FlipThem: Modeling Targeted Attacks with FlipIt for Multiple Resources

Aron Laszka[1], Gabor Horvath[2], Mark Felegyhazi[1], Levente Buttyan[1]

[1]Laboratory of Cryptography and System Security (CrySyS Lab)
[2]Analysis, Design and Development of ICT systems Laboratory (AddICT Lab)

Department of Networked Systems and Services
Budapest University of Technology and Economics

Abstract

The attack on RSA [12] in early 2011 represented a big surprise for the IT security industry. It showed that major security companies are attractive targets for stealthy attacks because of the important information they possess. The RSA attack induced an interesting discussion in the security industry and the research community alike. In particular, researchers at RSA modeled stealthy takeovers of a resource in their FlipIt game [13]. FlipIt is an attacker-defender game in which the players compete for the control of a resource, which can correspond to the practical case of updating and compromising a cryptographic key.

In this paper, we present FlipThem, a generalization of the FlipIt game to multiple resources. In particular, we consider two control models: In the AND control model, the attacker needs to compromise all resources to gain access to the target system, whereas in the OR control model, the attacker only needs to control a single resource to reach her goal. First, we propose combinations of basic, single-resource FlipIt strategies and study the best choices for the defender and the attacker. Then, we extend these basic strategies with the Markov strategy class to represent more complex combinations of moves.

Based on our FlipThem model, we can provide a few guidelines for the defenders. First, in the AND control model, we found that the defender should update her resources independently. On the other hand, the defender should generally update her resources synchronously in the OR control model. We also found that periodically updating resources is a good choice against a non-adaptive attacker in the FlipThem model, however, it suffers from the same weaknesses against an attacker with feedback as in the basic FlipIt model. Thus, the defender needs to carefully assess the potential information available to the attacker when choosing her strategy. In summary, our results enable a defender to plan her defense strategy against a range of attacker strategies.

Keywords: FlipIt, game theory, advanced persistent threats, targeted attacks, attacker-defender games

1 Introduction

In recent years, the world witnessed a series of high-profile targeted attacks against critical infrastructure, governmental organizations or key security companies. The attack on RSA [12] and several CAs [4, 6, 10] showed that even the networks of major security companies can be compromised. In case of RSA, the information on about 40 million SecureID token were stolen and that forced the company to issue a massive change of this product. These security companies are appealing for targeted attacks, because the credentials obtained in such attacks are key enablers of subsequent attacks against other high-profile targets. Indeed, it was later discovered that the stolen information in the RSA breach was later used to attack Lockheed Martin, a major security defense company [5, 9], although Lockheed Martin claims to have blocked the attempt using their internal security defense solution [3].

A few distinguishing properties make the RSA attack and similar targeted attacks stand out from the sea of malware incidents. First and most importantly, these attacks are politically motivated (e.g. by cyber-espionage) and hence they incorporated a *substantial development effort* with many advanced security properties. Second, these attacks have a small attack footprint, *stealthy operation* and *persist* for an extended duration, in some case for years. In the case of Red October, a malware campaign targeting governmental and diplomatic organizations, the estimated operation has been at least five years [7]. Because of the long operation, recent targeted malware attacks were widely referred to as advanced persistent threat (APT). Third, during the stealthy operation, the persistent malware *collects a substantial amount of information* from the target systems. Fourth, the malware typically uses *advanced attacking techniques* with a combination of zero-day vulnerabilities cutting-edge cryptography methods to penetrate systems that are highly protected, maybe use and air-gap to separate critical functions into an internal computer network. Stuxnet, Duqu and Flame demonstrated that air-gap protection as it was implemented is ineffective. For example, Stuxnet was capable of infecting computers via a USB key [1] and Flame further developed this capability as it was able to transmit data out from an isolated network using infected USB keys [1]. Furthermore, Flame was using advanced cryptographic techniques that allowed it to masquerade as a legitimate MS Windows update. Gauss [1] used a specific key generation method to render all decryption effort ineffective. The combination of these advanced techniques makes traditional security defenses (perimeter defenses, anti-virus products, intrusion detection systems, etc.) ineffective and the malware can exist undetected in spite of their operation.

These sophisticated targeted attacks were eye-openers to the security industry and intensified the efforts in targeted malware detection. Both established security companies and a new breed of startups started to work on detecting zero-day attacks and advanced threats. Researchers at RSA in particular modeled stealthy targeted attacks using game theory in their `FlipIt` game [13]. They call their model the `FlipIt` game, because the attacker and a defender fight over the control of a resource by "flipping" it for a certain cost without being able to observe who is controlling the resource before the flip happens. Thus in this model, the control over the resource is not guaranteed over an extended time period. This model is ideal to model a security resource with offline properties, such as the use and misuse of crytpographic keys. We detail the FlipIt model later in Section 2. In [11], the basic `FlipIt` game is extended by giving the players the option to "test" if they control the resource before making a move. The extended model is used to study periodic security assessments and their positive effects. To the best of our knowledge, there are no other extensions

In this paper, we extend the original `FlipIt` game for multiple resources. Existing work [13, 2, 11] studies a single resource, yet in practice the security of a key asset depends on multiple resources that an attacker has to compromise at the same time. Also, the severity of an attack typically depends on the number of compromised assets. This property is the key motivation to build our model, which we call `FlipThem`.

We make the following contributions in this paper:

- We extend the `FlipIt` game to multiple resources. To formulate the players' goals, we introduce two control models, namely the AND and the OR control model.
- To devise good multi-resource `FlipThem` strategies, we introduce two combinations of single-resource `FlipIt` strategies, namely the independent and the synchronous combination. We study and compare the two combinations and derive analytical results on the players' gains.
- To represent more complex multi-resource strategies, we introduce the Markov strategy class. We show how the best-response Markov strategy can be computed using a linear program. Using this linear program, we compare various defender strategies based on the resulting benefit for the defender.
- Based on our analytical and numerical results, we provide provide practical recommendations for defenders.

The organization of this paper is the following: In Section 2, we summarize the `FlipIt` game and the most important conclusions drawn in related work. In Section 3, we introduce `FlipThem`, the generalization

of FlipIt for multiple resources. In Section 4, we show how single-resource (i.e., basic FlipIt) strategies can be combined into multi-resource strategies and compute the players' benefits for various combinations. In Section 5, we introduce the Markov strategy class and show how a best-response Markov strategy can be computed using a linear program. In Section 6, we discuss the implications of our results and provide practical recommendations for defenders. Finally, in Section 7, we outline open research questions.

2 The FlipIt Game

In this section, we first summarize the FlipIt game. We also mention the most important conclusions drawn in related work. It is important to get familiar with the key concepts and notation of the original FlipIt game to understand our results for the multiple resources case. Table 1 contains the most important difference in notation between the original FlipIt game and our FlipThem game.

Table 1: List of Symbols

Symbol	Description
	FlipIt
c^i	player i's flipping cost
β^i	" asymptotic benefit rate
γ^i	" " gain rate
α^i	" " flip rate
Z^i	random variable representing the time since the last flip of player i
	FlipThem
N	number of resources
c_r^i	player i's flipping cost for resource r
α_r^i	" asymptotic flip rate for resource r
Z_r^i	random variable representing the time since the last flip of player i on resource r

FlipIt [13, 2] is a two-player, zero-sum game modeling stealthy takeovers, in which both players are trying take control of a single resource. One of the players is called the *defender* (denoted by D), while the other player is called the *attacker* (denoted by A). The game starts at time $t = 0$ and continues indefinitely (that is, $t \to \infty$). In general, time can be both continuous and discrete, with most results being applicable to both cases. At any time instance, player i may choose to take control of the resource by "flipping" it, which costs her c^i. Then, the resource remains under the control of player i until the other player flips it. Consequently, at any given time instance, the resource is controlled by either one or the other player. The interesting aspect of the FlipIt game is that neither of the players knows who is in control. As a result, the players occasionally make unnecessary flips (i.e., flip the resource when it is already under their control) since they have to execute their flips "blindly". For an illustration of the game, see Figure 1.

The state of the game is represented by the time dependent variables C^A and C^D: $C^A(t) = 1$ when the attacker controls the resource, and 0 otherwise; $C^D(t)$ is vice versa (i.e., $C^D(t) = 1 - C^A(t)$). Since players can (and, as we will soon see, should) employ randomized strategies, both $C^D(t)$ and $C^A(t)$ are random variables. The variables $C^D(t)$ and $C^A(t)$ can be also expressed using the times elapsed since the last flips made by the players as

$$C^D(t) = I_{Z^D(t) \leq Z^A(t)} \tag{1}$$

and

$$C^A(t) = I_{Z^D(t) > Z^A(t)} \ , \tag{2}$$

where Z^i is the time elapsed since the last flip of player i and I is the indicator function.

Player i's *asymptotic gain rate* γ^i is defined as the average fraction of time the resource is controlled by player i. Formally,

$$\gamma^i = \liminf_{t \to \infty} \frac{\int_0^t C^i(\tau) d\tau}{t} \ . \tag{3}$$

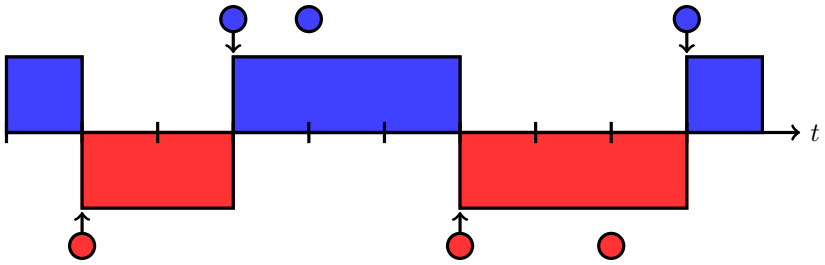


Figure 1: An illustration of the `FlipIt` game when flip timing is discrete. Blue and red circles represent the defender's and attacker's flips. Takeovers, that is flips changing the control of the resource, are indicated by arrows. Blue and red shaded rectangles represent control of the resource by the defender and the attacker, respectively.

Note that player i's asymptotic gain is equal to the probability that the resource is controlled by player i at a random time instance. Formally,

$$\gamma^i = \Pr\left[C^i = 1\right] \ . \tag{4}$$

Player i's *asymptotic flip rate* α^i is defined as the average number of flips made by player i in a unit of time. Formally,

$$\alpha^i = \liminf_{t\to\infty} \frac{n^i(t)}{t} \ , \tag{5}$$

where $n^i(t)$ denotes the number of flips made by player i up to time t. Finally, player i's game-theoretic utility, called player i's *asymptotic benefit* β^i, is defined as the average fraction of time the resource is controlled by the player minus the average cost of flips. Formally,

$$\beta^i = \gamma^i - c^i\alpha^i \ . \tag{6}$$

Since takeovers are assumed to be stealthy in the `FlipIt` game, players do not automatically know when the other player has last moved. However, when a player makes a move (i.e., flips the resource), she might be able to receive some feedback. For example, when an attacker compromises a system, she may learn when the defender last updated the system (that could be attributed as a flip action), and use this information to plan her next move. In [13], three models are introduced for *feedback* received by a player *during the game*:

- Non-adaptive (NA): The player does not receive any feedback when she moves.
- Last move (LM): The player learns the exact time of the other player's last flip.
- Full history (FH): The player learns the complete history of flips made by the other player.

Besides receiving feedback during the game, a player might also be able to receive information before the game starts. For example, an attacker might learn the defender's flip strategy and exploit this knowledge. In [13], two models are introduced for *information* received by a player *before the game starts*:

- Rate of Play (RP): The player knows the asymptotic flip rate α of the other player.
- Knowledge of Strategy (KS): Besides the asymptotic flip rate, the player knows additional information about the other player's strategy. For example, the player may know that the other player employs a renewal process to generate her flip sequence, and may also know the probability density function of the process. However, it is always assumed that the randomness of the other player's strategy remains secret; consequently, the player cannot know which realization of the renewal process will be used.

In our analysis of defender's strategies in Section 5, we assume a strong attacker model meaning that the attacker always has the Knowledge of Strategy. We assume that the attacker knows everything, except the randomness part of the defender's strategy. This complies with Kerckhoff's principle on security without obscurity.

2.1 Strategies

In this subsection, we summarize the most important strategies and the corresponding results from [13]. For a detailed analysis of these and some other strategies, we refer the reader to [13].

In this paper, we focus on non-adaptive strategies, which do not require feedback received by the player during the game. The rationale behind this is that

- defenders rarely know the exact strategies of the attackers (or even the identities of the attackers) in practice; thus, they have to use strategies that do not rely on feedback,
- defenders can choose randomized strategies that schedule their subsequent flips such that even an FH attacker has no more advantage than random guessing (see exponential strategy below), and
- in case of high-importance computer systems, attackers might have limited feedback options if they want to operate stealthily.

A renewal strategy is a non-adaptive strategy in which the time intervals between consecutive flips are generated by a renewal process. More formally, time intervals between consecutive moves are independent and identically distributed random variables chosen according to a probability density function f. Renewal strategies include (but are not limited to) *periodic strategies* and *non-arithmetic renewal strategies*, which we discuss below.

A player can also choose to drop out of the game (i.e., never flip the resource), which is a rational decision if her expected benefit is less than zero for every strategy choice available to her. This can happen when her opponent's flipping cost is much lower and her opponent can afford to flip the resource extremely fast.

Periodic $\mathcal{P}$: A strategy is periodic if the time intervals between consecutive flips are constant. It is assumed that a periodic strategy has a *random phase*, that is, the time of the first flip is chosen uniformly at random from $[0, \delta]$. A *periodic strategy with random phase* is characterized by the fixed time interval between consecutive flips, denoted by δ. It is easy to see that the flip rate of a periodic strategy is $\alpha = \frac{1}{\delta}$. The periodic strategy of rate α is denoted by P_α, and the class of all periodic strategies is denoted by $\mathcal{P}$.

Periodic is probably the strategy most widely used in practice as most systems require passwords, cryptographic keys, etc. to be changed at regular intervals, for example, every thirty days or every three months. In [13], it was shown that the periodic strategy strongly dominates all other renewal strategies if the other player uses a periodic or non-arithmetic renewal strategy. Thus, the periodic strategy is a good choice for an attacker who plays against a non-adaptive (NA) defender.

However, due to its completely deterministic nature[1], the periodic strategy is a very poor choice for defenders who face an attacker observing the last move of the defender (LM attacker). An LM attacker can learn the exact time of the defender's next flip, and schedule her own flip to be immediately after that. Consequently, if flipping costs are of the same order of magnitude, an attacker can keep the resource permanently under her control (with negligible interrupts from the defender). Therefore, a defender facing an LM attacker has two options: if her flipping cost is much lower than that of the attacker, she can flip fast enough to force the attacker to drop out; otherwise, she has to use a randomized strategy, such as the following ones.

Non-arithmetic renewal $\mathcal{R}$: A renewal process is called *non-arithmetic* if there is no positive real number $d > 0$ such that interarrival times are all the integer multiples of d. The renewal strategy generated by the non-arithmetic renewal process with probability density function f is denoted by R_f, and the class of all non-arithmetic renewal strategies is denoted by $\mathcal{R}$.

The class of non-arithmetic renewal strategies is very broad as there are an infinite number of possible probability density functions, even for a given flip rate. Of these probability density functions, the exponential is the most important one in the `FlipIt` game:

[1] The random phase ensures that an NA opponent cannot determine the flip times of the player; however, if the opponent learns the exact time of at least one flip made by the player, she is able to determine the time of every flip.

Exponential (Poisson) $\mathcal{E}$: An *exponential* (or *Poisson*) strategy is a non-arithmetic renewal strategy generated by a Poisson process. Formally, the interarrival times of the process follow an exponential distribution:

$$f(\tau) = \lambda e^{-\lambda\tau} \ , \tag{7}$$

where λ is the parameter characterizing the distribution. The flip rate of this strategy is simply $\alpha = \lambda$. The exponential strategy with rate λ is denoted by E_λ, and the class of all exponential strategies is denoted by $\mathcal{E}$.

The exponential strategy is of key importance, because the exponential distribution is the only *memoryless* continuous probability distribution. The memoryless property means that the conditional probability that we have to wait more than τ_1 time before the next flip, given that the time elapsed since the last flip is τ_2, is independent of τ_2. This implies that, if a defender uses an exponential strategy, an LM (or even an FH) attacker cannot learn *any* information regarding the time of the defender's next flip. Consequently, the exponential strategy is a good choice for a defender facing an LM attacker.

3 The `FlipThem` Game: `FlipIt` on Multiple Resources

In this section, we generalize the `FlipIt` game for multiple resources as follows. There are N resources, identified by integer numbers $1, \ldots, N$. Each resource can be flipped individually and, as a result, becomes controlled by the flipping player. The cost of flipping resource r for player i is c_r^i. Each resource has to be flipped individually; i.e., if a player chooses to flip multiple resources at the same time, she still has to pay the flipping cost for each resource that she flips.

The goal of the attacker is to control the system of resources, while the goal of defender is to prevent the attacker from doing so. The criterion for the attacker controlling the system can be defined in multiple ways. In this paper, we study two prototypical control models (see Figure 2 for an illustration):

- All resources [AND]: The attacker controls the system only if she controls *all* resources. Formally,

$$C^i(t) \ = \ Z_1^D(t) > Z_1^A(t) \ \wedge \ldots \wedge \ Z_N^D(t) > Z_N^A(t) \ . \tag{8}$$

  This models scenarios where the attacker has to compromise every resource (e.g., multiple passwords) in order to compromise her target.

- One resource [OR]: The attacker controls the system if she controls *at least one* resource. Formally,

$$C^i(t) \ = \ Z_1^D(t) > Z_1^A(t) \ \vee \ldots \vee \ Z_N^D(t) > Z_N^A(t) \ . \tag{9}$$

  This models scenarios where the attacker only has to compromise a single resource in order to compromise her target.

Similarly to the basic `FlipIt` game, the players receive benefits proportional to the time that they are controlling the system minus their costs of flipping the resources.

Notice that, for non-adaptive strategies, the two control models are completely symmetric: the benefit of one player in one model is equivalent to the benefit of the other player in the other model. Consequently, for non-adaptive strategies, it suffices to compute the benefits only in one control model (the AND model in our paper) as the formulas for the other model can be derived readily.

In the following sections, we introduce and study various `FlipThem` (i.e., multi-resource) strategies, compute the resulting asymptotic benefits, and discuss which strategies should be chosen by the players. First, in Section 4, we study combinations of multiple single-resource strategies. Then, in Section 5, we propose a novel multi-resource strategy class, called the *Markov strategy* class.

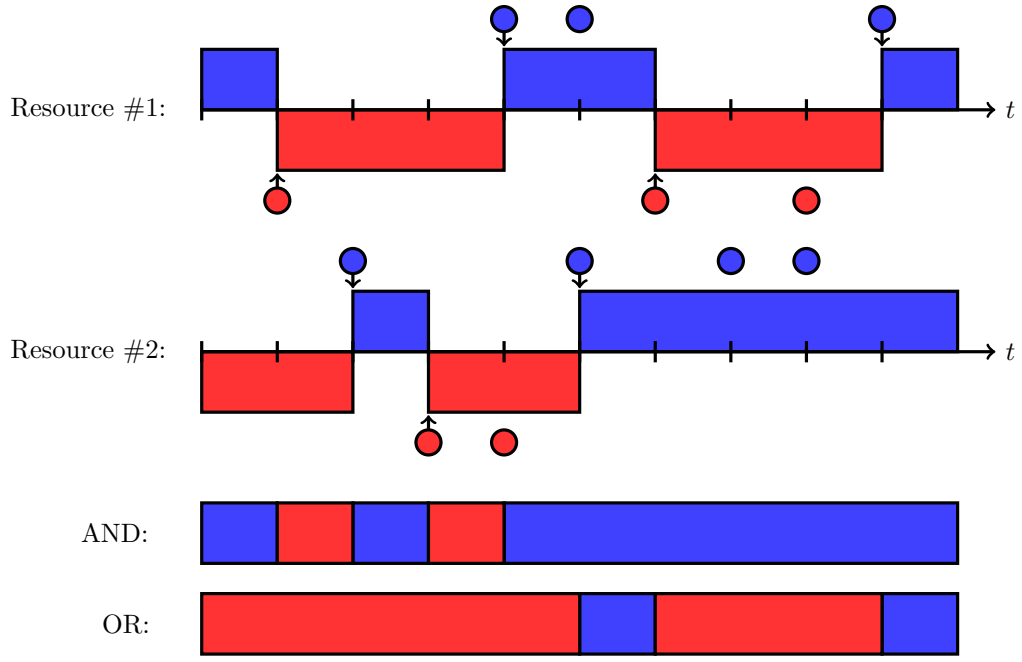


Figure 2: An illustration of the `FlipThem` game with the AND and OR control models (see Figure 1 for graphical notations).

4 Combining Single-Resource Strategies

An obvious way of finding good multi-resource strategies is to combine multiple single-resource strategies that are known to perform well in the basic `FlipIt` game. In this section, we study the two most straightforward combinations:

- *Independent:* The player flips each resource independently of the other resources. More specifically, the player uses N independent single-resource strategies (i.e., processes), one for each resource, with each one having its own flip rate α^i. The asymptotic benefit of a player i using the independent combination is

$$\beta^i = \gamma^i - \sum_{r=1}^{N} c_r^i \alpha_r^i \ . \tag{10}$$

- *Synchronized:* The player always flips all resources together. More specifically, the player uses only one single-resource strategy (i.e., process) for all of the resources, with a single flip rate α^i. The asymptotic benefit of a player i using the synchronized combination is

$$\beta^i = \gamma^i - \alpha^i \sum_{r=1}^{N} c_r^i \ . \tag{11}$$

Since the AND and OR control models are symmetric, we only compute the asymptotic gains in the AND model in this paper. Formulas for the asymptotic gains in the OR model can be derived from our results readily. Furthermore, since the defender's asymptotic gain γ^D can be computed from the attacker's asymptotic gain γ^A using the simple formula $\gamma^D = 1 - \gamma^A$, we only compute the asymptotic gain of the attacker.

The proofs of the formulas presented in this section can be found in Appendix A. Here, we first show the more general results for the $\mathcal{R} \cup \mathcal{P}$ strategy class (Table 2); then, we analyze the game for the $\mathcal{E}$ and $\mathcal{P}$ classes (Table 3 and Figure 3).

Table 2: Asymptotic Gain for Various Combinations of Single-Resource Strategies

Defender single-resource strategy	Defender combination	Attacker single-resource strategy	Attacker combination	Attacker's gain γ^A
$\mathcal{R} \cup \mathcal{P}$	independent	$\mathcal{R} \cup \mathcal{P}$	independent	$\prod_{r=1}^{N} \int_0^\infty f_{Z_r^D}(z_r) F_{Z_r^A}(z_r) dz_r$
			synchronized	$\int_0^\infty \prod_{r=1}^{N} \left(1 - F_{Z_r^D}(z)\right) f_{Z^A}(z) dz$
	synchronized			$\int_0^\infty f_{Z^D}(z) F_{Z^A}(z) dz$
			independent	$\int_0^\infty \prod_{r=1}^{N} F_{Z_r^A}(z) f_{Z^D}(z) dz$

Table 2 shows the attacker's asymptotic gain for various multi-resource strategies chosen by the defender and the attacker. The $\mathcal{R} \cup \mathcal{P}$ in the first and third column indicates that we assume the players use combinations of either non-arithmetic renewal ($\mathcal{R}$) or periodic ($\mathcal{P}$) single-resource strategies. The combinations used by the defender and the attacker are in the second and fourth columns, respectively. Finally, the attacker's gain γ^A for the given combinations is in the fifth column.

To express the attacker's gain, we use a notion similar to that of the basic `FlipIt` game, and let Z_r^i be the random variable representing the time elapsed since player i's last flip on resource r (Z^i if the player uses a synchronized strategy). We denote the cumulative distribution and density functions of Z_r^i by $F_{Z_r^i}(z)$ and $f_{Z_r^i}(z)$. These functions can easily be computed from the generating distribution of any non-arithmetic renewal strategy (see Appendix A).

It is noteworthy that, when both players use the synchronized combination, the game is equivalent to the basic `FlipIt` game (with $c^i = \sum_r c_r^i$): each player uses only one single-resource (i.e., basic `FlipIt`) strategy, and the state of all resources is the same as they are always flipped together. Consequently, the formula for the attacker's gain is identical to that of [13].

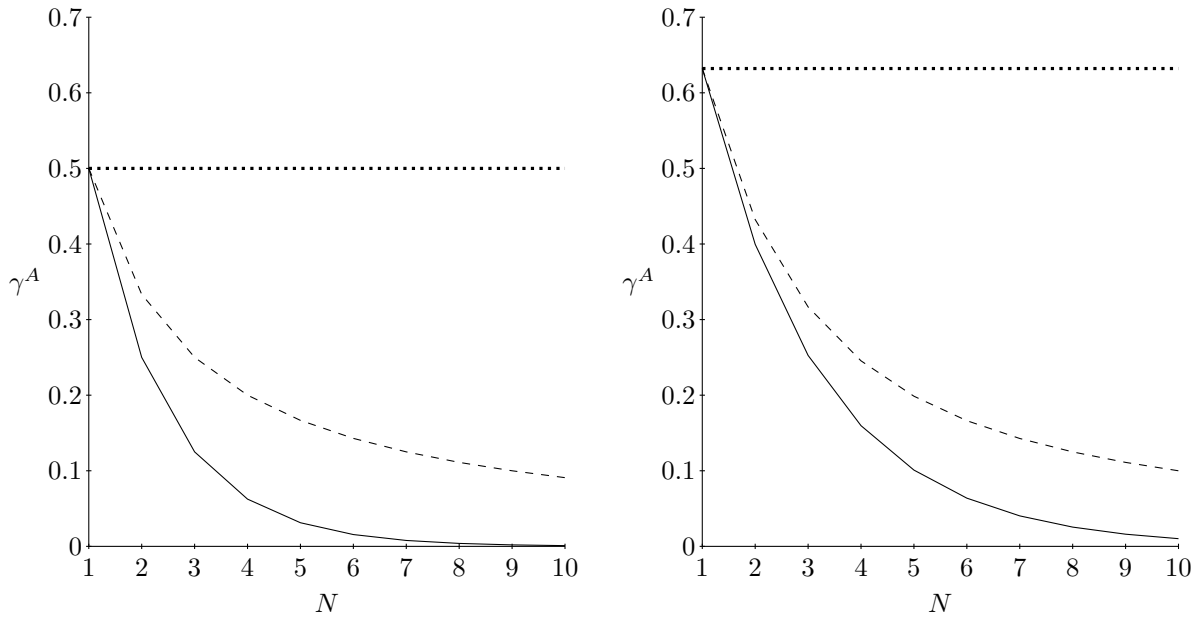
Table 3: Asymptotic Gain for Various Combinations of Exponential and Periodic Strategies

Defender single-resource strategy	Defender combination	Attacker single-resource strategy	Attacker combination	Attacker's gain γ^A
$\mathcal{E}$	independent	$\mathcal{E}$	independent	$\prod_{r=1}^{N} \frac{\alpha_r^A}{\alpha_r^A + \alpha_r^D}$
			synchronized	$\frac{\alpha^A}{\alpha^A + \sum_{r=1}^{N} \alpha_r^D}$
	synchronized			$\frac{\alpha^A}{\alpha^A + \alpha^D}$
	independent	$\mathcal{P}$	independent	$\prod_{r=1}^{N} \frac{\alpha_r^A}{\alpha_r^D} \left(1 - e^{-\frac{\alpha_r^D}{\alpha_r^A}}\right)$
			synchronized	$\frac{\alpha^A}{\sum_{r=1}^{N} \alpha_r^D} \left(1 - e^{-\frac{\sum_{r=1}^{N} \alpha_r^D}{\alpha^A}}\right)$
	synchronized			$\frac{\alpha^A}{\alpha^D} \left(1 - e^{-\frac{\alpha^D}{\alpha^A}}\right)$

Table 3 shows the attacker's asymptotic gain for various combinations of exponential and periodic strategies. We selected these single-resource strategies because they are known to be optimal in some respect (see Section 2). The table is similar to Table 2, except that the synchronized defender against independent attacker case is omitted to keep the table simple (it can be found in Appendix A) and because it is not a good strategy for either of the players.

The table shows that the independent combination is generally better than the synchronized one for the defender as her flip rates are added together in the former. This can be explained by the AND control model: since the defender only needs to control at least one resource, her best strategy is to flip one resource at a time. This forces the attacker to frequently flip all resources back as she cannot know which resources were flipped by the defender (since the exponential process is memoryless).

The formulas also suggest that the attacker should choose the synchronized combination over the independent one. When both players use exponential single-resource strategies, the attacker's gain decays exponentially as the number of resources increases ($\sim k^{-N}$) if she uses the independent combination, but only according to a power law ($\sim N^{-k}$) if she uses the synchronized one (given that flip rates stay the same). When the attacker uses the periodic single-resource strategy, the relationship between the number of resources and the attacker's gain is more complicated, but similar.



(a) Both players use exponential strategies.

(b) Defender uses exponential, attacker uses periodic strategy.

Figure 3: The attacker's asymptotic gain for various combinations of exponential and periodic strategies and varying number of resources. Plotted pairs of combination are: both players use independent (solid line), attacker uses synchronized while defender uses independent (dashed line), and both players use synchronized (dotted line). In this figure, the flip rates are assumed to be uniform, i.e., $\alpha^A = \alpha_r^A = \alpha^D = \alpha_r^D = 1,\ r = 1, \ldots, N$.

Figure 3 shows the attacker's asymptotic gain for various combinations of exponential and periodic strategies and varying number of resources. The plotted pairs of combinations are: both players use independent strategies (solid line —), attacker uses synchronized while defender uses independent strategy (dashed line - -), and both players use synchronized strategies (dotted line ····). The flip rates are assumed to be uniform, i.e., $\alpha^A = \alpha_r^A = \alpha^D = \alpha_r^D = 1,\ r = 1, \ldots, N$.

The figure shows that, for the given single-resource classes and parameters, the synchronized combination strongly dominates the independent one for the attacker. Again, this can be explained by the AND control model: since the attacker needs to control all resources, it makes to flip them at once. Otherwise, the probability that all resources become controlled by the attacker is very low. However, by using the synchronized combination, the attacker loses the freedom of choosing the flipping rate for each resource independently. Thus, when the heterogeneity of the attacker's flipping costs is very high, the independent combination may outperform the synchronized one.

The figure also supports our finding that the independent combination strongly dominates the synchronized one for the defender. Since the player has complete freedom in choosing her flip rates in the independent combination, this combination is better for the defender even for very heterogeneous flipping costs.

Finally, the figure supports that the periodic strategy dominates the exponential strategy as the attacker's gain is higher when she chooses the former.

5 The Markov Strategy Class

In the previous section, we studied how single-resource strategies can be combined into multi-resource strategies. However, such combinations represent only a tiny fraction of the actual multi-resource strategy space as there are an infinite number of multi-resource strategies that cannot be represented as such simple combinations. For example, a defender might choose to flip one resource periodically, then wait for a time interval chosen according to an exponential distribution, and then flip another resource. To model such complex multi-resource strategies, in this section, we introduce the *Markov* strategy class.

For the clarity of presentation, we derive results for two resources, yet the strategy is applicable for any number of resources. Furthermore, as opposed to the basic model, we are going to use a discrete model in this section. Note that the discrete model can be very realistic as players typically do not flip their resources at arbitrary times. Examples are the change of passwords, cryptographic keys or the application of software updates. We denote the duration of a time step by Δ. Finally, let us introduce the time-dependent age functions as follows. The random variable representing the number of time steps elapsed since the last flip against resource r at time step k is denoted by $Z_r^A(k)$ and $Z_r^D(k)$, respectively.

In discrete time, the attacker can perform one of the following actions in a given time slot:

- It does not flip any of the resources,
- it flips one of the resources,
- or, it flips both resources.

If the decision which action to choose depends only on the time elapsed since the previous flips against the resources, then $\{(Z_1^A(k), Z_2^A(k)), k = 0, 1, \dots\}$ defines a Markov process. The behavior of the attacker can be characterized by the following joint distributions corresponding to the events that can happen in two consecutive time steps.

$$p_{i,j}^{(0)} = \Pr\left[Z_1^A(0) = i, Z_2^A(0) = j, Z_1^A(1) = i+1, Z_2^A(1) = j+1\right] , \tag{12}$$

$$p_{i,j}^{(1)} = \Pr\left[Z_1^A(0) = i, Z_2^A(0) = j, Z_1^A(1) = 0, Z_2^A(1) = j+1\right] , \tag{13}$$

$$p_{i,j}^{(2)} = \Pr\left[Z_1^A(0) = i, Z_2^A(0) = j, Z_1^A(1) = i+1, Z_2^A(1) = 0\right] , \tag{14}$$

$$p_{i,j}^{(1,2)} = \Pr\left[Z_1^A(0) = i, Z_2^A(0) = j, Z_1^A(1) = 0, Z_2^A(1) = 0\right] , \tag{15}$$

where $p_{i,j}^{(0)}$ is the probability that no flip takes place in the next time step, $p_{i,j}^{(1)}$ ($p_{i,j}^{(2)}$) is the probability that there will be a flip against resource 1 or 2, while $p_{i,j}^{(1,2)}$ corresponds to the case when both resources will be flipped in the next time step.

We denote by M_p the Markov strategy generated by a Markov process with event probabilities $p = \{p_{i,j}^{(0)}, p_{i,j}^{(1)}, p_{i,j}^{(2)}, p_{i,j}^{(1,2)}$ for $i, j = 0, 1, \dots\}$, and by $\mathcal{M}$ the class of all Markov strategies. That is:

$$\mathcal{M} = \{M_p \mid p \text{ is the set of event probabilities}\} \ . \tag{16}$$

5.1 Linear Programming Solution

With these definitions and notations we can define a linear program to determine the optimal probabilities $p_{i,j}^{(\bullet)}$. However, since linear programming problems can only be solved with finite number of variables (in the general case), we have to restrict the game to a finite time horizon. The last time step we take into consideration is denoted by T.

The attacker wants to maximize her benefit β^A, which is composed of the gain probability and the cost of the flips against both resources as

$$\beta^A = \max_p \Bigg\{ \underbrace{\sum_{i=0}^{T}\sum_{j=0}^{T} q_{i,j} \Pr\left[Z_1^D > i, Z_2^D > j\right]}_{\gamma^A} - c_1^A \underbrace{\left(\sum_{i=0}^{T}\sum_{j=0}^{T} p_{i,j}^{(1)} + p_{i,j}^{(1,2)}\right)\frac{1}{\Delta}}_{\alpha_1^A} - c_2^A \underbrace{\left(\sum_{i=0}^{T}\sum_{j=0}^{T} p_{i,j}^{(2)} + p_{i,j}^{(1,2)}\right)\frac{1}{\Delta}}_{\alpha_2^A} \Bigg\} \ , \tag{17}$$

where $q_{i,j}$ is the probability that the age of the attack against resource 1 and 2 is i and j, respectively. This probability can be expressed easily as

$$q_{i,j} = p_{i,j}^{(0)} + p_{i,j}^{(1)} + p_{i,j}^{(2)} + p_{i,j}^{(1,2)} \ , \tag{18}$$

thus the subject function given by (17) defines a linear relation with regards to $p_{i,j}^{(\bullet)}$.

As variables $p_{i,j}^{(\bullet)}$ must be valid probabilities, we need to apply inequality constraints

$$p_{i,j}^{(0)} > 0, \quad p_{i,j}^{(1)} > 0, \quad p_{i,j}^{(2)} > 0, \quad p_{i,j}^{(12)} > 0 \ , \tag{19}$$

and we also need to ensure that the probabilities sum up to 1

$$\sum_{i=0}^{T}\sum_{j=0}^{T} p_{i,j}^{(0)} + p_{i,j}^{(1)} + p_{i,j}^{(2)} + p_{i,j}^{(1,2)} = 1 \ . \tag{20}$$

Further equality constraints are required to define the possible state transitions, yielding

$$q_{i,j} = p_{i-1,j-1}^{(0)}, \qquad \text{for } i > 0, j > 0, \tag{21}$$

$$q_{0,j} = \sum_{i=0}^{T} p_{i,j-1}^{(1)}, \qquad \text{for } j > 0, \tag{22}$$

$$q_{i,0} = \sum_{j=0}^{T} p_{i-1,j}^{(2)}, \qquad \text{for } i > 0, \tag{23}$$

$$q_{0,0} = \sum_{i=0}^{T}\sum_{j=0}^{T} p_{i,j}^{(1,2)} \ , \tag{24}$$

with $q_{i,j}$ given by (18).

Finally, we have to express with appropriate constraints that the age of resource r can not be T unless it is flipped in the next time step. We get

$$p_{i,j}^{(0)} = 0, \quad \text{for } i = T \text{ or } j = T, \tag{25}$$

$$p_{i,j}^{(1)} = 0, \quad \text{for } j = T, \tag{26}$$

$$p_{i,j}^{(2)} = 0, \quad \text{for } i = T. \tag{27}$$

5.2 Results

The linear program defined above helps to answer several questions regarding the `FlipThem` game. Such questions are:

- What is the optimal strategy of the attacker given the strategy of the defender?
- What are the optimal flip rates that maximize the benefit at the defender side given that the attacker always plays an optimal strategy?
- What is the Nash equilibrium of this game?

Solving the optimization problem using linear programming based approach poses some challenges. The length of the time horizon (T) is limited by the capabilities of the linear programming solution algorithm. In our examples, we used the built-in solver of MATLAB with $T = 30$, which means 900 variables if we have two resources.[2] Note that the number of variables increases polynomially in the length of the time horizon and exponentially in the number of resources. Specialized software can extend the analysis to optimize a large number of variables.

In the rest of the section we consider several numerical examples to demonstrate the usefulness of the model. In each of the examples the attacker is assumed to be Non-adaptive (NA, see Section 2), but she is also assumed to know the strategy of the defender (KS in Section 2). The defender, however, has no information about the attacker. For the definitions and rationale behind these modeling choices, see Section 2.

5.2.1 Optimal attack for a given defender strategy

In this example the defender flips the resources according to independent Poisson processes with parameters $\alpha_1^D = 1$ and $\alpha_2^D = 3$. The joint age function since flipping the resource is then

$$P(Z_1^D > i, Z_2^D > j) = e^{-\alpha_1^D i\Delta - \alpha_2^D j\Delta} \ . \tag{28}$$

The flip costs of the attacker are $c_1^A = 0.1$ and $c_2^A = 0.05$. The discrete problem is solved with $T = 30$ and $\Delta = 0.03$.

At this point we take the opportunity to introduce matrices $\boldsymbol{P^{(0)}}, \boldsymbol{P^{(1)}}, \boldsymbol{P^{(2)}}$ and $\boldsymbol{P^{(1,2)}}$ that help to visualize and to understand the strategy of the attacker. The entries of these matrices are

$$[\boldsymbol{P^{(0)}}]_{i,j} = \frac{p_{i,j}^{(0)}}{p_{i,j}^{(0)} + p_{i,j}^{(1)} + p_{i,j}^{(2)} + p_{i,j}^{(1,2)}} \ , \qquad [\boldsymbol{P^{(1)}}]_{i,j} = \frac{p_{i,j}^{(1)}}{p_{i,j}^{(0)} + p_{i,j}^{(1)} + p_{i,j}^{(2)} + p_{i,j}^{(1,2)}} \ , \tag{29}$$

$$[\boldsymbol{P^{(2)}}]_{i,j} = \frac{p_{i,j}^{(2)}}{p_{i,j}^{(0)} + p_{i,j}^{(1)} + p_{i,j}^{(2)} + p_{i,j}^{(1,2)}} \ , \qquad [\boldsymbol{P^{(1,2)}}]_{i,j} = \frac{p_{i,j}^{(2)}}{p_{i,j}^{(0)} + p_{i,j}^{(1)} + p_{i,j}^{(2)} + p_{i,j}^{(1,2)}} \ . \tag{30}$$

To simulate the attack, one has to follow the state of the attacker given by the i, j position in the matrix. In state (i, j), no flips occur with probability $[\boldsymbol{P^{(0)}}]_{i,j}$, and the next state of the attacker will be $(i+1, j+1)$. With probability $[\boldsymbol{P^{(1)}}]_{i,j}$ ($[\boldsymbol{P^{(2)}}]_{i,j}$) only resource 1 (resource 2) is flipped in the next time step, and the next state of the system is $(0, j+1)$ ($(i+1, 0)$), respectively. Finally, both resources are flipped in the next time step with probability $[\boldsymbol{P^{(1,2)}}]_{i,j}$, followed by a jump to state $(0, 0)$.

In this particular example all the non-zero entries of all four matrices are 1. The structure of the matrices is depicted in Figure 4, where black squares mean probability 1, and white squares mean probability 0 (matrix $\boldsymbol{P^{(1)}}$ is not depicted as it has only zero entries). Let us follow the strategy of the attacker starting from the initial state, which is $(0, 0)$ (bottom left corner of the plots in the figure). The black square located at $(0, 0)$

[2] This models, for example, the key update policy of a company over a duration of 2.5 years assuming that updates are defined by the granularity of a month.

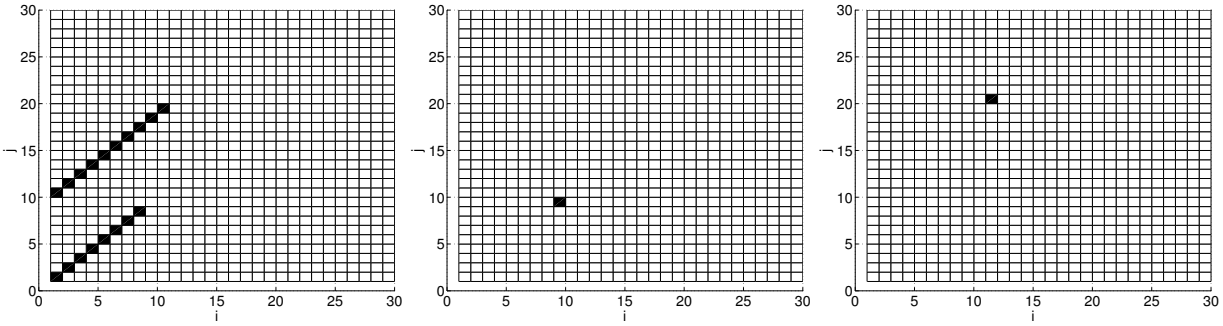


Figure 4: Optimal attack strategy against two resources flipped according to independent Poisson processes

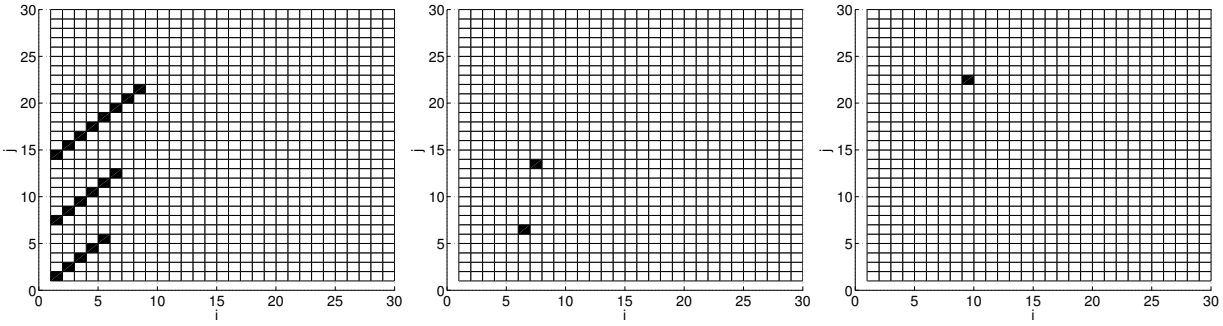


Figure 5: Optimal attack strategy against two resources flipped according to independent periodic processes

in matrix $\boldsymbol{P^{(0)}}$ means that no flip occurs in the current time instant with probability 1, and that the next state will be $(1,1)$. The black square in $(1,1)$ means that there will be no flip at the next time step either. The first flip occurs at time step 9, since matrix $\boldsymbol{P^{(2)}}$ has a probability 1 at position $(9,9)$. After the flip the new state of the attacker will be $(0,10)$. The state of the attacker increases to $(1,11),(2,12)$, etc., with no flips occurring. Finally, at the 20th time step, when the state of the attacker is $(20,11)$, she flips both resources (as $[\boldsymbol{P^{(2)}}]_{20,11}=1$) and returns to state $(0,0)$.

Thus, based on the matrices a periodic attack can be identified with a period of $\delta=20$. The resources are not flipped in a synchronized manner. Resource 2 is flipped at the 9th time step from the beginning of the period, while both resources are flipped at the end of the period.

If the defender flips both resources according to independent periodic strategies, the joint age process is given by

$$P(Z_1^D>i,Z_2^D>j)=\begin{cases}(1-\alpha_1^D i\Delta)(1-\alpha_2^D j\Delta), & \text{if } i\Delta<1/\alpha_1^D, j\Delta<1/\alpha_2^D\\ 0, & \text{otherwise.}\end{cases} \tag{31}$$

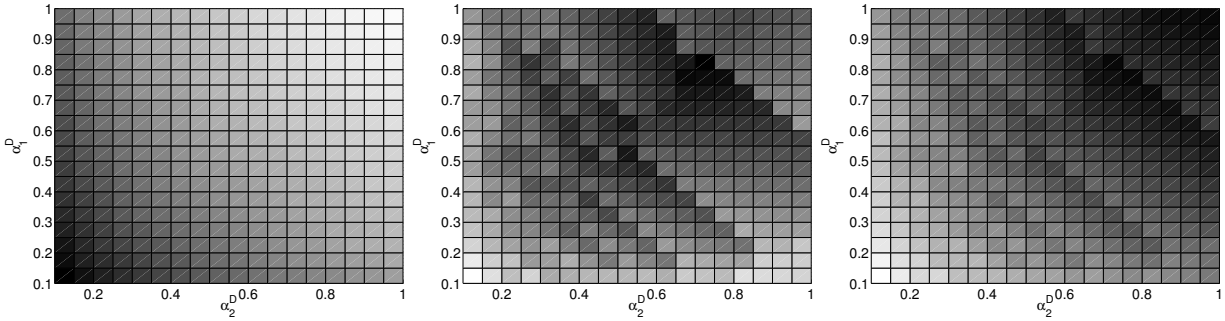
When keeping all parameters the same as before, the optimal strategy of the attacker is more complex in this case (see Figure 5). The period of her strategy is $\delta=22$ now. It flips solely resource 2 at time step 6 and at time step 13, while it flips both resources at time step 22, which also marks the end of the period.

It is worth noting that the benefit of the attacker is 0.265 in the Poisson, and it is 0.047 in the periodic case, meaning that periodic defense is less economical to attack (given, of course, that the attacker has no knowledge on the last move of the defender, thus it is of type NA).

5.2.2 Determining the optimal flip rates of the defender

The linear program can also be used to determine what the best flip rates of the defender are given that the attacker applies the optimal strategy. (Notice that we do not calculate the Nash equilibrium in this section, thus the defender does not take the strategy of the attacker into consideration).

Let us first consider the case when the defender flips his resources according to independent Poisson processes. The question is what the best choice for the flipping rates is. Assume the flipping costs of the attacker are $c_1^A = 0.1$ and $c_2^A = 0.2$. We solved the linear program with various combinations of α_1^D and α_2^D, and with two different settings for c_1^D and c_2^D parameters. The benefit of the attacker and the defender has been recorded in each case. The results are shown in Figure 6. As the benefit of the attacker is the subject of optimization in the linear program, the corresponding plot is obviously smooth, and gives higher values for lower flip rates of the defender. The corresponding gain rates, however, are not smooth. As the benefit of the defender is in direct relation with the gain rates of the attacker, the plots of the benefit of the defender are not smooth either. (Note that similar plots in Figure 5. in [13] are not smooth either.) The best benefit for the defender is 0.222 obtained at $\alpha_1^D = 0.8, \alpha_2^D = 0.7$.



β^A β^D, with $c_1^D = 0.1, c_2^D = 0.2$ β^D, with $c_1^D = 0.05, c_2^D = 0.1$

Figure 6: Benefits of the attacker and the defender by various flip rates of the defender (Poisson case). Darker shades of gray indicates higher benefit.

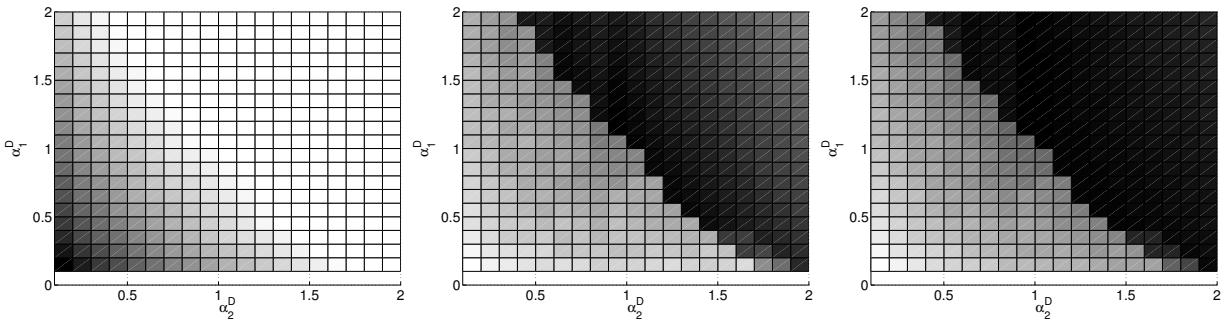
If the defender flips the resources according to independent periodic processes, higher flip rates are required to maximize the benefits. The corresponding results are depicted in Figure 7. The optimal flip rates are $\alpha_1^D = 0.9, \alpha_2^D = 1.2$, but also the benefit is higher when compared to the Poisson case, $\beta^D = 0.61595$. Observe that the attacker drops out in several cases, as indicated by the white area on the plot of her benefit and also by the sharp line appearing on the plots of the benefits of the defender.

5.2.3 Calculating the Nash equilibrium

The proposed linear program can be applied to calculate the optimal strategies of both the defender and the attacker. We can thus obtain a simple iterative algorithm to determine the Nash equilibrium of the game. This algorithm starts with assigning a random strategy to the defender, followed by the alternating optimization of the attacker and the defender strategies. In practice, however, we found that this algorithm does not converge in the vast majority of the cases, but it starts oscillating after a given number of iterations, suggesting that no Nash equilibrium exists.

6 Discussion and Recommendations

Extending the `FlipIt` game for multiple resources requires to model the goals of the players as functions of the compromised resources. We selected the two most intuitive choices, namely the AND and OR control models, to represent the gains derived from controlling the resources. From the attacker's viewpoint, the AND control model represents the case when all resources need to be compromised to get access to the



β^A β^D, with $c_1^D = 0.1, c_2^D = 0.2$ β^D, with $c_1^D = 0.05, c_2^D = 0.1$

Figure 7: Benefits of the attacker and the defender by various flip rates of the defender (periodic case). Darker shades of gray indicates higher benefit.

desired system. This is similar to the *total effort* model of security interdependence in the state-of-the-art [8, 14]. The OR control model represents the case when the compromise of a single resource suffices to get access. This second choice relates to the *weakest link* model of security interdependence.

We considered two major classes of multi-resource strategies: combinations of single-resource strategies (independent processes and synchronized processes) and the Markov class of strategies. Based on our result, we can formulate a set of recommendations for the defender. These recommendations can readily be used in practice where the assumptions of the `FlipIt` game apply, for example, when defining the key update strategy for a security infrastructure.

- For the AND control model, we found that the defender should use independent flipping strategies. In practice, this means that cryptograhic keys should not be updated at the same time, but rather independently.
- On the other hand, for the OR control model, the defender should use synchronous flipping strategies. In practice, this means to update cryptographic keys synchronously. However, the defender needs to pay attention to the cost of updating keys in the OR control model. If these costs are very heterogeneous, the key update processes remain synchronized, but with different update rates across the keys.
- If the attacker is non-adaptive, then the periodic defender strategy is a good choice according to the numerical results.[3] Periodic strategies have multiple advantageous properties such as higher benefits for the defender, robustness to optimization errors and ease of implementation in practice. However, periodic strategies perform poorly against an LM attacker [13]. Thus, the defender carefully needs to assess the potential information available to the attacker when choosing her strategy.
- Surprisingly, the defender's benefit is not a smooth or monotonous function of her flip rates, which makes optimization in practice much more difficult. Numerical results imply that this observation holds for any combination of the periodic and the exponential strategy classes. The major reason behind this non-monotonous property is that, as the defender's flip rate reaches a threshold, the attacker drops out of the game. In realistic cases, the defender's flipping cost is much lower than the attacker's flipping cost, which causes the attacker to drop out.

7 Future Work

In this paper, we present the `FlipThem` game, the generalization of the `FlipIt` game for multiple resources. There are several avenues to further develop this model.

[3] This complies with the results of the basic `FlipIt` game for a single resource in [13].

First, we solved the case for a non-adaptive attacker modeling a situation with limited defender knowledge and limited attacker options for feedback (such as highly-protected computer systems). There exist cases when the attacker can potentially obtain (a limited) feedback from the targeted system and can use this knowledge to implement a feedback strategy (LM). In the original `FlipIt` paper [13], the authors introduce a greedy strategy as a good response for an LM attacker against a defender employing a renewal strategy. We can generalize this greedy single-resource strategy in the `FlipThem` game for multiple resources.

In this paper, we extensively study the AND and OR control models and derive results based on them. There exist cases when the attacker neither has to compromise all resources nor does she reach her goal by compromising only a single resource. In general, the attacker can have various options to compromise a subset of the available resources to reach her goal. For example, in order to compromise trust within a group of ten nodes, an attacker might have to compromise any subset consisting of at least five nodes.

Targeted attacks typically involve the compromise of multiple resources where the attack of a resource is based on the successful compromise of another resource. An attacker typically has to realize such an attack path to reach her goal. Attack trees were proposed to represent the collection of possible attack paths for the attacker. Clearly, these attack paths represent subsets of the available resources and hence the methodology of playing `FlipThem` for a subset of resources can be applied to this special case.

References

[1] Boldizsár Bencsáth, Gábor Pék, Levente Buttyán, and Mark Felegyhazi. The Cousins of Stuxnet: Duqu, Flame, and Gauss. *Future Internet*, 4(4):971–1003, 2012.

[2] Kevin Bowers, Marten van Dijk, Robert Griffin, Ari Juels, Alina Oprea, Ronald Rivest, and Nikos Triandopoulos. Defending against the unknown enemy: Applying FlipIt to system security. In *Proc. of the 3rd Conference on the Decision and Game Theory for Security (GameSec)*, pages 248–263, 2012.

[3] Peter Bright. Defense contractor: aggressive action kept cyberattackers at bay. Ars Technica, May 2011.

[4] cnet.com. Comodo hack may reshape browser security. cnet.com, Apr 4 2011.

[5] Jim Finkle and Andrea Shalal-Esa. Hackers breached U.S. defense contractors. Reuters, May 27 2011.

[6] Fox-IT. Black Tulip: Report of the investigation into the DigiNotar Certificate Authority breach. Fox-IT, Aug 13 2012.

[7] Kaspersky Lab. "Red October" diplomatic cyber attacks investigation. Kaspersky Lab, Jan 14 2013.

[8] Aron Laszka, Mark Felegyhazi, and Levente Buttyán. A survey of interdependent security games. Technical Report CRYSYS-TR-2012-11-15, CrySyS Lab, Budapest University of Technology and Economics, Nov 2012.

[9] Mandiant. APT1: Exposing One of China's Cyber Espionage Units. Mandiant, Feb 18 2013.

[10] Joseph Menn. Key Internet operator VeriSign hit by hackers. Reuters, feb 2 2012.

[11] Viet Pham and Carlos Cid. Are we compromised? Modelling security assessment games. In *Proc. of the 3rd Conference on the Decision and Game Theory for Security (GameSec)*, pages 234–247, 2012.

[12] Uri Rivner. Anatomy of an attack. RSA, Apr 2011.

[13] Marten van Dijk, Ari Juels, Alina Oprea, and Ronald L. Rivest. FlipIt: The game of "Stealthy Takeover". Cryptology ePrint Archive, Report 2012/103, 2012.

[14] Hal Varian. System reliability and free riding. *Economics of Information Security*, pages 1–15, 2004.

A Asymptotic Gain of the Attacker for Various Combinations of Single-Resource Strategies

To compute the attacker's asymptotic gain γ^A, we rely on that it is equal to the probability of the attacker being in control at a random time instant. In the AND model, we can write this as

$$\gamma^A = Pr\left[C^A = 1\right] = Pr\left[Z_1^D > Z_1^A \ \wedge \ \ldots \ \wedge \ Z_N^D > Z_N^A\right] \ . \tag{32}$$

We also use the following two results from [13].

For a non-arithmetic renewal process given by a cumulative distribution function F, the age density $f_{Z(t)}$ and cumulative distribution $F_{Z(t)}$ functions converge as

$$\lim_{t\to\infty} f_{Z(t)}(z) = \frac{1-F(z)}{\mu} \tag{33}$$

and

$$\lim_{t\to\infty} F_{Z(t)}(z) = \frac{\int_0^z (1-F(x))dx}{\mu} \ , \tag{34}$$

where μ is the expected value of the distribution generating the renewal process.

For a periodic strategy with random phase, the age density and cumulative distribution functions are

$$f_{Z(t)}(z) = \begin{cases} \alpha, & z < \frac{1}{\alpha} \\ 0, & z \geq \frac{1}{\alpha} \end{cases} \tag{35}$$

and

$$F_{Z(t)}(z) = \begin{cases} \alpha z, & z < \frac{1}{\alpha} \\ 1, & z \geq \frac{1}{\alpha} \end{cases} \ . \tag{36}$$

Since we also want to find the attacker's gain for the exponential strategy, we have to compute the age density and cumulative distribution functions for the exponential strategy. The cumulative distribution function and the mean of the exponential distribution are $1-e^{\lambda x}$ and λ^{-1}; thus, we have that

$$\lim_{t\to\infty} f_{Z(t)}(z) = \frac{1-(1-e^{-\lambda z})}{\lambda^{-1}} = \lambda e^{-\lambda z} = \alpha e^{-\alpha z} \ . \tag{37}$$

and

$$\lim_{t\to\infty} F_{Z(t)}(z) = \lambda \int_0^z e^{-\lambda x} dx = 1 - e^{-\lambda z} = 1 - e^{-\alpha z} \ . \tag{38}$$

A.1 Both Players Use Non-arithmetic Renewal Strategies

A.1.1 Both Players Use Independent Strategies

$$\begin{aligned}
\gamma^A &= \Pr\left[Z_1^D > Z_1^A \ \wedge \ \ldots \ \wedge \ Z_N^D > Z_N^A\right] && (39)\\
&= \Pr\left[Z_1^D > Z_1^A\right] \cdot \ldots \cdot \Pr\left[Z_N^D > Z_N^A\right] && (40)\\
&= \int_0^\infty f_{Z_1^D}(z_1) F_{Z_1^A}(z_1) dz_1 \cdot \ldots \cdot \int_0^\infty f_{Z_N^D}(z_N) F_{Z_N^A}(z_N) dz_N && (41)\\
&= \prod_{r=1}^{N} \int_0^\infty f_{Z_r^D}(z_r) F_{Z_r^A}(z_r) dz_r \ . && (42)
\end{aligned}$$

For exponential strategies,

$$\gamma^A = \prod_{r=1}^{N} \int_0^\infty \alpha_r^D e^{-\alpha_r^D z_r} \left(1 - e^{-\alpha_r^A z_r}\right) dz_r \tag{43}$$

$$= \prod_{r=1}^{N} \left(\alpha_r^D \int_0^\infty e^{-\alpha_r^D z} dz - \alpha_r^D \int_0^\infty e^{-(\alpha_r^D + \alpha_r^A) z}) dz\right) \tag{44}$$

$$= \prod_{r=1}^{N} \left(1 - \alpha_r^D \frac{1}{\alpha_r^A + \alpha_r^D}\right) \tag{45}$$

$$= \prod_{r=1}^{N} \frac{\alpha_r^A}{\alpha_r^A + \alpha_r^D} \ . \tag{46}$$

A.1.2 Synchronized Attack

$$\gamma^A = \int_0^\infty \Pr\left[z < Z_1^D \ \wedge \ \ldots \ \wedge \ z < Z_N^D\right] f_{Z^A}(z) dz \tag{47}$$

$$= \int_0^\infty \Pr\left[z < Z_1^D\right] \cdot \ldots \cdot \Pr\left[z < Z_N^D\right] f_{Z^A}(z) dz \tag{48}$$

$$= \int_0^\infty \prod_{r=1}^{N} \Pr\left[z < Z_r^D\right] f_{Z^A}(z) dz \tag{49}$$

$$= \int_0^\infty \prod_{r=1}^{N} \left(1 - F_{Z_r^D}(z)\right) f_{Z^A}(z) dz \tag{50}$$

$$\tag{51}$$

For exponential strategies,

$$\gamma^A = \int_0^\infty \prod_{r=1}^{N} \left(1 - (1 - e^{-\alpha_r^D z})\right) \alpha^A e^{-\alpha^A z} dz \tag{52}$$

$$= \alpha^A \int_0^\infty \prod_{r=1}^{N} \left(e^{-\alpha_r^D z}\right) e^{-\alpha^A z} dz \tag{53}$$

$$= \alpha^A \int_0^\infty e^{-z\left(\alpha^A + \sum_{r=1}^{N} \alpha_r^D\right)} dz \tag{54}$$

$$= \frac{\alpha^A}{\alpha^A + \sum_{r=1}^{N} \alpha_r^D} \ . \tag{55}$$

A.1.3 Synchronized Defense

$$\gamma^A = \int_0^\infty \Pr\left[z > Z_1^A \ \wedge \ \ldots \ \wedge \ z > Z_N^A\right] f_{Z^D}(z)dz \tag{56}$$

$$= \int_0^\infty \Pr\left[z > Z_1^A\right] \cdot \ldots \cdot \Pr\left[z > Z_N^A\right] f_{Z^D}(z)dz \tag{57}$$

$$= \int_0^\infty \prod_{r=1}^{N} \Pr\left[z > Z_r^A\right] f_{Z^D}(z)dz \tag{58}$$

$$= \int_0^\infty \prod_{r=1}^{N} F_{Z_r^A}(z) f_{Z^D}(z)dz \ . \tag{59}$$

For exponential strategies,

$$\gamma^A = \int_0^\infty \prod_{r=1}^{N} \left(1 - e^{-\alpha_r^A z}\right) \alpha^D e^{-\alpha^D z} dz \ . \tag{60}$$

A.1.4 Both Players Use Synchronized Strategies

For this strategy profile, the game is equivalent to the basic `FlipIt` game. Consequently, we already have from [13] that

$$\gamma^A = \int_0^\infty f_{Z^D}(z) F_{Z^A}(z)dz \ . \tag{61}$$

For exponential strategies,

$$\gamma^A = \int_0^\infty \alpha^D e^{-\alpha^D z}(1 - e^{-\alpha^A z})dz \tag{62}$$

$$= \alpha^D \int_0^\infty e^{-\alpha^D z}dz - \alpha^D \int_0^\infty e^{-(\alpha^D + \alpha^A)z})dz \tag{63}$$

$$= 1 - \alpha^D \frac{1}{\alpha^A + \alpha^D} \tag{64}$$

$$= \frac{\alpha^A}{\alpha^A + \alpha^D} \ . \tag{65}$$

A.2 Defender Uses Non-arithmetic Renewal, Attacker Uses Periodic Strategy

A.2.1 Both Players Use Independent Strategies

$$\gamma^A = \int_0^\infty \cdots \int_0^\infty \Pr\left[z_1 < Z_1^D \wedge \ldots \wedge z_N < Z_N^D\right] f_{Z^A}(z_1, \ldots, z_N) \ dz_1 \ldots dz_N \tag{66}$$

$$= \int_0^\infty \cdots \int_0^\infty \Pr\left[z_1 < Z_1^D\right] \cdot \ldots \cdot \Pr\left[z_N < Z_N^D\right] f_{Z_1^A}(z_1) \cdot \ldots \cdot f_{Z_N^A}(z_N) \ dz_1 \ldots dz_N \tag{67}$$

$$= \int_0^\infty \cdots \int_0^\infty \prod_{r=1}^{N} \Pr\left[z_r < Z_r^D\right] \prod_{r=1}^{N} f_{Z_r^A}(z_r) \ dz_1 \ldots dz_N \tag{68}$$

$$= \int_0^{\frac{1}{\alpha_1^A}} \cdots \int_0^{\frac{1}{\alpha_N^A}} \prod_{r=1}^{N} \left(1 - F_{Z_r^D}(z_r)\right) \prod_{r=1}^{N} \alpha_r^A \ dz_1 \ldots dz_N \ . \tag{69}$$

For exponential defender strategy,

$$\gamma^A = \int_0^{\frac{1}{\alpha_1^A}} \cdots \int_0^{\frac{1}{\alpha_N^A}} \prod_{r=1}^{N} e^{-\alpha_r^D z_r} \prod_{r=1}^{N} \alpha_r^A \; dz_1 \cdots dz_N \tag{70}$$

$$= \int_0^{\frac{1}{\alpha_1^A}} \cdots \int_0^{\frac{1}{\alpha_{N-1}^A}} \alpha_N^A \int_0^{\frac{1}{\alpha_N^A}} e^{-\alpha_N^D z_N} dz_N \prod_{r=1}^{N-1} e^{-\alpha_r^D z_r} \prod_{r=1}^{N-1} \alpha_r^A \; dz_1 \cdots dz_{N-1} \tag{71}$$

$$= \int_0^{\frac{1}{\alpha_1^A}} \cdots \int_0^{\frac{1}{\alpha_{N-1}^A}} \frac{\alpha_N^A}{\alpha_N^D} \left(1 - e^{\frac{\alpha_N^D}{\alpha_N^A}}\right) \prod_{r=1}^{N-1} e^{-\alpha_r^D z_r} \prod_{r=1}^{N-1} \alpha_r^A \; dz_1 \cdots dz_{N-1} \tag{72}$$

$$= \frac{\alpha_N^A}{\alpha_N^D} \left(1 - e^{\frac{\alpha_N^D}{\alpha_N^A}}\right) \int_0^{\frac{1}{\alpha_1^A}} \cdots \int_0^{\frac{1}{\alpha_{N-1}^A}} \prod_{r=1}^{N-1} e^{-\alpha_r^D z_r} \prod_{r=1}^{N-1} \alpha_r^A \; dz_1 \cdots dz_{N-1} = \ldots \tag{73}$$

$$= \prod_{r=1}^{N} \frac{\alpha_r^A}{\alpha_r^D} \left(1 - e^{-\frac{\alpha_r^D}{\alpha_r^A}}\right) \; . \tag{74}$$

A.2.2 Synchronized Attack

$$\gamma^A = \int_0^{\infty} \Pr\left[z < Z_1^D \; \wedge \; \ldots \; \wedge \; z < Z_N^D\right] f_{Z^A}(z) dz \tag{75}$$

$$= \int_0^{\infty} \Pr\left[z < Z_1^D\right] \cdot \ldots \cdot \Pr\left[z < Z_N^D\right] f_{Z^A}(z) dz \tag{76}$$

$$= \int_0^{\infty} \prod_{r=1}^{N} \Pr\left[z < Z_r^D\right] f_{Z^A}(z) dz \tag{77}$$

$$= \int_0^{\frac{1}{\alpha^A}} \prod_{r=1}^{N} \left(1 - F_{Z_r^D}(z)\right) \alpha^A dz \; . \tag{78}$$

For exponential defender strategy,

$$\gamma^A = \int_0^{\frac{1}{\alpha^A}} \prod_{r=1}^{N} e^{-\alpha_r^D z} \alpha^A dz \tag{79}$$

$$= \alpha^A \int_0^{\frac{1}{\alpha^A}} e^{-z \sum_r \alpha_r^D} dz \tag{80}$$

$$= \frac{\alpha^A}{\sum_{r=1}^{N} \alpha_r^D} \left(1 - e^{-\frac{\sum_{r=1}^{N} \alpha_r^D}{\alpha^A}}\right) \; . \tag{81}$$

A.2.3 Synchronized Defense

$$\gamma^A = \int_0^\infty \Pr\left[z > Z_1^A \ \wedge \ \ldots \ \wedge \ z > Z_N^A\right] f_{Z^D}(z)dz \tag{82}$$

$$= \int_0^\infty \Pr\left[z > Z_1^A\right] \cdot \ldots \cdot \Pr\left[z > Z_N^A\right] f_{Z^D}(z)dz \tag{83}$$

$$= \int_0^\infty \prod_{r=1}^{N} \Pr\left[z > Z_r^A\right] f_{Z^D}(z)dz \tag{84}$$

$$= \int_0^{\min_r \frac{1}{\alpha_r^A}} \prod_{r=1}^{N} \left(\alpha_r^A z\right) f_{Z^D}(z)dz \tag{85}$$

$$= \left(\prod_{r=1}^{N} \alpha_r^A\right) \int_0^{\min_r \frac{1}{\alpha_r^A}} z^N f_{Z^D}(z)dz \ . \tag{86}$$

For exponential defender strategy,

$$\gamma^A = \left(\prod_{r=1}^{N} \alpha_r^A\right) \int_0^{\min_r \frac{1}{\alpha_r^A}} z^N \alpha^D e^{-\alpha^D z} dz \ . \tag{87}$$

A.2.4 Both Players Use Synchronized Strategies

For this strategy profile, the game is equivalent to the basic `FlipIt` game. Thus,

$$\gamma^A = 1 - \int_0^\infty F_{Z^D}(z) f_{Z^A}(z)dz \tag{88}$$

$$= 1 - \int_0^{\frac{1}{\alpha^A}} F_{Z^D}(z) \frac{1}{\alpha^A} dz \ . \tag{89}$$

For exponential defender strategy,

$$\gamma^A = 1 - \int_0^{\frac{1}{\alpha^A}} (1 - e^{-\alpha^D z}) \frac{1}{\alpha^A} dz \tag{90}$$

$$= 1 - \left(\alpha^A \frac{1}{\alpha^A} + \alpha^A \frac{e^{-\frac{\alpha^D}{\alpha^A}} - 1}{\alpha^D}\right) \tag{91}$$

$$= \frac{\alpha^A}{\alpha^D} \left(1 - e^{-\frac{\alpha^D}{\alpha^A}}\right) \ . \tag{92}$$