{A}} \phi_a(d^*)(a) u(d^*, a) \\
&\leq \sum_{a \in \mathcal{A}} \phi_a(d^*)(a) u(d^*, a^*) && \text{(since } u(d^*, a^*) = \max_a u(d^*, a)) \\
&= u(d^*, a^*) && \text{(since } \phi_a(d^*) \text{ is a distribution)} \\
&= U(d^*, \phi_a^*(d^*)) && \text{(by definition of } a^*) \\
&= \sum_{d \in \mathcal{D}} \delta(d) u(d^*, \phi_a^*(d^*)) && \text{(since } \delta \text{ is a distribution)} \\
&\leq \sum_{d \in \mathcal{D}} \delta(d) u(d, \phi_a^*(d)) && \text{(by definition of } d^* \text{ and of } \phi_a^*) \\
&= U(\delta, \phi_a^*)
\end{aligned}$$

□

As a consequence of Theorems 3 and 4 we can show that in the games we consider the payoff of the optimal mixed and behavioral strategy profiles coincide. Note that this result could also be derived from the result from standard game theory which states that, in the cases we consider, for any behavioral strategy there is a mixed strategy that yields the same payoff, and viceversa [15]. However, the proof of [15] relies on Khun's theorem, which is non-constructive (and rather complicated, because it is for more general cases). In our scenario the proof is very simple, as we will see in the following corollary. Furthermore, since such a result does not hold for leakage games with hidden choice, we think it will be useful to show the proof formally in order to analyze the difference.

Corollary 2 (Equivalence of optimal strategies of types $\mathbb{D}(\mathcal{D} \rightarrow \mathcal{A})$ and $\mathcal{D} \rightarrow \mathbb{D}(\mathcal{A})$ in Game II). *Consider a defender-first sequential game with visible choice, and let d^* , s_a^* and ϕ_a^* be defined as in Theorems 3 and 4 respectively. Then $u(d^*, s_a^*(d^*)) = U(d^*, \phi_a^*(d^*))$.*

Proof. The result follows immediately by observing that $u(d^*, s_a^*(d^*)) = \max_a u(d^*, a) = u(d^*, a^*) = U(d^*, \phi_a^*(d^*))$. \square

5.1.3. Game III (attacker-first with visible choice)

This game is also a sequential game, but with the attacker as the leader. Therefore it can be represented as tuple of the form $(\mathcal{A} \rightarrow \mathcal{D}, \mathcal{A}, u)$. It is the same as Game II, except that the roles of the attacker and the defender are inverted. In particular, the payoff of a mixed strategy profile $(\sigma_d, \alpha) \in \mathbb{D}(\mathcal{A} \rightarrow \mathcal{D}) \times \mathbb{D}\mathcal{A}$ is given by

$$U(\sigma_d, \alpha) \stackrel{\text{def}}{=} \mathbb{E}_{\substack{s_d \leftarrow \sigma_d \\ a \leftarrow \alpha}} u(s_d(a), a) = \sum_{\substack{s_d: \mathcal{A} \rightarrow \mathcal{D} \\ a \in \mathcal{A}}} \sigma_d(s_d) \alpha(a) u(s_d(a), a)$$

and by Theorem 2(2) the whole system can be equivalently regarded as channel $\left[\begin{array}{c} \cdot \\ s_d \leftarrow \sigma_d \\ a \leftarrow \alpha \end{array} \right] C_{s_d(a)a}$. For a behavioral strategy $(\phi_d, \alpha) \in (\mathcal{A} \rightarrow \mathbb{D}(\mathcal{D})) \times \mathbb{D}\mathcal{A}$, the payoff is given by

$$U(\phi_d, \alpha) \stackrel{\text{def}}{=} \mathbb{E}_{a \leftarrow \alpha} \mathbb{E}_{d \leftarrow \phi_d(a)} u(d, a) = \sum_{a \in \mathcal{A}} \alpha(a) \sum_{d \in \mathcal{D}} \phi_d(a)(d) u(d, a)$$

and by Theorem 2(2) the whole system can be equivalently regarded as channel $\left[\begin{array}{c} \cdot \\ a \leftarrow \alpha \\ d \leftarrow \phi_d(a) \end{array} \right] C_{da}$.

Obviously, the same results that we have obtained in previous section for Game II hold also for Game III, with the role of attacker and defender switched. We collect all these results in the following theorem.

Theorem 5 (Pure-strategy Nash equilibria in Game III and equivalence of $\mathbb{D}(\mathcal{A} \rightarrow \mathcal{D})$ and $(\mathcal{A} \rightarrow \mathbb{D}(\mathcal{D}))$). *Consider a defender-first sequential game with visible choice. Let $a^* \stackrel{\text{def}}{=} \operatorname{argmax}_a \min_d u(d, a)$. Let $s_d^* : \mathcal{A} \rightarrow \mathcal{D}$ be defined as $s_d^*(a) \stackrel{\text{def}}{=} \operatorname{argmin}_d u(d, a)$, and let $\phi_d^* : \mathcal{A} \rightarrow \mathbb{D}(\mathcal{D})$ be defined as $\phi_d^*(a)(d) \stackrel{\text{def}}{=} 1$ if $d = \operatorname{argmin}_d u(d, a)$. Then:*

1. For every $\alpha \in \mathbb{D}\mathcal{A}$ and $\sigma_d \in \mathbb{D}(\mathcal{A} \rightarrow \mathcal{D})$ we have $U(s_d^*, \alpha) \leq u(s_d^*(a^*), a^*) \leq U(\sigma_d, a^*)$.
2. For every $\alpha \in \mathbb{D}\mathcal{A}$ and $\phi_d : \mathcal{A} \rightarrow \mathbb{D}(\mathcal{D})$ we have: $U(\phi_d^*, \alpha) \leq U(\phi_d^*(a^*), a^*) \leq U(\phi_d(a^*), a^*)$.
3. $u(s_d^*(a^*), a^*) = U(\phi_d^*(a^*), a^*)$.

Proof. These results can be proved by following the same lines as the proofs of Theorems 3 and 4, and Corollary 2. \square

Example 7. Consider now the example of Section 3 in the setting of Game III, with uniform prior. If the attacker chooses 0 then the defender chooses 0 and the payoff is $1/2$. If the attacker chooses 1 then the defender chooses 1 and the payoff is $2/3$. The latter case is more convenient for the attacker, hence the solution of the game is the strategy profile (δ^*, α^*) with $\delta^*(0) = 0, \alpha^*(0) = 0$.

5.1.4. Game IV (simultaneous with hidden choice)

The simultaneous game with hidden choice is a tuple $(\mathcal{D}, \mathcal{A}, u)$. However, it is not an ordinary game in the sense that the payoff a mixed strategy profile cannot be defined by averaging the payoff of the corresponding pure strategies. More precisely, the payoff of a mixed profile is defined by averaging on the strategy of the attacker, but not on that of the defender. In fact, when hidden choice is used, there is an additional level of uncertainty in the relation between the observables and the secret from the point of view of the attacker, since he is not sure about which channel is producing those observables. A mixed strategy δ for the defender produces a convex combination of channels (the channels associated to the pure strategies) with the same coefficients, and we know from previous sections that the vulnerability is a convex function of the channel, and in general is not linear.

In order to define the payoff of a mixed strategy profile (δ, α) , we need therefore to consider the channel that the attacker perceives given his limited knowledge. Let us assume that the action that the attacker draws from α is a . He does not know the action of the defender, but we can assume that he knows his strategy (each player can derive the optimal strategy of the opponent, under the assumption of common knowledge and rational players).

The channel the attacker will see is $\sum_{d \leftarrow \delta} C_{da}$, obtaining a corresponding payoff of $\mathbb{V}[\pi, \sum_{d \leftarrow \delta} C_{da}]$. By averaging on the strategy of the attacker we obtain

$$U(\delta, \alpha) \stackrel{\text{def}}{=} \mathbb{E}_{a \leftarrow \alpha} \mathbb{V} \left[\pi, \sum_{d \leftarrow \delta} C_{da} \right] = \sum_{a \in \mathcal{A}} \alpha(a) \mathbb{V} \left[\pi, \sum_{d \leftarrow \delta} C_{da} \right].$$

From Theorem 2(2) we derive: $U(\delta, \alpha) = \mathbb{V}[\pi, \lfloor \cdot \rfloor_{a \leftarrow \alpha} \sum_{d \leftarrow \delta} C_{da}]$ and hence the whole system can be equivalently regarded as channel $\lfloor \cdot \rfloor_{a \leftarrow \alpha} \sum_{d \leftarrow \delta} C_{da}$. Note that, by Proposition 4.4(3), the order of the operators is interchangeable, and the system can be equivalently regarded as $\sum_{d \leftarrow \delta} \lfloor \cdot \rfloor_{a \leftarrow \alpha} C_{da}$. This shows the robustness of this model.

From Corollary 1 we derive that $U(\delta, \alpha)$ is convex in δ and linear in η , hence we can compute the Nash equilibrium by the minimax method.

Example 8. Consider now the example of Section 3 in the setting of Game IV. For $\delta \in \mathbb{D}\mathcal{D}$ and $\alpha \in \mathbb{D}\mathcal{A}$, let $p = \delta(0)$ and $q = \alpha(0)$. The system can be represented by the channel $(C_{00} \oplus_p C_{10}) \oplus_q (C_{01} \oplus_p C_{11})$ represented below.

$C_{00} \oplus_p C_{10}$	$y = 0$	$y = 1$
$x = 0$	p	\bar{p}
$x = 1$	1	0

 $\oplus_q \lfloor \cdot \rfloor$

$C_{01} \oplus_p C_{11}$	$y = 0$	$y = 1$
$x = 0$	$1/3 + 2/3 p$	$2/3 - 2/3 p$
$x = 1$	$2/3 - 2/3 p$	$1/3 + 2/3 p$

For uniform π , we have $\mathbb{V}[\pi, C_{00} \oplus_p C_{10}] = 1 - 1/2 p$, while $\mathbb{V}[\pi, C_{10} \oplus_p C_{11}]$ is equal to $2/3 - 2/3 p$ if $p \leq 1/4$, and equal to $1/3 + 2/3 p$ if $p > 1/4$. Hence the payoff, expressed in terms of p and q , is $U(p, q) = q(1 - 1/2 p) + \bar{q}(2/3 - 2/3 p)$ if $p \leq 1/4$, and $U(p, q) = q(1 - 1/2 p) + \bar{q}(1/3 + 2/3 p)$ if $p > 1/4$. The Nash equilibrium can be computed by imposing that the partial derivatives of $U(p, q)$ with respect to p and q are both 0, which means that we are in a saddle point. We have:

$$\frac{\partial U(p, q)}{\partial q} = \begin{cases} \frac{1}{3} + \frac{1}{6} p & , \text{ if } p \leq \frac{1}{4} \\ \frac{2}{3} - \frac{7}{6} p & , \text{ if } p > \frac{1}{4} \end{cases} \quad \frac{\partial U(p, q)}{\partial p} = \begin{cases} -\frac{2}{3} + \frac{1}{6} q & , \text{ if } p \leq \frac{1}{4} \\ \frac{2}{3} - \frac{7}{6} q & , \text{ if } p > \frac{1}{4} \end{cases}$$

We can see that the equations $\partial U(p,q)/\partial q = 0$ and $\partial U(p,q)/\partial p = 0$ do not have solutions in $[0, 1]$ for $p \leq 1/4$, while for $p > 1/4$ they have solution $p^* = q^* = 4/7$. The pair (p^*, q^*) thus constitutes the Nash equilibrium, and the corresponding payoff is $U(p^*, q^*) = 5/7$.

5.1.5. Game V (defender-first with hidden choice)

This is a defender-first sequential game with imperfect information, hence it can be represented as a tuple of the form $(\mathcal{D}, K_a \rightarrow \mathcal{A}, u)$, where K_a is a partition of \mathcal{D} . Since we are assuming perfect recall, and the attacker does not know anything about the action chosen by the defender in Phase 2, i.e., at the moment of the attack (except the probability distribution determined by his strategy), we must assume that the attacker does not know anything in Phase 1 either. Hence the indistinguishability relation must be total, i.e., $K_a = \{\mathcal{D}\}$. But $\{\mathcal{D}\} \rightarrow \mathcal{A}$ is equivalent to \mathcal{A} , hence this kind of game is equivalent to Game IV. It is also a well known fact in game theory that when in a sequential game the follower does not know the leader's move before making his choice, the game is equivalent to a simultaneous game.⁶

5.1.6. Game VI (attacker-first with hidden choice)

This game is also a sequential game with the attacker as the leader, hence it is a tuple of the form $(\mathcal{A} \rightarrow \mathcal{D}, \mathcal{A}, u)$. It is similar to Game III, except that the payoff is convex on the strategy of the defender, instead of linear. We will see, however, that this causes quite some deviation from the properties of Game III, and from standard game theory.

The payoff of the mixed strategy profile $(\sigma_d, \alpha) \in \mathbb{D}(\mathcal{A} \rightarrow \mathcal{D}) \times \mathbb{D}\mathcal{A}$ is

$$U(\sigma_d, \alpha) \stackrel{\text{def}}{=} \mathbb{E}_{a \leftarrow \alpha} \mathbb{V} \left[\pi, \sum_{s_d \leftarrow \sigma_d} C_{s_d(a)a} \right] = \mathbb{V} \left[\pi, \left[\cdot \right]_{a \leftarrow \alpha} \sum_{s_d \leftarrow \sigma_d} C_{s_d(a)a} \right],$$

so the whole system can be equivalently regarded as channel $\left[\cdot \right]_{a \leftarrow \alpha} \sum_{s_d \leftarrow \sigma_d} C_{s_d(a)a}$.

The first important difference from Game III is that in Game VI there may not exist optimal strategies, either mixed or behavioral, that are deterministic for the defender. On the other hand, for the attacker there are always deterministic optimal strategies, and this is true independently from whether the defender uses mixed or behavioral strategies.

To show the existence of deterministic optimal strategies for the attacker, let us first introduce some standard notation for functions: given a variable x and an expression M , $\lambda x.M$ represents the function that on the argument x gives as result the value of M . Given two sets X and Y where Y is provided with an ordering \leq , the *point-wise ordering* on $X \rightarrow Y$ is defined as follows: for $f, g : X \rightarrow Y$, $f \leq g$ if and only if $\forall x \in X. f(x) \leq g(x)$.

Theorem 6 (Attacker's pure-strategy Nash equilibrium in Game VI). *Consider an attacker-first sequential game with hidden choice.*

1. Mixed strategies – type $\mathbb{D}(\mathcal{A} \rightarrow \mathcal{D})$. Let $a^* \stackrel{\text{def}}{=} \operatorname{argmax}_a \min_{\sigma_d} \mathbb{V} \left[\pi, \sum_{s_d \leftarrow \sigma_d} C_{s_d(a)a} \right]$, and let $\sigma_d^* \stackrel{\text{def}}{=} \operatorname{argmin}_{\sigma_d} \lambda a. \mathbb{V} \left[\pi, \sum_{s_d \leftarrow \sigma_d} C_{s_d(a)a} \right]$. Then, for every $\alpha \in \mathbb{D}\mathcal{A}$ and $\sigma_d \in \mathbb{D}(\mathcal{A} \rightarrow \mathcal{D})$ we have

$$U(\sigma_d^*, \alpha) \leq U(\sigma_d^*, a^*) \leq U(\sigma_d, a^*)$$

⁶ However, one could argue that, since the defender has already committed, the attacker does not need to perform the action corresponding to the Nash equilibrium, any payoff-maximizing solution would be equally good for him.

2. Behavioral strategies – type $\mathcal{A} \rightarrow \mathbb{D}(\mathcal{D})$. Let $a^* \stackrel{\text{def}}{=} \operatorname{argmax}_a \min_\delta \mathbb{V}[\pi, \sum_{d \leftarrow \delta} C_{da}]$, and let $\phi_d^* \stackrel{\text{def}}{=} \operatorname{argmin}_{\phi_d} \lambda a. \mathbb{V}[\pi, \sum_{d \leftarrow \phi_d(a)} C_{da}]$. (The minimization is with respect to the point-wise ordering.) Then, for every $\alpha \in \mathbb{D}\mathcal{A}$ and $\phi_d : \mathcal{A} \rightarrow \mathbb{D}(\mathcal{D})$ we have

$$U(\phi_d^*, \alpha) \leq U(\phi_d^*, a^*) \leq U(\phi_d, a^*)$$

Proof. 1. Let α and σ_d be arbitrary elements of $\mathbb{D}\mathcal{A}$ and $\mathbb{D}(\mathcal{A} \rightarrow \mathcal{D})$, respectively. Then:

$$\begin{aligned} U(\sigma_d^*, \alpha) &= \sum_{a \in \mathcal{A}} \alpha(a) \mathbb{V} \left[\pi, \sum_{s_d \leftarrow \sigma_d^*} C_{s_d(a)a} \right] \\ &\leq \sum_{a \in \mathcal{A}} \alpha(a) \mathbb{V} \left[\pi, \sum_{s_d \leftarrow \sigma_d^*} C_{s_d(a^*)a^*} \right] && \text{(by definition of } a^* \text{ and } \sigma_d^*) \\ &= \mathbb{V} \left[\pi, \sum_{s_d \leftarrow \sigma_d^*} C_{s_d(a^*)a^*} \right] (= U(\sigma_d^*, a^*)) && \text{(since } \alpha \text{ is a distribution)} \\ &\leq \mathbb{V} \left[\pi, \sum_{s_d \leftarrow \sigma_d} C_{s_d(a^*)a^*} \right] && \text{(by definition of } \sigma_d^*) \\ &= U(\sigma_d, a^*) \end{aligned}$$

2. Let α and ϕ_d be arbitrary elements of $\mathbb{D}\mathcal{A}$ and $\mathcal{A} \rightarrow \mathbb{D}(\mathcal{D})$, respectively. Then:

$$\begin{aligned} U(\phi_d^*, \alpha) &= \sum_{a \in \mathcal{A}} \alpha(a) \mathbb{V} \left[\pi, \sum_{d \leftarrow \phi_d^*(a)} C_{da} \right] \\ &\leq \sum_{a \in \mathcal{A}} \alpha(a) \mathbb{V} \left[\pi, \sum_{d \leftarrow \phi_d^*(a^*)} C_{da^*} \right] && \text{(by definition of } a^* \text{ and } \phi_d^*) \\ &= \mathbb{V} \left[\pi, \sum_{d \leftarrow \phi_d^*(a^*)} C_{da^*} \right] (= U(\phi_d^*, a^*)) && \text{(since } \alpha \text{ is a distribution)} \\ &\leq \mathbb{V} \left[\pi, \sum_{d \leftarrow \phi_d(a^*)} C_{da^*} \right] && \text{(by definition of } \phi_d^*) \\ &= U(\phi_d, a^*) \end{aligned}$$

□

We show now, with the following example, that the optimal strategies for the defender are necessarily probabilistic.

Example 9. Consider the channel matrices C_{ij} defined in Section 3, and define the following new channels: $C'_{00} = C'_{11} = C_{01}$ and $C'_{10} = C'_{01} = C_{10}$. Define D_p as the result of the hidden choice, with probability p , between C'_{00} and C'_{10} , i.e., $D_p \stackrel{\text{def}}{=} C'_{00} \oplus_p C'_{10}$, and observe that $D_p[0,0] = D_p[1,1] = p$, and $D_p[1,0] = D_p[0,1] = 1 - p$. Furthermore, $D_p = C'_{01} \oplus_{1-p} C'_{11}$. The vulnerability of D_p , for uniform π , is $\mathbb{V}[\pi, D_p] = 1 - 1/2p$ for $p \leq 1/2$ and $\mathbb{V}[\pi, D_p] = p$ for $p > 1/2$, hence, independently from the choice of the attacker, the best strategy for the defender is to choose $p = 1/2$. Every other value for p gives a strictly higher vulnerability. Therefore, the best mixed strategy for the defender is σ_d^* defined as $\sigma_d^*(\lambda_a.0) = \sigma_d^*(\lambda_a.1) = 1/2$. Similarly, the best behavioral strategy for the defender is ϕ_d^* defined as $\phi_d^*(0) = \phi_d^*(1) = \lambda_d.1/2$.

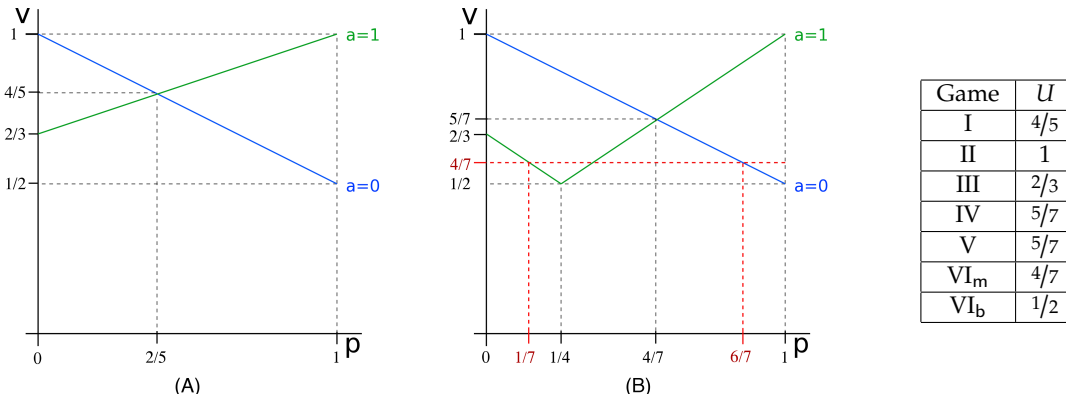


Figure 2. Summary of the results for the running example introduced in Section 3, for uniform prior. Graph (A) is for the case of visible choice: it represents the Bayes vulnerability \mathbb{V} of $C_{00} \sqcup C_{10}$ and of $C_{01} \sqcup C_{11}$ (cases $a = 0$ and $a = 1$ respectively), as a function of p . Graph (B) is for the case of hidden choice and it represents the vulnerability of $C_{00} \oplus C_{10}$ and of $C_{01} \oplus C_{11}$ as a function of p . The table on the right gives the payoff in correspondence of the Nash equilibrium for the various games. VI_m and VI_b represent the attacker-first sequential games with defender strategy of type $\mathbb{D}(\mathcal{A} \rightarrow \mathcal{D})$ (mixed) and $\mathcal{A} \rightarrow \mathbb{D}(\mathcal{D})$ (behavioral), respectively.

The second important difference from Game III is that in Game VI behavioral strategies and mixed strategies are not necessarily equivalent. More precisely, there are cases in which the optimal strategy profile yields a different payoff depending on whether the defender adopts mixed strategies or behavioral ones. The following is an example in which this difference manifests itself.

Example 10. Consider again the example of Section 3, this time in the setting of Game VI, and still with uniform prior π . Let us analyze first the case in which the defender uses behavioral strategies.

Behavioral strategies – type $\mathcal{A} \rightarrow \mathbb{D}(\mathcal{D})$. If the attacker chooses 0, which corresponds to committing to the system $C_{00} \oplus C_{10}$, then the defender will choose $p = 1/4$, which minimizes its vulnerability. If he chooses 1, which corresponds to committing to the system $C_{01} \oplus C_{11}$, then the defender will choose $p = 1$, which minimizes the vulnerability. In both cases, the leakage is $p = 1/2$, hence both these strategies are solutions to the minimax. Note that in the first case the strategy of the defender is probabilistic, while that of the attacker is pure in both cases.

Mixed strategies – type $\mathbb{D}(\mathcal{A} \rightarrow \mathcal{D})$. Observe that there are only four possible pure strategies for the defender, corresponding to the four functions $f_{ij} : \mathcal{A} \rightarrow \mathcal{D}$ for $i, j \in \{0, 1\}$ defined as $f_{ij}(a) \stackrel{\text{def}}{=} i$ if $i = j$ and $f_{ij}(a) \stackrel{\text{def}}{=} a \oplus i$ if $i \neq j$. Consider a distribution $\sigma_d \in \mathbb{D}(\mathcal{A} \rightarrow \mathcal{D})$ and let $p_{ij} \stackrel{\text{def}}{=} \sigma_d(f_{ij})$. Then we have $p_{ij} \geq 0$ and $\sum_{i,j} p_{ij} = 1$. Observe that the attacker's choice $a = 0$ determines the matrix $C_{00} \oplus C_{10}$, with $p = p_{00} + p_{10}$, whose vulnerability is $\mathbb{V}[\pi, C_{00} \oplus C_{10}] = 1 - 1/2p$. On the other hand, the attacker's choice $a = 1$ determines the matrix $C_{01} \oplus C_{11}$, with $p' = p_{00} + p_{01}$, whose vulnerability is $\mathbb{V}[\pi, C_{01} \oplus C_{11}] = 2/3 - 2/3p$ for $p' \leq 1/4$, and $\mathbb{V}[\pi, C_{01} \oplus C_{11}] = 1/3 + 2/3p$ for $p' > 1/4$. By geometrical considerations (cf. the red dashed line in Figure 2) we can see that the optimal solutions for the defender are all those strategies which give $p = 6/7$ and $p' = 1/7$, which yield payoff $4/7$.

The facts that behavioral and mixed strategies are not equivalent is related to the non-existence of deterministic optimal strategies. In fact, it is easy to see that from a behavioral deterministic strategy we can construct a (deterministic) mixed strategy, and vice versa.

Figure 2 illustrates the graphs of the vulnerability of the various channel compositions, and summarizes the results of this section.

6. Comparing the leakage games

In previous section we have computed the vulnerability for the running example in the various kind of games introduced in Section 5. The values we have obtained, listed in decreasing order, are as follows: II : 1; I : 4/5; IV : 5/7; V : 5/7; III : 2/3; VI_m : 4/7; VI_b : 1/2. This order is not accidental: in this section we will prove that some of these relations between games hold for any vulnerability function, and for any prior. These results will allow us to reason about which kind of scenarios and compositions are more convenient for the defender, or, vice versa, for the attacker.

6.1. Simultaneous games vs. sequential games

The relations between II, I, and III, and between IV-V and VI_m are typical in game theory: in any zero-sum sequential game the leader's payoff is less than or equal to his payoff in the corresponding simultaneous game. In fact, by acting first, the leader commits to an action, and this commitment can be exploited by the attacker to choose the best possible strategy relatively to that action⁷. In the following propositions we give the precise formulation of these results in our framework, and we show how they can be derived formally.

Proposition 1 (Game II \geq Game I).

$$\min_{\delta} \max_{\sigma_a} \mathbb{V} \left[\pi, \left[\cdot \right]_{\substack{d \leftarrow \delta \\ s_a \leftarrow \sigma_a}} C_{ds_a}(d) \right] \geq \min_{\delta} \max_{\alpha} \mathbb{V} \left[\pi, \left[\cdot \right]_{\substack{d \leftarrow \delta \\ a \leftarrow \alpha}} C_{da} \right]$$

Proof. We prove the first inequality as follows. Independently of δ , consider the attacker's strategy σ_a^* that assigns probability 1 to the function s_a^* defined as $s_a^*(d) = \operatorname{argmax}_a \mathbb{V}[\pi, C_{da}]$ for any $d \in \mathcal{D}$. Then we have that:

$$\begin{aligned} \min_{\delta} \max_{\sigma_a} \mathbb{V} \left[\pi, \left[\cdot \right]_{\substack{d \leftarrow \delta \\ s_a \leftarrow \sigma_a}} C_{ds_a}(d) \right] &\geq \min_{\delta} \mathbb{V} \left[\pi, \left[\cdot \right]_{\substack{d \leftarrow \delta \\ s_a \leftarrow \sigma_a^*}} C_{ds_a}(d) \right] && \text{(by maximization on } \sigma_a) \\ &= \min_{\delta} \mathbb{V} \left[\pi, \left[\cdot \right]_{d \leftarrow \delta} C_{ds_a^*}(d) \right] && \text{(by definition of } \sigma_a^*) \\ &= \min_{\delta} \sum_d \delta(d) \mathbb{V}[\pi, C_{ds_a^*}(d)] && \text{(by Theorem 2(2))} \\ &= \min_{\delta} \sum_d \delta(d) \max_{\alpha} \sum_a \alpha(a) \mathbb{V}[\pi, C_{ds_a^*}(d)] && \text{(since } \alpha \text{ is a distribution)} \\ &\geq \min_{\delta} \sum_d \delta(d) \max_{\alpha} \sum_a \alpha(a) \mathbb{V}[\pi, C_{da}] && \text{(by definition of } s_a^*) \\ &\geq \min_{\delta} \max_{\alpha} \sum_d \delta(d) \sum_a \alpha(a) \mathbb{V}[\pi, C_{da}] \\ &= \min_{\delta} \max_{\alpha} \mathbb{V} \left[\pi, \left[\cdot \right]_{\substack{d \leftarrow \delta \\ a \leftarrow \alpha}} C_{da} \right] && \text{(by Theorem 2(2))} \end{aligned}$$

⁷ The fact that the leader has a disadvantage may seem counterintuitive because in many real games it is the opposite, the player who moves first has an advantage. Such discrepancy is due to the fact that these games feature preemptive moves, i.e. moves that, when made by one player, make impossible other moves for the other player. The games we are considering in this paper, on the contrary, do not consider preemptive moves.

□

Note that the strategy σ_a^* is optimal for the attacker, so the first of the above inequalities is actually an equality. It is easy to see that the second inequality is an equality as well, because of the maximization on α . Therefore, the only inequality that may be strict is the third one, and the reason why it may be strict is that in the left-hand side α depends on d (and on δ), while on the right-hand side α depends on δ , but not the actual d (that will be sampled from δ). This corresponds to the fact that in the defender-first sequential game the attacker chooses his strategy after he knows the action d chosen by the defender, while in the simultaneous game the attacker knows the strategy of the defender (i.e., the distribution δ he will use to choose probabilistically his actions), but not the actual action d that the defender will choose.

Analogous considerations can be done for the simultaneous versus the attacker-first case, that we will examine next.

Proposition 2 (Game I \geq Game III).

$$\min_{\delta} \max_{\alpha} \mathbb{V} \left[\pi, \left[\cdot \right]_{\substack{d \leftarrow \delta \\ a \leftarrow \alpha}} C_{da} \right] \geq \max_{\alpha} \min_{\sigma_d} \mathbb{V} \left[\pi, \left[\cdot \right]_{\substack{s_d \leftarrow \sigma_d \\ a \leftarrow \alpha}} C_{s_d(a)a} \right]$$

Proof. Independently of α , consider the defender's strategy σ_d^* that assigns probability 1 to the function s_d^* defined as $s_d^*(a) = \operatorname{argmin}_d \mathbb{V}[\pi, C_{da}]$ for any $a \in \mathcal{A}$. Then we have that:

$$\begin{aligned} \min_{\delta} \max_{\alpha} \mathbb{V} \left[\pi, \left[\cdot \right]_{\substack{d \leftarrow \delta \\ a \leftarrow \alpha}} C_{da} \right] &= \min_{\delta} \max_{\alpha} \sum_d \delta(d) \sum_a \alpha(a) \mathbb{V}[\pi, C_{da}] && \text{(by Theorem 2(2))} \\ &= \max_{\alpha} \min_{\delta} \sum_d \delta(d) \sum_a \alpha(a) \mathbb{V}[\pi, C_{da}] && \text{(by Theorem 1)} \\ &\geq \max_{\alpha} \sum_a \alpha(a) \min_d \mathbb{V}[\pi, C_{da}] \\ &= \max_{\alpha} \sum_a \alpha(a) \mathbb{V}[\pi, C_{s_d^*(a)a}] && \text{(by definition of } s_d^*) \\ &= \max_{\alpha} \sum_a \alpha(a) \sum_{s_d} \sigma_d^*(s_d) \mathbb{V}[\pi, C_{s_d(a)a}] && \text{(by definition of } \sigma_d^*) \\ &= \max_{\alpha} \mathbb{V} \left[\pi, \left[\cdot \right]_{\substack{s_d \leftarrow \sigma_d^* \\ a \leftarrow \alpha}} C_{s_d(a)a} \right] && \text{(by Theorem 2(2))} \\ &\geq \max_{\alpha} \min_{\sigma_d} \mathbb{V} \left[\pi, \left[\cdot \right]_{\substack{s_d \leftarrow \sigma_d \\ a \leftarrow \alpha}} C_{s_d(a)a} \right] && \text{(by minimization on } \sigma_d) \end{aligned}$$

□

Again, the strategy σ_d^* is optimal for the attacker, so the last of the above inequalities is actually an equality. Therefore, the only inequality that may be strict is the first one and the strictness is due to the fact that in the left-hand side δ does not depend on a while in the right-hand side it does. Intuitively, this corresponds to the intuition that if the defender knows the action of the attacker then it may be able to choose a better strategy to reduce the leakage.

Proposition 3 (Game IV \geq Game VI_m).

$$\min_{\delta} \max_{\alpha} \mathbb{V} \left[\pi, \left[\cdot \right]_{a \leftarrow \alpha} \sum_{d \leftarrow \delta} C_{da} \right] \geq \max_{\alpha} \min_{\sigma_d} \mathbb{V} \left[\pi, \left[\cdot \right]_{a \leftarrow \alpha} \sum_{s_d \leftarrow \sigma_d} C_{s_d(a)a} \right]$$

Proof. Given $\alpha \in \mathbb{D}\mathcal{A}$, let $\delta_{\alpha}^* \stackrel{\text{def}}{=} \min_{\delta} \sum_a \alpha(a) \mathbb{V} [\pi, \sum_{d \leftarrow \delta} C_{da}]$. For any $d \in \mathcal{D}$, let f_d be the constant function defined as $f_d(a) = d$ for any $a \in \mathcal{A}$, and define $\sigma_d^* \in \mathbb{D}(\mathcal{A} \rightarrow \mathcal{D})$ as $\sigma_d^*(f_d) \stackrel{\text{def}}{=} \delta_{\alpha}^*(d)$ for any $d \in \mathcal{D}$. Let $U(\delta, \alpha) = \mathbb{V} [\pi, \left[\cdot \right]_{a \leftarrow \alpha} \sum_{d \leftarrow \delta} C_{da}]$. Then $U(\delta, \alpha)$ is convex in δ and linear in α . Hence we have:

$$\begin{aligned} \min_{\delta} \max_{\alpha} \mathbb{V} \left[\pi, \left[\cdot \right]_{a \leftarrow \alpha} \sum_{d \leftarrow \delta} C_{da} \right] &= \max_{\alpha} \min_{\delta} \mathbb{V} \left[\pi, \left[\cdot \right]_{a \leftarrow \alpha} \sum_{d \leftarrow \delta} C_{da} \right] && \text{(by Theorem 1)} \\ &= \max_{\alpha} \mathbb{V} \left[\pi, \left[\cdot \right]_{a \leftarrow \alpha} \sum_{d \leftarrow \delta_{\alpha}^*} C_{da} \right] && \text{(by definition of } \delta_{\alpha}^*) \\ &= \max_{\alpha} \mathbb{V} \left[\pi, \left[\cdot \right]_{a \leftarrow \alpha} \sum_{f_d \leftarrow \sigma_d^*} C_{f_d(a)a} \right] && \text{(by definition of } \sigma_d^*) \\ &\geq \max_{\alpha} \min_{\sigma_d} \mathbb{V} \left[\pi, \left[\cdot \right]_{a \leftarrow \alpha} \sum_{s_d \leftarrow \sigma_d} C_{s_d(a)a} \right] && \text{(by minimization on } \sigma_d) \end{aligned}$$

□

6.2. Visible choice vs. hidden choice

We consider now the case of Games III and IV-V. In the running example the payoff for III is lower than for IV-V, but it is easy to find other cases in which the situation is reversed. For instance, if in the running example we set C_{11} to be the same as C_{01} , the payoff for III will be 1 (corresponding to the choice $a = 1$ for the attacker), and that for IV-V will be $2/3$ (corresponding to the Nash equilibrium $p^* = q^* = 2/3$). So we conclude that Games III and IV-V are incomparable: there is no general ordering between them.

The relation between Games I and IV comes from the fact that they are both simultaneous games, and the only difference is the way in which the payoff is defined. The same holds for the case of Games III and VI_m, which are both attacker-first sequential games. The essence of the proof is expressed by the following proposition.

Proposition 4 (visible choice \geq hidden choice). *For every $a \in \mathcal{A}$ and every $\delta \in \mathbb{D}\mathcal{D}$ we have:*

$$\mathbb{V} [\pi, \left[\cdot \right]_{d \leftarrow \delta} C_{da}] \geq \mathbb{V} [\pi, \sum_{d \leftarrow \delta} C_{da}].$$

Proof.

$$\begin{aligned} \mathbb{V} \left[\pi, \left[\cdot \right]_{d \leftarrow \delta} C_{da} \right] &= \sum_{d \in \mathcal{D}} \delta(d) \mathbb{V} [\pi, C_{da}] && \text{(by Theorem 2(2))} \\ &\geq \mathbb{V} \left[\pi, \sum_{d \leftarrow \delta} C_{da} \right] && \text{(by Theorem 2(1))} \end{aligned}$$

□

From the above proposition we can derive immediately the following corollaries:

Corollary 3 (Game I \geq Game IV).

$$\min_{\delta} \max_{\alpha} \mathbb{V} \left[\pi, \left[\cdot \right]_{\substack{d \leftarrow \delta \\ a \leftarrow \alpha}} C_{da} \right] \geq \min_{\delta} \max_{\alpha} \mathbb{V} \left[\pi, \left[\cdot \right]_{\substack{a \leftarrow \alpha \\ d \leftarrow \delta}} \sum C_{da} \right]$$

Corollary 4 (Game III \geq Game VI_m).

$$\max_{\alpha} \min_{\sigma_d} \mathbb{V} \left[\pi, \left[\cdot \right]_{\substack{s_d \leftarrow \sigma_d \\ a \leftarrow \alpha}} C_{s_d(a)a} \right] \geq \max_{\alpha} \min_{\sigma_d} \mathbb{V} \left[\pi, \left[\cdot \right]_{\substack{a \leftarrow \alpha \\ s_d \leftarrow \sigma_d}} \sum C_{s_d(a)a} \right]$$

Finally, we show that the vulnerability for the optimal solution in Game VI_m is always greater than or equal to that of Game VI_b, which means that for the defender it is always convenient to use behavioral strategies. We can state actually a more general result: for any mixed strategy, there is always a behavioral strategy that gives the same payoff.

Proposition 5. For any $\alpha \in \mathbb{D}\mathcal{A}$ and any $\sigma_d \in \mathbb{D}(\mathcal{A} \rightarrow \mathcal{D})$ there exists $\phi_d : \mathcal{A} \rightarrow \mathbb{D}(\mathcal{D})$ such that:

$$\mathbb{V} \left[\pi, \left[\cdot \right]_{\substack{a \leftarrow \alpha \\ d \leftarrow \phi_d(a)}} \sum C_{da} \right] = \mathbb{V} \left[\pi, \left[\cdot \right]_{\substack{a \leftarrow \alpha \\ s_d \leftarrow \sigma_d}} \sum C_{s_d(a)a} \right]$$

Proof. For $\sigma_d \in \mathbb{D}(\mathcal{A} \rightarrow \mathcal{D})$, define $\phi_d : \mathcal{A} \rightarrow \mathbb{D}(\mathcal{D})$ as follows: for any $a \in \mathcal{A}$ and $d \in \mathcal{D}$,

$$\phi_d(a)(d) \stackrel{\text{def}}{=} \sum_{s_d(a)=d} \sigma_d(s_d)$$

and observe that for every $a \in \mathcal{A}$ we have $\sum_{d \leftarrow \phi_d(a)} C_{da} = \sum_{s_d \leftarrow \sigma_d} C_{s_d(a)a}$. \square

From this proposition we derive immediately the following corollary:

Corollary 5.

$$\max_{\alpha} \min_{\sigma_d} \mathbb{V} \left[\pi, \left[\cdot \right]_{\substack{a \leftarrow \alpha \\ s_d \leftarrow \sigma_d}} \sum C_{s_d(a)a} \right] \geq \max_{\alpha} \min_{\phi_d} \mathbb{V} \left[\pi, \left[\cdot \right]_{\substack{a \leftarrow \alpha \\ d \leftarrow \phi_d(a)}} \sum C_{da} \right]$$

The lattice in Figure 3 illustrates the results of this section about the relations between the various games. These relations can be used by the defender as guidelines to better protect the system, if he has some control over the rules of the game. Obviously, for the defender the games lower in the ordering are to be preferred to compose protocols, since they yield a lower vulnerability for the result.

7. Case study: a safer, faster password-checker

In this section we apply our game-theoretic, compositional approach to show how a defender can mitigate an attacker's typical timing side-channel attack, while avoiding the usual burden imposed on the password-checker's efficiency by measures that make time consumption constant.

The following sections are organized as follows: We first provide a formalization of the trade-off between efficiency and security in password checkers using our framework of leakage games. We then illustrate the approach in a simple instance of the program for 3-bit passwords. Finally, we provide general results for the n -bit case regarding the defender's optimal strategy in equilibrium.

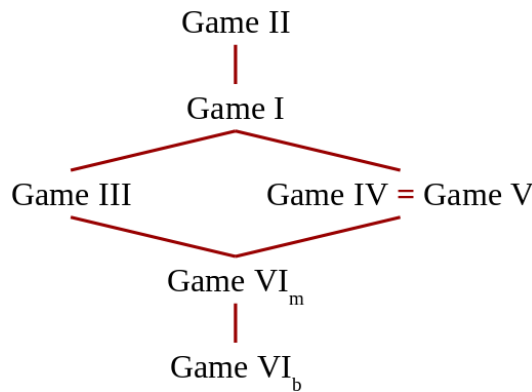


Figure 3. Order of games w.r.t. the payoff in the Nash equilibrium. Games higher in the lattice have larger payoff.

7.1. Modeling the trade-off between efficiency and security as a game

Consider the password-checker $\text{PWD}_{1..n}$ of Algorithm 1, which performs a bitwise-check of an n -bit low-input $a = a_1a_2 \dots a_n$ provided by the attacker against an n -bit secret password $x = x_1x_2 \dots x_n$. The bits are compared in increasing order $(1, 2, \dots, n)$, with the low-input being rejected as soon as it mismatches the secret, and accepted otherwise.

Algorithm 1: Password-checker $\text{PWD}_{1..n}$.

High input: $x \in \{0, 1\}^n$

Low input : $a \in \{0, 1\}^n$

Output : $y \in \{T, F\}$

$accept := T;$

for $i := 1$ **to** n **do**

if $a_i \neq x_i$ **then**

$accept := F;$

break;

return $accept;$

The attacker can choose low-inputs to try to gain information about the password. Obviously, in case $\text{PWD}_{1..n}$ accepts the low-input, the attacker learns the password value is $a = x$. Yet, even when the low-input is rejected, there is some leakage of information: from the duration of the execution the attacker can estimate how many iterations have been performed before the low-input was rejected, thus inferring a prefix of the secret password.

To model this scenario, let $\mathcal{X} = \{0, 1\}^n$ be the set of all possible n -bit passwords, and $\mathcal{Y} = \{F\} \times \{1, 2, \dots, n\} \cup \{T\} \times \{n\} = \{(F, 1), (F, 2), (F, 3), \dots, (F, n), (T, n)\}$ be the set of observables produced by the system. Each observable is an ordered pair whose first element indicates whether or not the password was accepted (T or F , respectively), and the second element indicates the duration of the computation ($1, 2, \dots$, or n iterations).

For instance, consider a scenario with 3-bit passwords. Let PWD_{123} be a password checker that performs the bitwise comparison in increasing order $(1, 2, 3)$. Channel $C_{123,101}$ in Table 3a models PWD_{123} 's behavior when the attacker provides low-input $a = 101$. Note that this channel represents the fact that PWD_{123} accepts the low-input when the secret is $x = 101$ (the channel outputs $(T, 3)$ with probability 1), and otherwise rejects the low-input in a certain number of steps (e.g., the checker rejects the low-input in 2 steps when the password is $x = 110$, so in this case the channel outputs $(F, 2)$ with probability 1).

Table 3. Channels C_{da} modeling different versions of 3-bit password checker for defender's action d and attacker's action a .

(a) Channel $C_{123,101}$ modeling the case in which the defender compares bits in order (1,2,3), and the attacker picks low-input 101.

$C_{123,101}$	$y = (F, 1)$	$y = (F, 2)$	$y = (F, 3)$	$y = (T, 3)$
$x = 000$	1	0	0	0
$x = 001$	1	0	0	0
$x = 010$	1	0	0	0
$x = 011$	1	0	0	0
$x = 100$	0	0	1	0
$x = 101$	0	0	0	1
$x = 110$	0	1	0	0
$x = 111$	0	1	0	0

(b) Channel $C_{\text{cons},101}$ modeling the case in which the defender runs a constant-time checker, and the attacker picks low-input 101.

$C_{\text{cons},101}$	$y = (F, 3)$	$y = (T, 3)$
$x = 000$	1	0
$x = 001$	1	0
$x = 010$	1	0
$x = 011$	1	0
$x = 100$	1	0
$x = 101$	0	1
$x = 110$	1	0
$x = 111$	1	0

To quantify the password checker's leakage of information, we will adopt Bayes vulnerability, so the prior Bayes vulnerability $\mathbb{V}[\pi]$ will correspond to the probability that the attacker guesses correctly the password in one try, whereas the posterior Bayes vulnerability $\mathbb{V}[\pi, C]$ will correspond to the probability that the attacker guesses correctly the password in one try, after he observes the output of the channel (i.e., after he has measured the time needed for the checker to accept or reject the low-input). For instance, in the 3-bit password scenario, if the prior distribution on all possible 3-bit passwords is $\hat{\pi} = (0.0137, 0.0548, 0.2191, 0.4382, 0.0002, 0.0002, 0.0548, 0.2191)$, the corresponding prior Bayes vulnerability is $\mathbb{V}[\hat{\pi}] = 0.4382$. For prior $\hat{\pi}$ above, the posterior Bayes vulnerability of channel $C_{123,101}$ is $\mathbb{V}[\hat{\pi}, C_{123,101}] = 0.6577$, which represents an increase in Bayes vulnerability of about 50%.

A way to mitigate this timing side-channel is to make the checker's execution time independent of the secret. This can be done by eliminating the `break` command within the loop in `PWD1..n`, so no matter when the matching among high and low input happens, the password checker will always need n iterations to complete. For instance, in the context of our 3-bit password example, we can let `PWDcons` be a constant-time 3-bit password checker that applies this counter measure. Channel $C_{\text{cons},101}$ from Table 3b models `PWDcons`'s behavior when the attacker's low-input is $a = 101$. Note that this channel reveals only whether or not the low-input matches the secret value, but does not allow the attacker to infer a prefix of the password. Indeed, this channel's posterior Bayes vulnerability is $\mathbb{V}[\hat{\pi}, C_{123,101}] = 0.4384$, which brings the multiplicative Bayes leakage down to an increase of only about 0.05%.

However, the original program is substantially more efficient than the modified one. Consider the general case of n -bit passwords and assume that either the password, or the program's low input, are chosen uniformly at random. Because of this assumption, each bit being checked in the original program has probability $1/2$ of being rejected. Hence, the program will finish after 1 iteration with probability $1/2$, after 2 iterations with probability $1/4$, and so on, up to the n -th iteration. After that the program always finishes, so with the remaining probability 2^{-n} the program finishes after n iterations, giving a total expected time of

$$\sum_{k=1}^n k2^{-k} + n2^{-n} = 2(1 - 2^{-n}).$$

The above derivation is based on the series $\sum_{k=1}^n kz^k = z \frac{1-(n+1)z^n + nz^{n+1}}{(1-z)^2}$. Hence the expected running time of the original program (under the uniform assumption) is constant: always bounded by 2, and converging to 2 as n grows. On the other hand, the running time of the modified constant-time program is n iterations, an $n/2$ -fold increase.

Table 4. Payoff for each pure strategy profile of 3-bit password scenario.

		Attacker's action a							
		$U(d, a)$	000	001	010	011	100	101	110
Defender's action d	123	0.7257	0.7257	0.9311	0.9311	0.6577	0.6577	0.7122	0.7122
	132	0.8900	0.9311	0.8900	0.9311	0.7122	0.7122	0.7122	0.7122
	213	0.5068	0.5068	0.9311	0.9311	0.4934	0.4934	0.7668	0.7668
	231	0.5068	0.5068	0.7668	0.9311	0.5068	0.5068	0.7668	0.9311
	312	0.7257	0.9311	0.7257	0.9311	0.7122	0.8766	0.7122	0.8766
	321	0.6712	0.7122	0.7257	0.9311	0.6712	0.7122	0.7257	0.9311

Seeking some compromise between security and efficiency, assume the defender can employ different versions of the password-checker, each performing the bitwise comparison among low-input a and secret password x in a different order. More precisely, there is one version of the checker for every possible order in which the index i ranges in the control of the loop in Algorithm 1.

To determine a defender's best choice of which versions of the checker to run, we model this problem as game. The attacker's set of actions \mathcal{A} consists in all possible 2^n low-inputs to the checker, and the defender's set of actions \mathcal{D} consists in all $n!$ orders in which the checker can perform the bitwise comparison. There is, then, a channel $C_{ad}: \mathcal{X} \times \mathcal{Y} \rightarrow \mathbb{R}$ for each possible combination of $d \in \mathcal{D}$, $a \in \mathcal{A}$. In our framework, the payoff of a mixed strategy profile (δ, α) is given by $U(\delta, \alpha) = \mathbb{E}_{a \leftarrow \alpha} \mathbb{V}[\pi, \sum_{d \leftarrow \delta} C_{da}]$. For each pure strategy profile (d, a) , the payoff of the game will be the posterior Bayes vulnerability of the resulting channel C_{da} (since, if we are measuring the information leakage, the prior vulnerability is the same for every channel once the prior is fixed).

In the 3-bit password scenario, the attacker's actions $\mathcal{A} = \{000, 001, 010, 011, 100, 101, 110, 111\}$ are all possible 3-bit low-inputs, and the defender's $\mathcal{D} = \{123, 132, 213, 231, 312, 321\}$ are all possible versions of the password checker (each action represents the order in which the 3 bits are checked). Table 4 depicts the corresponding payoffs of all 48 possible resulting channels C_{ad} with $d \in \mathcal{D}$, $a \in \mathcal{A}$, when the prior is still $\hat{\pi} = (0.0137, 0.0548, 0.2191, 0.4382, 0.0002, 0.0002, 0.0548, 0.2191)$. Note that the attacker's and defender's actions substantially affect the effectiveness of the attack: vulnerability ranges between 0.4934 and 0.9311 (and so multiplicative leakage is in the range between an increase of 12% and one of 112%). Using techniques from [6], we can compute the best (mixed) strategy for the defender in this game, which turns out to be

$$\delta^* = (0.1667, 0.1667, 0.1667, 0.1667, 0.1667, 0.1667).$$

This strategy is part of an equilibrium and guarantees that for any choice of the attacker the posterior Bayes vulnerability is at most 0.6573 (so the multiplicative leakage is bounded by 50%, an intermediate value between the minimum of about 12% and the maximum of about 112%).

The running time, on the other hand, of this new password-checker is the same as that of the original one. Under the assumption that either the password, or the low-input, are uniformly distributed, each check fails with probability $1/2$, giving a total expected time of $2(1 - 2^{-n})$. Hence this technique substantially decreases the program's information leakage, without affecting at all its expected running time.

7.2. On optimal strategies for the defender

Interestingly, in the 3-bit password case study from the previous section, the defender's optimal strategy consists in uniformly sampling among all available versions of the checker. A uniform distribution seems to be an adequate candidate for the defender, but is it always the best choice for any prior and any number of bits?

We first answer this question in the case of a uniform prior π for arbitrary n -bit passwords, which already turns out to be challenging. Under this prior, and exploiting a crucial symmetry of the

password checker (see the proof of Theorem 7), we can show that all strategies for the adversary are in fact equivalent, namely

$$U(\alpha, \delta) = U(\alpha', \delta) \quad \text{for all } \alpha, \alpha', \delta.$$

For the defender, on the other hand, the situation is far from trivial: although all *pure* strategies d are still equivalent, $U(\alpha, \delta)$ does in general depend on δ . By exploiting another symmetry of the password checker together with the symmetry of \mathbb{V} , we can show that a uniform strategy is indeed optimal for the defender, as stated in the following result.

Theorem 7. *Consider the password checker program of Algorithm 1 for n -bit passwords, where the attacker controls the low input to the checker, and the defender controls the order in which the bits are checked. If the prior π on possible passwords is uniform, and the payoff is given by the posterior Bayes-vulnerability $U(\delta, \alpha) = \mathbb{E}_{a \leftarrow \alpha} \mathbb{V}[\pi, \sum_{d \leftarrow \delta} C_{da}]$, then the strategy (δ^*, α) where δ^* is uniform and α is arbitrary is an equilibrium strategy.*

Perhaps surprisingly, however, Theorem 7 does not generalize to non-uniform priors (or to different vulnerability metrics). More precisely, when the prior on passwords is not uniform, the defender may benefit from assigning different probabilities to different versions of the password checker. This subtlety arises from the fact that the defender's goal is not to maximize the attacker's uncertainty about the selected password checker itself (i.e., the defender's action), but it is rather to maximize the attacker's uncertainty about the secret value. The following examples illustrate this (perhaps counter-intuitive) fact.

Consider again a 3-bit password scenario, similar to that of the previous section. Assume that the attacker knows only that the first bit of the password is surely 0, so that the prior on secrets is

$$\pi^{(A)} = (0.25, 0.25, 0.25, 0.25, 0, 0, 0, 0).$$

The payoff table for this case is presented in Table 5a, and a corresponding equilibrium defender's best strategy, computed using techniques from [6], is

$$\delta^{*(A)} = (0.25, 0.25, 0, 0.25, 0, 0.25).$$

Note that this equilibrium means that the defender never has to check the bits in the order (2,1,3) or in the order (3,1,2). The game's payoff (i.e., posterior Bayes vulnerability) in this case is 0.5625, which is smaller than the payoff of 0.5833 that would ensue in case the defender's strategy were uniform. This means that, from the point of view of the defender, uniformly randomizing is not optimal.

Now assume that the attacker knows that some passwords are more likely than others, even if all are possible, as reflected in the prior

$$\pi^{(B)} = (0.25, 0.20, 0.15, 0.10, 0.10, 0.10, 0.05, 0.05).$$

The payoff table for this case is presented in Table 5b, and a corresponding equilibrium defender's best strategy can be computed to be

$$\delta^{*(B)} = (0.1974, 0.1974, 0.1316, 0.1316, 0.1711, 0.1711).$$

Note that this equilibrium means that every version of the checker may be selected by the defender, but the probability distribution is not uniform. The game's payoff (i.e., posterior Bayes vulnerability) in this case is 0.4553, which is again smaller than the payoff of 0.4666 that would ensue in case the defender's strategy were uniform.

Table 5. Payoff tables, under different priors, for each pure strategy profile of 3-bit password scenario.(a) Under prior $\pi^{(A)}$.

		Attacker's action a							
		$U(d, a)$	000	001	010	011	100	101	110
Defender's action d	123	0.75	0.75	0.75	0.75	0.25	0.25	0.25	0.25
	132	0.75	0.75	0.75	0.75	0.25	0.25	0.25	0.25
	213	0.75	0.75	0.75	0.75	0.50	0.50	0.50	0.50
	231	0.75	0.75	0.75	0.75	0.75	0.75	0.75	0.75
	312	0.75	0.75	0.75	0.75	0.50	0.50	0.50	0.50
	321	0.75	0.75	0.75	0.75	0.75	0.75	0.75	0.75

(b) Under prior $\pi^{(B)}$.

		Attacker's action a							
		$U(d, a)$	000	001	010	011	100	101	110
Defender's action d	123	0.70	0.70	0.60	0.60	0.50	0.50	0.45	0.4
	132	0.70	0.65	0.70	0.65	0.50	0.50	0.50	0.5
	213	0.70	0.70	0.55	0.55	0.60	0.60	0.50	0.5
	231	0.70	0.70	0.55	0.55	0.70	0.70	0.55	0.5
	312	0.70	0.65	0.70	0.65	0.60	0.60	0.60	0.6
	321	0.70	0.65	0.65	0.60	0.70	0.65	0.65	0.6

8. Related work

Many studies have applied game theory to analyses of security and privacy in networks [18–20], cryptography [21], anonymity [22], location privacy [23], and intrusion detection [24], to cite a few. See [25] for a survey.

In the context of quantitative information flow, most works consider only passive attackers. Boreale and Pampaloni [4] consider adaptive attackers, but not adaptive defenders, and show that in this case the attacker's optimal strategy can be always deterministic. Mardziel et al. [5] propose a model for both adaptive attackers and defenders, but in none of their extensive case-studies the attacker needs a probabilistic strategy to maximize leakage. In this paper we characterize when randomization is necessary, for either attacker or defender, to achieve optimality in our general information leakage games.

Security games have been employed to model and analyze payoffs between interacting agents, especially between a defender and an attacker. Korzhyk et al. [26] theoretically analyze security games and study the relationships between Stackelberg and Nash Equilibria under various forms of imperfect information. Khouzani and Malacaria [27] study leakage properties when perfect secrecy is not achievable due to constraints on the allowable size of the conflating sets, and provide universally optimal strategies for a wide class of entropy measures, and for g -entropies. In particular, they prove that designing a channel with minimum leakage is equivalent to computing Nash equilibria in a corresponding two-player zero-sum games of incomplete information for a range of entropy measures. These works, contrarily to ours, do not consider games with hidden choice, in which optimal strategies differ from traditional game-theory.

Several security games have modeled leakage when the sensitive information are the defender's choices themselves, rather than a system's high input. For instance, Alon et al. [28] propose zero-sum games in which a defender chooses probabilities of secrets and an attacker chooses and learns some of the defender's secrets. Then they present how the leakage on the defender's secrets gives influences on the defender's optimal strategy. More recently, Xu et al. [29] show zero-sum games in which the attacker obtains partial knowledge on the security resources that the defender protects, and provide

the defender's optimal strategy under the attacker's such knowledge. Contrarily to these studies, in this paper we assume that a secret value is drawn from some prior distribution and is not the defender's strategy itself.

Security games have also been used to provide optimal trade-offs between two conflicting desirable properties. Khouzani et al. [30] study the clash between security and usability in the password selection, and present a game-theoretic framework for determining an optimal trade-off. They analyze guessing attacks and derive the optimal policies for secret picking as Nash/Stackelberg Equilibria. Yang et al. [31] propose a game-theoretic framework to analyze user behavior in anonymity networks. They consider the cost of anonymity in terms of the loss of utility. They also consider incentives and their impact on users' cooperation. Shokri et al. [32] present a game-theoretic model for a designer to find the optimal privacy mechanism by taking the adversary's knowledge into account. More specifically, they show a Stackelberg Bayesian game in which a user first chooses a location obfuscation mechanism to maximize his location privacy and then an adversary tries to estimate the user's location to minimize its error. In contrast, our work presents a more general framework that is not limited to a particular domain, and focuses on protocol composition as a method to limit the leakage.

Regarding channel operators, sequential and parallel composition of channels have been studied (e.g., [33]), but we are unaware of any explicit definition and investigation of hidden and visible choice operators. Although Kawamoto et al. [34] implicitly use the hidden choice to model a probabilistic system as the weighted sum of systems, they do not derive the set of algebraic properties we do for this operator, and for its interaction with the visible choice operator.

9. Conclusion and future work

In this paper we used protocol composition to model the introduction of noise performed by the defender to prevent leakage of sensitive information. More precisely, we formalized visible and hidden probabilistic choices of different protocols. We then formalized the interplay between defender and attacker in a game-theoretic framework adapted to the specific issues of QIF, where the payoff is information leakage. We considered various kinds of leakage games, depending on whether players act simultaneously or sequentially, and whether the choices of the defender are visible or not to the attacker. We established a hierarchy of these games, and provided methods for finding the optimal strategies (at the points of equilibrium) in the various cases. We also proved that in a sequential game with hidden choice, the behavioral strategies are more advantageous for the defender than the mixed strategies. This contrast with the standard game theory, where the two types of strategies are equivalent.

As future research, we would like to extend leakage games to the case of repeated observations, i.e., when the attacker can observe the outcomes of the system in successive runs, under the assumption that both attacker and defender may change the channel in each run. We would also like to extend our framework to non zero-sum games, in which the costs of attack and defense are not equivalent, and to analyze differentially-private mechanisms.

Acknowledgments: The authors are thankful to Arman Khouzani and Pedro O. S. Vaz de Melo for valuable discussions. This work was supported by JSPS and Inria under the project LOGIS of the Japan-France AYAME Program, by the PEPS 2018 project MAGIC, and by the project Epistemic Interactive Concurrency (EPIC) from the STIC AmSud Program. Mário S. Alvim was supported by CNPq, CAPES, and FAPEMIG. Yusuke Kawamoto was supported by JSPS KAKENHI Grant Number JP17K12667.

Author Contributions: All authors contributed to the technical results, and are listed in alphabetical order.

Conflicts of Interest: The authors declare no conflict of interest.

Abbreviations

The following abbreviations are used in this manuscript:

QIF Quantitative information flow

Appendix A. Proofs of Technical Results

In this section we provide the proofs of the technical results which are not in the main body of the paper.

Appendix A.1. Preliminaries for proofs

We start by providing some necessary background for the subsequent technical proofs.

Definition 1 states that two compatible channels (i.e., with the same input space) are equivalent if they yield the same value of vulnerability for every prior and every vulnerability function. The result below, from the literature, provides necessary and sufficient conditions for two channels being equivalent. The result employs the extension of channels with an all-zero column as follows. For any channel C of type $\mathcal{X} \times \mathcal{Y} \rightarrow \mathbb{R}$, its *zero-column extension* C^0 is the channel of type $\mathcal{X} \times (\mathcal{Y} \cup \{y_0\}) \rightarrow \mathbb{R}$, with $y_0 \notin \mathcal{Y}$, s.t. $C^0(x, y) = C(x, y)$ for all $x \in \mathcal{X}$, $y \in \mathcal{Y}$, and $C^0(x, y_0) = 0$ for all $x \in \mathcal{X}$.

Lemma A1 (Characterization of channel equivalence [13,17]). *Two channels C_1 and C_2 are equivalent iff every column of C_1 is a convex combination of columns of C_2^0 , and every column of C_2 is a convex combination of columns of C_1^0 .*

Note that the result above implies that for being equivalent, any two channels must be compatible.

Appendix A.2. Proofs of Section 4

Proposition 4.1 (Type of hidden choice). *Given a family $\{C_i\}_{i \in \mathcal{I}}$ of channels of type $\mathcal{X} \times \mathcal{Y} \rightarrow \mathbb{R}$, and a distribution μ on \mathcal{I} , the hidden choice $\sum_{i \leftarrow \mu} C_i$ is a channel of type $\mathcal{X} \times \mathcal{Y} \rightarrow \mathbb{R}$.*

Proof. Since hidden choice is defined as a summation of matrices, the type of $\sum_{i \leftarrow \mu} C_i$ is the same as the type of every C_i in the family.

To see that $\sum_{i \leftarrow \mu} C_i$ is a channel (i.e., all of its entries are non-negative, and all of its rows sum up to 1), first note that, since each C_i in the family is a channel matrix, $C_i(x, y)$ lies in the interval $[0, 1]$ for all $x \in \mathcal{X}$, $y \in \mathcal{Y}$. Since μ is a set of convex coefficients, from the definition of hidden choice it follows that also $\sum_{i \in \mathcal{I}} \mu(i) C_i(x, y)$ must lie in the interval $[0, 1]$ for every x, y .

Second, note that for all $x \in \mathcal{X}$:

$$\begin{aligned} \sum_{y \in \mathcal{Y}} \left(\sum_{i \leftarrow \mu} C_i \right) (x, y) &= \sum_{y \in \mathcal{Y}} \sum_{i \in \mathcal{I}} \mu(i) C_i(x, y) && \text{(def. of hidden choice)} \\ &= \sum_{i \in \mathcal{I}} \mu(i) \sum_{y \in \mathcal{Y}} C_i(x, y) \\ &= \sum_{i \in \mathcal{I}} \mu(i) \cdot 1 && (C_i: \mathcal{X} \times \mathcal{Y} \rightarrow \mathbb{R} \text{ are channels)} \\ &= 1 && (\mu \text{ is a prob. dist.}) \end{aligned}$$

□

Proposition 4.2 (Type of visible choice). *Given a family $\{C_i\}_{i \in \mathcal{I}}$ of compatible channels s.t. each C_i has type $\mathcal{X} \times \mathcal{Y}_i \rightarrow \mathbb{R}$, and a distribution μ on \mathcal{I} , the result of the visible choice $\sqcup_{i \leftarrow \mu} C_i$ is a channel of type $\mathcal{X} \times (\bigsqcup_{i \in \mathcal{I}} \mathcal{Y}_i) \rightarrow \mathbb{R}$.*

Proof. Visible choice applied to a family $\{C_i\}$ of channels scales each matrix C_i by a factor $\mu(i)$, which preserves the type $\mathcal{X} \times \mathcal{Y}_i \rightarrow \mathbb{R}$ of each matrix, and then concatenates all the matrices so produced, yielding a result of type $\mathcal{X} \times (\bigsqcup_{i \in \mathcal{I}} \mathcal{Y}_i) \rightarrow \mathbb{R}$.

To see that $\bigsqcup_{i \leftarrow \mu} C_i$ is a channel (i.e., that all of its entries are non-negative, and that all rows sum-up to 1), note that each element of the visible choice on the family $\{C_i\}$ can be denoted by $(\bigsqcup_{i \leftarrow \mu} C_i)(x, (y, j))$, where $x \in \mathcal{X}$, $j \in \mathcal{I}$, and $y \in \mathcal{Y}_j$. Then, note that for all $x \in \mathcal{X}$, $j \in \mathcal{I}$, and $y \in \mathcal{Y}_j$:

$$\begin{aligned} \left(\bigsqcup_{i \leftarrow \mu} C_i \right)(x, (y, j)) &= (\diamond_{i \in \mathcal{I}} \mu(i) C_i)(x, (y, j)) && \text{(def. of visible choice)} \\ &= (\mu(j) C_j)(x, y) && \text{(def. of concatenation)} \\ &= \mu(j) C_j(x, y) && \text{(def. of scalar mult.)} \end{aligned} \quad (\text{A1})$$

which is a non-negative value, since, both $\mu(j)$ and $C_j(x, y)$ are non-negative.

Finally, note that for all $x \in \mathcal{X}$:

$$\begin{aligned} \sum_{\substack{j \in \mathcal{I} \\ y \in \mathcal{Y}_j}} \left(\bigsqcup_{i \leftarrow \mu} C_i \right)(x, (y, j)) &= \sum_{\substack{j \in \mathcal{I} \\ y \in \mathcal{Y}_j}} \mu(j) C_j(x, y) && \text{(by Eq. (A1))} \\ &= \sum_{j \in \mathcal{I}} \mu(j) \sum_{y \in \mathcal{Y}_j} C_j(x, y) \\ &= \sum_{j \in \mathcal{I}} \mu(j) \cdot 1 && (C_j: \mathcal{X} \times \mathcal{Y}_j \rightarrow \mathbb{R} \text{ are channels}) \\ &= 1 && (\mu \text{ is a prob. dist.}) \end{aligned}$$

□

Proposition 4.3 (Idempotency). *Given a family $\{C_i\}_{i \in \mathcal{I}}$ of channels s.t. $C_i = C$ for all $i \in \mathcal{I}$, and a distribution μ on \mathcal{I} , then: (a) $\sum_{i \leftarrow \mu} C_i = C$; and (b) $\bigsqcup_{i \leftarrow \mu} C_i \approx C$.*

Proof. a) Idempotency of hidden choice:

$$\begin{aligned} \sum_{i \leftarrow \mu} C_i &= \sum_i \mu(i) C_i && \text{(def. of hidden choice)} \\ &= \sum_i \mu(i) C && \text{(since every } C_i = C) \\ &= C \sum_i \mu(i) \\ &= C && \text{(since } \mu \text{ is a prob. dist.)} \end{aligned}$$

b) Idempotency of visible choice:

$$\begin{aligned} \bigsqcup_{i \leftarrow \mu} C_i &= \diamond_i \mu(i) C_i && \text{(def. of visible choice)} \\ &= \diamond_i \mu(i) C && \text{(since every } C_i = C) \\ &\approx C && \text{(by Lemma A1)} \end{aligned}$$

In the above derivation, we can apply Lemma A1 because every column of the channel on each side of the equivalence can be written as a convex combination of the zero-column extension of the channel on the other side.

□

Proposition 4.4 (“Reorganization of operators”). *Given a family $\{C_{ij}\}_{i \in \mathcal{I}, j \in \mathcal{J}}$ of channels indexed by sets \mathcal{I} and \mathcal{J} , a distribution μ on \mathcal{I} , and a distribution η on \mathcal{J} :*

- (a) $\sum_{i \leftarrow \mu} \sum_{j \leftarrow \eta} C_{ij} = \sum_{j \leftarrow \eta} \sum_{i \leftarrow \mu} C_{ij}$, if all C_i 's have the same type;
- (b) $\lfloor \cdot \rfloor_{i \leftarrow \mu} \lfloor \cdot \rfloor_{j \leftarrow \eta} C_{ij} \approx \lfloor \cdot \rfloor_{i \leftarrow \mu} \sum_{j \leftarrow \eta} C_{ij}$, if all C_i 's are compatible; and
- (c) $\sum_{i \leftarrow \mu} \lfloor \cdot \rfloor_{j \leftarrow \eta} C_{ij} \approx \lfloor \cdot \rfloor_{j \leftarrow \eta} \sum_{i \leftarrow \mu} C_{ij}$, if, for each i , all C_{ij} 's have same type $\mathcal{X} \times \mathcal{Y}_j \rightarrow \mathbb{R}$.

Proof. a)

$$\begin{aligned}
\sum_{i \leftarrow \mu} \sum_{j \leftarrow \eta} C_{ij} &= \sum_{i \leftarrow \mu} \left(\sum_j \eta(j) C_{ij} \right) && \text{(def. of hidden choice)} \\
&= \sum_i \mu(i) \left(\sum_j \eta(j) C_{ij} \right) && \text{(def. of hidden choice)} \\
&= \sum_{i,j} \eta(i) \eta(j) C_{ij} && \text{(reorganizing the sums)} \\
&= \sum_{\substack{i \leftarrow \mu \\ j \leftarrow \eta}} C_{ij} && \text{(def. of hidden choice)}
\end{aligned}$$

b)

$$\begin{aligned}
\lfloor \cdot \rfloor_{i \leftarrow \mu} \lfloor \cdot \rfloor_{j \leftarrow \eta} C_{ij} &= \lfloor \cdot \rfloor_{i \leftarrow \mu} (\diamond_j C_{ij}) && \text{(def. of visible choice)} \\
&= \diamond_i \mu(i) (\diamond_j \eta(j) C_{ij}) && \text{(def. of visible choice)} \\
&\approx \diamond_{ij} \mu(i) \eta(j) C_{ij} && \text{(by Lemma A1)} \\
&= \lfloor \cdot \rfloor_{\substack{i \leftarrow \mu \\ j \leftarrow \eta}} C_{ij} && \text{(def. of visible choice)}
\end{aligned}$$

In the above derivation, we can apply Lemma A1 because every column of the channel on each side of the equivalence can be written as a convex combination of the zero-column extension of the channel on the other side.

c)

$$\begin{aligned}
\sum_{i \leftarrow \mu} \lfloor \cdot \rfloor_{j \leftarrow \eta} C_{ij} &= \sum_{i \leftarrow \mu} (\diamond_j C_{ij}) && \text{(def. of visible choice)} \\
&= \sum_i \mu(i) (\diamond_j \eta(j) C_{ij}) && \text{(def. of hidden choice)} \\
&\approx \diamond_j \eta(j) \left(\sum_i \mu(i) C_{ij} \right) && \text{(by Lemma A1)} \\
&= \lfloor \cdot \rfloor_{j \leftarrow \eta} \sum_{i \leftarrow \mu} C_{ij} && \text{(def. of operators)}
\end{aligned}$$

In the above derivation, we can apply Lemma A1 because every column of the channel on each side of the equivalence can be written as a convex combination of the zero-column extension of the channel on the other side.

□

Appendix A.3. Proofs of Section 7

Theorem A1. Consider the password checker program of Algorithm 1 for n -bit passwords, where the attacker controls the low input to the checker, and the defender controls the order in which the bits are checked. If the

prior π on possible passwords is uniform, and the payoff is given by the posterior Bayes-vulnerability $U(\delta, \alpha) = \mathbb{E}_{a \leftarrow \alpha} \mathbb{V}[\pi, \sum_{d \leftarrow \delta} C_{da}]$, then the strategy (δ^*, α) where δ^* is uniform and α is arbitrary is an equilibrium strategy.

Proof. We show that (δ^*, α) where δ^* is uniform and α is arbitrary, is a saddle point of $U(\delta, \alpha)$. The proof will rely on two crucial *symmetries* of the password checker, in combination with the use of a uniform prior. Note that, under uniform π , the Bayes-vulnerability of a channel C is proportional to the sum of the column maxima of C , namely $\mathbb{V}[\pi, C] = \frac{1}{|\mathcal{X}|} \sum_y \max_x C(x, y)$. Given a permutation σ of \mathcal{X} , define C^σ as C with its rows permuted, namely $C^\sigma(x, y) = C(\sigma(x), y)$. Note that C^σ can be written as $M_\sigma C$ where M_σ is a permutation matrix. Permuting the rows does not affect the column maxima, hence $\mathbb{V}[\pi, C] = \mathbb{V}[\pi, C^\sigma]$.

For the attacker things are simple, since we can show that under a uniform π all attacker strategies are equivalent, that is,

$$U(\alpha, \delta) = U(\alpha', \delta) \quad \text{for all } \alpha, \alpha', \delta.$$

To show this, we use the first important symmetry of the password checker, which is due to the fact that the output of the algorithm only depends on whether $x_i \neq a_i$ is true or false for each bit. Hence, any modification on a can be matched with a modification on x that preserves the output, namely for all a, a' there exists a permutation σ of passwords such that $C_{da} = C_{da'}^\sigma$ for all d . Denote $C_{\delta a} = \sum_{d \leftarrow \delta} C_{da}$, we have that

$$C_{\delta a} = \sum_{d \leftarrow \delta} C_{da} = \sum_{d \leftarrow \delta} M_\sigma C_{da'} = M_\sigma \sum_{d \leftarrow \delta} C_{da'} = C_{\delta a'}^\sigma$$

From $\mathbb{V}[\pi, C_{\delta a}] = \mathbb{V}[\pi, C_{\delta a'}^\sigma]$ we have that $U(a, \delta) = U(a', \delta)$ for all pure strategies a, a' , which implies $U(\alpha, \delta) = U(\alpha', \delta)$ since $U(\alpha, \delta)$ is linear on α . So in the remaining of the proof we assume that the attacker plays the fixed pure strategy a_0 , having the zero bitstring 0..0 as the low input.

For the defender, on the other hand, the situation is far from trivial: although all *pure* strategies d are still equivalent, $U(\alpha, \delta)$ is *only convex* on δ and as a consequence the payoff highly depends on δ . Our goal is to show that

$$U(a_0, \delta) = \frac{1}{|\mathcal{X}|} \sum_y \max_x \sum_d \delta(d) C_{da_0}(x, y)$$

is minimized on the uniform δ^* . To do so, we show something stronger, namely that *each addend* of the y -sum is *simultaneously* minimized on the uniform δ^* . Fix an arbitrary y , and let

$$f(\delta) = \max_x \delta \cdot \psi_x$$

where \cdot is the dot product and $\psi_x \in \mathbb{R}^{|\mathcal{D}|}$ is the vector defined by $\psi_x(d) = C_{da_0}(x, y)$. Note that, since C_{da_0} is deterministic, all elements of ψ_x are either 0 or 1. Intuitively, $\psi_x(d) = 1$ if x produces the fixed output y when the bit checking order is d .

We need to show that $f(\delta)$ is minimized on δ^* . However, f seen as a function on the whole $\mathbb{R}^{|\mathcal{D}|}$ has no global minimum. In our case, though, δ is a probability distribution taking values in $\mathbb{D}\mathcal{D} \subset \mathbb{R}^{|\mathcal{D}|}$. That is, only $|\mathcal{D}| - 1$ elements of δ are free, so we can reduce the dimension by one as follows: fix some $d_0 \in \mathcal{D}$ and let $\tilde{\delta}, \tilde{\psi}_x \in \mathbb{R}^{|\mathcal{D}|-1}$ be the same as δ, ψ_x with the element corresponding to d_0 removed. We have that $\delta(d_0) = 1 - \sum_{d \neq d_0} \delta(d) = 1 - \tilde{\delta} \cdot \mathbf{1}$, where $\mathbf{1}$ is the “ones” vector, hence we can rewrite $f(\delta)$ as a function of $\tilde{\delta}$:

$$\begin{aligned} f(\delta) &= \max_x \tilde{\delta} \cdot \tilde{\psi}_x + \delta(d_0) \psi_x(d_0) \\ &= \max_x \tilde{\delta} \cdot \tilde{\psi}_x + (1 - \tilde{\delta} \cdot \mathbf{1}) \psi_x(d_0) \\ &= \max_x \tilde{\delta} \cdot (\tilde{\psi}_x - \psi_x(d_0) \mathbf{1}) + \psi_x(d_0) \end{aligned}$$

So it is sufficient to show that $f(\tilde{\delta})$ is minimized on $\tilde{\delta}^*$.

In the following, we use the fundamental concept of *subgradients* from convex analysis, which generalize gradients for non-differentiable functions. A vector v is a subgradient of a possibly non-differentiable convex function $g : S \rightarrow \mathbb{R}$ at $x_0 \in S$ iff

$$g(x) - g(x_0) \geq v \cdot (x - x_0) \quad \text{for all } x \in S.$$

It is well-known that g has a global minimum on x_0 iff the $\mathbf{0}$ vector belongs to the set of subgradients of g at x_0 (this generalizes the fact that differentiable convex functions have zero gradient on their global minimum). Recall also that $g(x) = x \cdot v + c$ has a single subgradient v , while for $g(x) = \max_i g_i(x)$, any subgradient of g_i for *any branch* i giving the max, is also a subgradient of g . Finally, the set of subgradients is convex, so any convex combination of them is also a subgradient.

Hence, the subgradients of $f(\tilde{\delta})$ are $\tilde{\psi}_x - \psi_x(d_0)\mathbf{1}$, for any x giving the maximum, as well as their convex combinations. Our goal is to show that on $\tilde{\delta}^*$ these include the zero vector.

We finally arrive to the second important symmetry of the password checker. Let ρ be a permutation of the set $\{1, \dots, n\}$ of password bits. Such a permutation could be seen both as a permutation on \mathcal{X} (i.e. $\rho(x) = x_{\rho(1)} \dots x_{\rho(n)}$; note that $\rho(x)$ has the same number of 0s and 1s as x), as well as a permutation on \mathcal{D} (a bit checking order d is *itself* a permutation of bits, so we can set $\rho(d) = \rho \circ d$). Since all low bits of a_0 are the same, applying ρ to both x and d does not change the outcome of the algorithm, that is $C_{da_0} = C_{\rho(d)a_0}^\rho$.

Intuitively, if d is selected uniformly, then the attacker cannot distinguish x from $\rho(x)$ since they produce the same output with the same probability. More concretely, let $x \sim x'$ denote that $x' = \rho(x)$ for some bit permutation ρ . Due to the aforementioned symmetry we have that ψ_x and $\psi_{x'}$ have the same number of 1s which means that $\delta^* \cdot \psi_x = \delta^* \cdot \psi_{x'}$. Now, let $x^* \in \operatorname{argmax}_x \delta^* \cdot \psi_x$. We have that *all* $x \sim x^*$ also give the max, hence all vectors

$$\tilde{\psi}_x - \psi_x(d_0)\mathbf{1} \quad x \sim x^* \tag{A2}$$

are subgradients of f on $\tilde{\delta}^*$. Finally, for any d, d' there is a bit permutation ρ such that $d' = \rho(d)$. Since $\psi_x(d) = \psi_{\rho(x)}(\rho(d))$ we have that $(\sum_{x \sim x^*} \psi_x)(d) = k$ (for some integer k) independently from d . Hence, letting $c = |\{x \mid x \sim x^*\}|$ and averaging all subgradients of (A2) we get that

$$\frac{1}{c} \sum_{x \sim x^*} (\tilde{\psi}_x - \psi_x(d_0)\mathbf{1}) = \frac{1}{c}(k\mathbf{1} - k\mathbf{1}) = \mathbf{0}$$

is also a subgradient, which concludes the proof. \square

Appendix B. Properties of binary versions of channel operators

In this section we derive some relevant properties of the binary versions of the hidden and visible choice operators. We start with results regarding each operator individually.

Proposition B.1 (Properties of the binary hidden choice). *For any channels C_1 and C_2 of the same type, and any values $0 \leq p, q \leq 1$, the binary hidden choice operator satisfies the following properties:*

- a) Idempotency: $C_1 \underset{p}{\oplus} C_1 = C_1$
- b) Commutativity: $C_1 \underset{p}{\oplus} C_2 = C_2 \underset{\bar{p}}{\oplus} C_1$
- c) Associativity:

$$C_1 \underset{p}{\oplus} (C_2 \underset{q}{\oplus} C_3) = (1/q \cdot C_1 \underset{p}{\oplus} C_2) \underset{q}{\oplus} \bar{p} \cdot C_3$$

if $q \neq 0$.

- d) Absorption:

$$(C_1 \underset{p}{\oplus} C_2) \underset{q}{\oplus} (C_1 \underset{r}{\oplus} C_2) = C_1 \underset{(pq+\bar{q}r)}{\oplus} C_2.$$

Proof. We will prove each property separately.

a) *Idempotency*:

$$\begin{aligned} C_1 \underset{p}{\oplus} C_1 &= p \cdot C_1 + \bar{p} \cdot C_1 && \text{(def. of hidden choice)} \\ &= (p + \bar{p}) \cdot C_1 \\ &= C_1 && (p + \bar{p} = 1) \end{aligned}$$

b) *Commutativity*:

$$\begin{aligned} C_1 \underset{p}{\oplus} C_2 &= p \cdot C_1 + \bar{p} \cdot C_2 && \text{(def. of hidden choice)} \\ &= \bar{p} \cdot C_2 + p \cdot C_1 \\ &= C_2 \underset{\bar{p}}{\oplus} C_1 && \text{(def. of hidden choice)} \end{aligned}$$

c) *Associativity*:

$$\begin{aligned} &C_1 \underset{p}{\oplus} (C_2 \underset{q}{\oplus} C_3) \\ &= p \cdot C_1 + \bar{p}(q \cdot C_2 + \bar{q} \cdot C_3) && \text{(def. of hidden choice)} \\ &= p \cdot C_1 + \bar{p}q \cdot C_2 + \bar{p}\bar{q} \cdot C_3 \\ &= q(p(1/q) \cdot C_1) + \bar{p} \cdot C_2 + \bar{q}(\bar{p} \cdot C_3) \\ &= (1/q) \cdot C_1 \underset{p}{\oplus} C_2 \underset{q}{\oplus} \bar{p} \cdot C_3 && \text{(def. of hidden choice)} \end{aligned}$$

d) *Absorption*: First note that:

$$\begin{aligned} &(C_1 \underset{p}{\oplus} C_2) \underset{q}{\oplus} (C_1 \underset{r}{\oplus} C_2) \\ &= q(pC_1 + \bar{p}C_2) + \bar{q}(rC_1 + \bar{r}C_2) && \text{(def. of hidden choice)} \\ &= pqC_1 + \bar{p}qC_2 + \bar{q}rC_1 + \bar{q}\bar{r}C_2 \\ &= (pq + \bar{q}r)C_1 + (\bar{p}q + \bar{q}\bar{r})C_2 \\ &= C_1 \underset{(pq+\bar{q}r)}{\oplus} C_2 && (*) \end{aligned}$$

To complete the proof, note that in step (*) above, $pq + \bar{q}r$ and $\bar{p}q + \bar{q}\bar{r}$ form a valid binary probability distribution (they are both non-negative, and they add up to 1), then apply the definition of hidden choice.

□

Proposition B.2 (Properties of binary visible choice). *For any compatible channels C_1 and C_2 , and any values $0 \leq p, q \leq 1$, the visible choice operator satisfies the following properties:*

- a) *Idempotency*: $C_1 \underset{p}{\sqcup} C_1 \approx C_1$
- b) *Commutativity*: $C_1 \underset{p}{\sqcup} C_2 \approx C_2 \underset{\bar{p}}{\sqcup} C_1$
- c) *Associativity*: $C_1 \underset{p}{\sqcup} (C_2 \underset{q}{\sqcup} C_3) \approx (1/q) \cdot C_1 \underset{p}{\sqcup} C_2 \underset{q}{\sqcup} \bar{p} \cdot C_3$ if $q \neq 0$.

Proof. We will prove each property separately.

- a) *Idempotency*: $C_1 \underset{p}{\sqcup} C_1 \approx C_1$, by immediate application of Lemma A1, since every column of the channel on each side of the equivalence can be written as a convex combination of the zero-column extension of the channel on the other side.

b) *Commutativity*:

$$\begin{aligned}
C_1 \mathbin{p}\sqcup C_2 &= p \cdot C_1 \diamond \bar{p} \cdot C_2 && \text{(def. of visible choice)} \\
&\approx \bar{p} \cdot C_2 \diamond p \cdot C_1 && \text{(by Lemma A1)} \\
&= C_2 \mathbin{\bar{p}}\sqcup C_1 && \text{(def. of visible choice).}
\end{aligned}$$

In the above derivation, we can apply Lemma A1 because every column of the channel on each side of the equivalence can be written as a convex combination of the zero-column extension of the channel on the other side.

c) *Associativity*:

$$\begin{aligned}
C_1 \mathbin{p}\sqcup (C_2 \mathbin{q}\sqcup C_3) &= p \cdot C_1 \diamond \bar{p}(q \cdot C_2 \diamond \bar{q} \cdot C_3) && \text{(def. of visible choice)} \\
&\approx p \cdot C_1 \diamond (\bar{p}q \cdot C_2 \diamond \bar{p}\bar{q} \cdot C_3) && \text{(by Lemma A1)} \\
&= q(p \mathbin{1/q}\cdot C_1 \diamond \bar{p} \cdot C_2) \diamond \bar{q}(\bar{p} \cdot C_3) \\
&= (1/q \cdot C_1 \mathbin{p}\sqcup C_2) \mathbin{q}\sqcup \bar{p} \cdot C_3 && \text{(def. of visible choice)}
\end{aligned}$$

In the above derivation, we can apply Lemma A1 because every column of the channel on each side of the equivalence can be written as a convex combination of the zero-column extension of the channel on the other side.

□

Now we turn our attention to the interaction between hidden and visible choice operators.

A first result is that hidden choice does not distribute over visible choice. To see why, note that $C_1 \mathbin{p}\oplus (C_2 \mathbin{q}\sqcup C_3)$ and $(C_1 \mathbin{p}\oplus C_2) \mathbin{q}\sqcup (C_1 \mathbin{p}\oplus C_3)$ cannot be both defined: if the former is defined, then C_1 must have the same type as $C_2 \mathbin{q}\sqcup C_3$, whereas if the latter is defined, C_1 must have the same type as C_2 , but $C_2 \mathbin{q}\sqcup C_3$ and C_2 do not have the same type (they have different output sets).

However, as the next result shows, visible choice distributes over hidden choice.

Proposition B.3 (Distributivity of $\mathbin{p}\sqcup$ over $\mathbin{q}\oplus$). *Let C_1, C_2 and C_3 be compatible channels, and C_2 and C_3 have the same type. Then, for any values $0 \leq p, q \leq 1$:*

$$C_1 \mathbin{p}\sqcup (C_2 \mathbin{q}\oplus C_3) \approx (C_1 \mathbin{p}\sqcup C_2) \mathbin{q}\oplus (C_1 \mathbin{p}\sqcup C_3).$$

Proof.

$$\begin{aligned}
&C_1 \mathbin{p}\sqcup (C_2 \mathbin{q}\oplus C_3) \\
&= (C_1 \mathbin{q}\oplus C_1) \mathbin{p}\sqcup (C_2 \mathbin{q}\oplus C_3) && \text{(idempotency of visible choice)} \\
&= p(q \cdot C_1 + \bar{q} \cdot C_1) \diamond \bar{p}(q \cdot C_2 + \bar{q} \cdot C_3) && \text{(def. of operators)} \\
&= (pq \cdot C_1 + p\bar{q} \cdot C_1) \diamond (\bar{p}q \cdot C_2 + \bar{p}\bar{q} \cdot C_3) \\
&\approx (pq \cdot C_1 \diamond \bar{p}q \cdot C_2) + (p\bar{q} \cdot C_1 \diamond \bar{p}\bar{q} \cdot C_3) && \text{(by Lemma A1)} \\
&= q(p \cdot C_1 \diamond \bar{p} \cdot C_2) + \bar{q}(p \cdot C_1 \diamond \bar{p} \cdot C_3) \\
&= q(C_1 \mathbin{p}\sqcup C_2) + \bar{q}(C_1 \mathbin{p}\sqcup C_3) && \text{(def. of visible choice)} \\
&= (C_1 \mathbin{p}\sqcup C_2) \mathbin{q}\oplus (C_1 \mathbin{p}\sqcup C_3) && \text{(def. of hidden choice)}
\end{aligned}$$

In the above derivation, we can apply Lemma A1 because every column of the channel on each side of the equivalence can be written as a convex combination of the zero-column extension of the channel on the other side. \square

1. Sun, Q.; Simon, D.R.; Wang, Y.M.; Russell, W.; Padmanabhan, V.N.; Qiu, L. Statistical identification of encrypted web browsing traffic. *Proc. of S&P. IEEE*, 2002, pp. 19–30.
2. Dwork, C.; Mcsherry, F.; Nissim, K.; Smith, A. Calibrating noise to sensitivity in private data analysis. *Proc. of TCC. Springer*, 2006, Vol. 3876, LNCS, pp. 265–284.
3. Chaum, D. The Dining Cryptographers Problem: Unconditional Sender and Recipient Untraceability. *Journal of Cryptology* **1988**, *1*, 65–75.
4. Boreale, M.; Pampaloni, F. Quantitative information flow under generic leakage functions and adaptive adversaries. *Logical Methods in Computer Science* **2015**, *11*.
5. Mardziel, P.; Alvim, M.S.; Hicks, M.W.; Clarkson, M.R. Quantifying Information Flow for Dynamic Secrets. *Proc. of S&P*, 2014, pp. 540–555.
6. Alvim, M.S.; Chatzikokolakis, K.; Kawamoto, Y.; Palamidessi, C. Information Leakage Games. *Proc. of GameSec. Springer*, 2017, Vol. 10575, LNCS, pp. 437–457.
7. Rizzo, J.; Duong, T. The CRIME attack, 2012.
8. Alvim, M.S.; Chatzikokolakis, K.; McIver, A.; Morgan, C.; Palamidessi, C.; Smith, G. Axioms for Information Leakage. *Proc. of CSF*, 2016, pp. 77–92.
9. Smith, G. On the Foundations of Quantitative Information Flow. *Proc. of FOSSACS. Springer*, 2009, Vol. 5504, LNCS, pp. 288–302.
10. Chatzikokolakis, K.; Palamidessi, C.; Panangaden, P. On the Bayes risk in information-hiding protocols. *J. of Comp. Security* **2008**, *16*, 531–571.
11. Shannon, C.E. A Mathematical Theory of Communication. *Bell System Technical Journal* **1948**, *27*, 379–423, 625–56.
12. Massey. Guessing and Entropy. *Proceedings of the IEEE Int. Symposium on Information Theory. IEEE*, 1994, p. 204.
13. Alvim, M.S.; Chatzikokolakis, K.; Palamidessi, C.; Smith, G. Measuring Information Leakage Using Generalized Gain Functions. *Proc. of CSF*, 2012, pp. 265–279.
14. Alvim, M.S.; Chatzikokolakis, K.; Kawamoto, Y.; Palamidessi, C. Leakage and protocol composition in a game-theoretic perspective. *Proceedings of POST. Springer*, 2018, LNCS.
15. Osborne, M.J.; Rubinstein, A. *A Course in Game Theory*; The MIT Press, 1994.
16. Braun, C.; Chatzikokolakis, K.; Palamidessi, C. Quantitative Notions of Leakage for One-try Attacks. *Proc. of MFPS. Elsevier*, 2009, Vol. 249, ENTCS, pp. 75–91.
17. McIver, A.; Morgan, C.; Smith, G.; Espinoza, B.; Meinicke, L. Abstract Channels and Their Robust Information-Leakage Ordering. *Proc. of POST. Springer*, 2014, Vol. 8414, LNCS, pp. 83–102.
18. Basar, T. The Gaussian test channel with an intelligent jammer. *IEEE Trans. Information Theory* **1983**, *29*, 152–157.
19. Grossklags, J.; Christin, N.; Chuang, J. Secure or Insure?: A Game-theoretic Analysis of Information Security Games. *Proc. of WWW*, 2008, pp. 209–218.
20. Alpcan, T.; Buchegger, S. Security Games for Vehicular Networks. *IEEE Trans. Mob. Comput.* **2011**, *10*, 280–290.
21. Katz, J. Bridging Game Theory and Cryptography: Recent Results and Future Directions. *Proc. of TCC*, 2008, pp. 251–272.
22. Acquisti, A.; Dingedine, R.; Syverson, P.F. On the Economics of Anonymity. *Proc. of FC*, 2003, pp. 84–102.
23. Freudiger, J.; Manshaei, M.H.; Hubaux, J.P.; Parkes, D.C. On Non-cooperative Location Privacy: A Game-theoretic Analysis. *Proc. of CCS*, 2009, pp. 324–337.
24. Zhu, Q.; Fung, C.J.; Boutaba, R.; Basar, T. A game-theoretical approach to incentive design in collaborative intrusion detection networks. *Proc. of GAMENETS. IEEE*, 2009, pp. 384–392.
25. Manshaei, M.H.; Zhu, Q.; Alpcan, T.; Bacşar, T.; Hubaux, J.P. Game Theory Meets Network Security and Privacy. *ACM Comput. Surv.* **2013**, *45*, 25:1–25:39.

26. Korzhyk, D.; Yin, Z.; Kiekintveld, C.; Conitzer, V.; Tambe, M. Stackelberg vs. Nash in Security Games: An Extended Investigation of Interchangeability, Equivalence, and Uniqueness. *J. Artif. Intell. Res.* **2011**, *41*, 297–327.
27. Khouzani, M.H.R.; Malacaria, P. Relative Perfect Secrecy: Universally Optimal Strategies and Channel Design. *Proc. of CSF*, 2016, pp. 61–76.
28. Alon, N.; Emek, Y.; Feldman, M.; Tennenholtz, M. Adversarial Leakage in Games. *SIAM J. Discrete Math.* **2013**, *27*, 363–385.
29. Xu, H.; Jiang, A.X.; Sinha, A.; Rabinovich, Z.; Dughmi, S.; Tambe, M. Security Games with Information Leakage: Modeling and Computation. *Proc. of IJCAI*, 2015, pp. 674–680.
30. Khouzani, M.H.R.; Mardziel, P.; Cid, C.; Srivatsa, M. Picking vs. Guessing Secrets: A Game-Theoretic Analysis. *Proc. of CSF*, 2015, pp. 243–257.
31. Yang, M.; Sassone, V.; Hamadou, S. A Game-Theoretic Analysis of Cooperation in Anonymity Networks. *Proc. of POST*, 2012, pp. 269–289.
32. Shokri, R.; Theodorakopoulos, G.; Troncoso, C. Privacy Games Along Location Traces: A Game-Theoretic Framework for Optimizing Location Privacy. *ACM Trans. Priv. Secur.* **2017**, *19*, 11:1–11:31.
33. Kawamoto, Y.; Chatzikokolakis, K.; Palamidessi, C. On the Compositionality of Quantitative Information Flow. *Logical Methods in Computer Science* **2017**, *13*, 1–31.
34. Kawamoto, Y.; Biondi, F.; Legay, A. Hybrid Statistical Estimation of Mutual Information for Quantifying Information Flow. *Proc. of FM 2016*, 2016, pp. 406–425.

© 2018 by the authors. Accepted to *Entropy* for possible open access publication under the terms and conditions of the Creative Commons Attribution (CC BY) license (<http://creativecommons.org/licenses/by/4.0/>).