KU LEUVEN

FACULTEIT INGENIEURSWETENSCHAPPEN

# Gametheory and Cybersecurity: a study FlipIt and multiple resources

Sophie Marien

Thesis voorgedragen tot het behalen
van de graad van Master of Science
in de ingenieurswetenschappen:
computerwetenschappen,
hoofdspecialisatie Veilige software

**Promotor:**
Prof. dr. ir. Tom Holvoet

**Assessoren:**
Ir. W. Eetveel
W. Eetrest

**Begeleider:**
Ir. Jonathan Merlevede, Ir. Kristof
Coninx

Academiejaar 2014 – 2015

# Voorwoord

I would like to thank everybody who kept me busy the last year, especially my promotor and my assistants. I would also like to thank the jury for reading the text. My sincere gratitude also goes to my wive and the rest of my family.

*Sophie Marien*

# Inhoudsopgave

# Todo list

# Samenvatting

There are many possible ways to attack a company network. Everyday they suffer frrom multiple attakcs and stealthy attacks. We will make use of a gamemodel FlipIt to find out what the best strategies are for a network manager to defend his network. A worm or a virus will propagate through the network and will cause nodes to be infected. By flipping it the network manager can keep his network clean. In this thesis I present a work of gametheory merged with cybersecurity. The `abstract` environment contains a more extensive overview of the work. But it should be limited to one page.

# Samenvatting

In dit `abstract` environment wordt een al dan niet uitgebreide Nederlandse samenvatting van het werk gegeven. Wanneer de tekst voor een Nederlandstalige master in het Engels wordt geschreven, wordt hier normaal een uitgebreide samenvatting verwacht, bijvoorbeeld een tiental bladzijden.

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetuer id, vulputate a, magna. Donec vehicula augue eu neque. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Mauris ut leo. Cras viverra metus rhoncus sem. Nulla et lectus vestibulum urna fringilla ultrices. Phasellus eu tellus sit amet tortor gravida placerat. Integer sapien est, iaculis in, pretium quis, viverra ac, nunc. Praesent eget sem vel leo ultrices bibendum. Aenean faucibus. Morbi dolor nulla, malesuada eu, pulvinar at, mollis ac, nulla. Curabitur auctor semper nulla. Donec varius orci eget risus. Duis nibh mi, congue eu, accumsan eleifend, sagittis quis, diam. Duis eget orci sit amet orci dignissim rutrum.

# Lijst van figuren en tabellen

## Lijst van figuren

## Lijst van tabellen

# List of Abbreviations and Symbols

## Abbreviations

| | |
|---|---|
| LoG | Laplacian-of-Gaussian |
| MSE | Mean Square error |
| PSNR | Peak Signal-to-Noise ratio |

# Hoofdstuk 1

# Introduction

## 1.1 Introduction

In this era where digitalization becomes prominent in every aspect of our lives, where technology is growing fast and where business are always under attack, security becomes an issue of increasing complexity. Without security, their is no protection to keep somebody out of a system. It is the same as leaving the door of your house open for everybody to come in. It is important to keep a system secure. A hacker will be a person that seeks exploits or weaknesses in a system or network, to be able to gain access. Many of those attacks have a different cause. Some of the attacks by a hacker can be benign, others can be harmful. Their are variant ways to break into a system. Virusses, worms, spyware and other malware are nr 2 for the top external threats [security report kaspersky 2014]. Furthermore these kind of threat also causes the greatest percentage in loss of data. These threats will infect the network by means of a virus that will propagate through the network.

waarom? : lekken van informatie, DOSS attacks, ..

It is difficult to protect a system or a network against APT. Game theory is more and more used to model Cyber Security. Game theory analyses in this case a game where the interactions are between an attacker and a defender of a system, where both players have to make decisions. Both players want to .. Researchers at RSA made a game theoretic framework to model targeted attacks. Specific for a scenario where a system or network is taken over completely by an attacker repeatedly and this attack is not immediately detected by the defender of the system or network. This game theoretic game "flipitïs a two players game where the attacker and the defender are competing to get control over a shared resource. Both players do not know who is currently in control of the resource until they move. In FlipIt every move gives them immediately control over the resource. But what if the attacker moves and it takes a while before the attacker gets full control over the resource? FlipIt does not take into account that a move may not be instantaneous, but has a certain delay. Take for example as a resource a network with different nodes ( laptops, datacenters). The defender drops a virus on one of the nodes and wait till this virus infects the whole network. The defender will only be in control of the resource if the whole network is infected. This an lead us to the following resaurch questions: Gaat er een specieke grootte zijn van een delay waarbij de attacker al weet dat hem niet meer moet gaan spelen ? ( is niet gelijk aan de grootte van de periode van de attacker) When working with a network and a delay we can .. graph model en uitleggen hoe we de graph kunnen maken zodat de delay altijd zo groot mogelijk gaat zijn.

In this paper we want to focus on situations where a computer network is attacked by APT.

We propose an addition to the basic FlipIt model to model a scenario where the moves by the attacker will not be instantaneous. Next we analyse what the new Nash equilibria will be and ..

# Hoofdstuk 2

# The FlipIt game

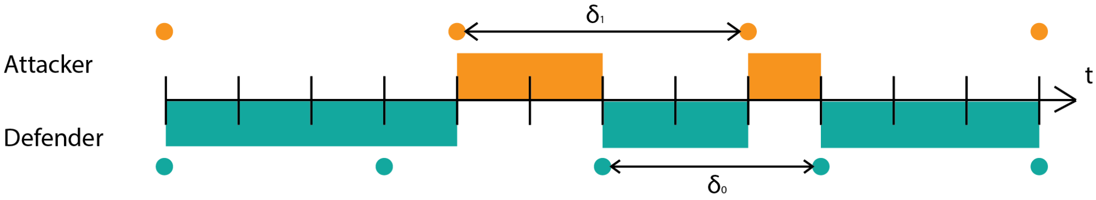
In this chapter, we introduce the game FlipIt [**?** ]. FlipIt is a game introduced by van Dijk et al. First we explain the framework of FlipIt and it's important results. In the next section the formulas and assumptions are made that will be used throughout the paper. To understand how to model a FlipIt game with virus propagation it is important to get familiar with the concepts of the normal FlipIt game and it's notations.

## 2.1 The FlipIt game

FlipIt is a two-players game with a shared (single) resource that the players want to control as long as possible. The shared resource can be a password, a network or a secret key depending on the setting being modelled. In the rest of the paper we will call the two players the attacker, denoted by the subscript $A$ and the defender, denoted by subscript $D$.

The game begins at $t = 0$ and continuous indefinitely ($t \to \infty$). The time in the game can be viewed as being continuous, but a discrete time can also be viewed. To get control over the resource, the players $i$ can flip the resource at any given time. A flip will be regarded as a move from a player $i$. Each move will imply a certain cost $k_i$ and the cost can vary for each player. Both players will try to minimize their cost. By adding a cost, it will prevent players to move to frequently.

The unique feature of FlipIt is that every move will happen in a stealthy way, meaning that the player has no clue that (his adversary) the other player has flipped the resource. For instance, the defender will not find out if the resource has already been compromised by the attacker, but he can only potentially know it after he flips the resource himself. The goal of the player is to maximize the time that he or she has control over the resource while minimizing total cost of the moves. A move can also result in a "wasted move", called a flop. It may happen that the resource was already under control by the defender. If the defender moves when he or she has already control over the resource, he or she would have wasted a move since it does not result in a change of ownership and a cost is involved.



Figuur 2.1: A representation of a FlipIt game where both players are playing periodically and discrete. Every move or "flipïs indicated by a blue or orange circle. The blue and orange rectangles represent the amount of time one of the players is in control of the resource.

verwijzen naar de figuur 2.1

The state of the resource is denoted as a time independent variable $C = C_i(t)$. $C_D(t)$ is either 1 if the game is under control by the defender and 0 if the game is under control by the attacker. For $C^A(t)$ it is visa versa, $C^A(t) = 1 - C^D(t)$.

The game starts with defender being in control of the game, $C_D(0) = 1$.

The players receive a benefit equal to the time of units that they were in possession of the resource minus the cost of making their moves. The cost of a player $i$ is denoted by $k_i$. The total gain of player $i$ is equal to the total amount of time that a player $i$ has owned the resource from the beginning of the game up to time $t$. It is expressed as follows:

$$G_i(t) = integraal[0][t]C_i(x)dx. \tag{2.1}$$

The average gain of player $i$ is defined as:

$$\gamma_i(t) = G_i(t)/t. \tag{2.2}$$

Let $\beta_i(t)$ denote player's $i$ average benefit upto time $t$:

$$\beta_i(t) = \gamma_i(t) - k_i\alpha_i. \tag{2.3}$$

This is equal to the fraction of time the resource has been owned by player $i$, minus the cost of making the moves. $\alpha_i$ defines the average move rate by player $i$ up to time $t$.

Because the players move in a stealthy way, there are different types of feedback that a player can get while moving:

- Non-adaptive (NA): The player does not receive any feedback during the game while flipping.
- Last move (LM): When a player flips it will find out the exact time that the opponent played the last time.

| **Categories** | **Classes of Strategies** |
|---|---|
| Non-adaptive (NA) | Exponential |
| | Periodic |
| | Renewal |
| | General non-adaptive |
| Adaptive (AD) | Last move (LM) |
| | Full History (FH) |

Tabel 2.1: Classes of strategies in FlipIt

- Full History (FH): When a player flips it will find out the whole history of the opponents move.

The game can be extended by the amount of information that a player receives. It can also be possible for a player to get information at the start of the game. Both interesting cases are:

- Rate-of-play (RP): The player finds out the exact rate of play of the opponent.
- Knowledge-of-strategy (KS): The player finds out the complete information of the strategy that the opponent is playing.

In our analyses of the FlipIt game with a virus propagation in section [], we assume the strategy of both players to be non-adaptive. None of the players has information of the strategy of the opponent. The defender will never know when the attacker will attack the network with a virus. Conversely, the attacker does not know how often the defender defends his network.

### 2.1.1 Strategies

In this subsection we elaborate about the strategy of FlipIt . For the other strategies of FlipIt we refer the reader to [the paper of FlipIt]

There are two different kinds of strategies, the *non-adaptive strategies* and the *renewal strategies*. If there is no need for feedback for both of the players, we say that we have a non-adaptive strategy. Because the player does not receive any feedback during the game it will play in the same manner against every opponent. They are not dependent on the opponents movements. This means that they can already generate the time sequence for all the moves in advance. But they can depend on some randomness because the non-adaptive strategies can be randomised. In this paper we will focus in the beginning on the non-adaptive strategies. Reasons behind this is that a player (defender or attacker) rarely knows what the strategies are of his opponent. [If the attacker wants to move stealthily, it might have limited attack options FLIPTHEM].

nog redenen zoeken

A renewal strategy is a non-adaptive strategy where the time intervals between two consecutive moves are generated by a renewal process.

Periodic

Non-Arithmetic Renewal

Exponential

# Hoofdstuk 3

# Related work

## 3.1 Related work

Difference with FlipThem: sub part of nodes for control and strategy difference: dependant of grade of the nodes, instead of just periodic. - Literatuurstudie - Flipit - Game Motivation - Formal definition

## 3.2 Extensions on FlipIt

In this section we discuss the extentions that can be made on the FlipIt game.

There a various possible ways to extend FlipIt. Laszka et al. made a lot of additions and extensions on the original game of FlipIt. For instance Laszka et al. extended the basic FlipIt game to multiple resources. The incentive is that for compromising a system in a real case it needs more than just taking over just one resource. An example is that one resource can be gaining access to a system and breaking the password of the system is another resource. The model is called FlipThem [**?** ]. They use two ways to flip the multiple resources: the AND and the OR control model. In the AND model the attacker only controls the system if he controls all the resources of the system, whereas in the OR model the attacker only needs to compromise one resource to be in control of the entire system.
Another addition of Laszka et al. to the game of FlipIt [mitigating covert compromises] is extending the game to also consider non-targeted attacks by non-strategic players. In this game the defender tries to maintain control over the resource that is subjected to both targeted and non-targeted attacks. Non-targeted attacks can include phishing, while targeted attacks may include threats delivered through zero day attack vulnerabilities.
One of the last important addition from Laszka et al. [Mitigation of Targeted and] is to consider a game where the moves made by the attacker are still covert but the moves made by the defender are known to the attacker. This means that the attacker can base his attacks on the defender's moves. Both the targeted and non-targeted attacks don't succeed immidiatly. For the targeted attack the time till it succeeds

misschien meer uitleggen

is given by an exponential distributed random variable with a known rate. The non-targeted attacks are modelled as a single attacker and the time till it succeeds is given by a Poisson process.. The conclusion of this paper is that the optimal strategy for the defender is moving periodically.

citatie needed voor Are We Compromised?

Other authors used the FlipIt game to apply it on a specific scenario. To be able to use the FlipIt game, modifications where required for the FlipIt model. One of the scenarios by Pham[**?** ] [] was to find out whether a resource was compromised or not by the attacker. This could be verified by the defender, who has an extra move "test"beside the flip move. The basic idea is to test with an extra action if the resource has been compromised or not. This move involves also an extra cost. This model is useful if somebody wants to know for example if his or her password has been compromised.

Finally researchers also have investigated the behavioural of humans when playing FlipIt. A Nochenson and Grossklags [A behavioural investigation of the FlipIt game] investigate how people really act when given temporal decisions. [Risk-seeking in a continuous game of timing] Reitter et al. they observe continuous games, 20-seconds FlipIt game..

Distributed Worm Simulation with a Realistic Internet 2005
Modelling of congestions of network through worm propagation. Mathematical model focussing on the underlying network infrastructure.(diff no game theory)

Of threats and Costs: A Game-theoretic approach to security risk management 2013
Model network security of networks with a non-cooperative node through game theory. Attacker knows the defence strategies and the defender has knowledge of the possible attacks. Each actor considers the actions of the other before deciding to strive to optimize their own utility. (diff not stealthy)

Game theory meet network security and privacy (2013)
Chapter 3 addresses several games in game theory for modelling network security.

Game theoretic approach for cost-benefit analysis of malware proliferation prevention
(..) Introduces SIS and SIR together with 'patch', 'removal' and 'patch and removal'.

### 3.2.1 What can be done in further research

- Looking for the dynamics of the spread of the virus/worm limited by the bandwidth of the network links, BPG routing failure with high volume scan traffic

## 3.3 Conclusion

## 3.4 Why Game Theory to model security problem

Actors in a security protocol must follow the systems and some arbitrarily actors that are malicious and do not follow the protocol. [Bridging Game Theory and Cryptography]. Game theoretic approach proposes a model where all the actors act with self-interest.

## 3.5 ..

Flip-it. Some authors have written other papers about flipit. One of them is the [Game theoretic approach for cost-benefit analysis of malware proliferation prevention].

# Hoofdstuk 4

# APT

## 4.1 Advanced Persistent Threats

A targeted attack follows most of the time a serie of stages to attack its victim. This pattern of stages is also know as the Kill Chain, first mentioned by .. []. An APT will not always follow exact each step of this chain but it will give a good guideline of how an APT works.

1. **Reconnaissance**: During the first step of the Kill Chain an attacker will look for information to find an interesting victim. This information can be emailaddresses, IP addresses, conference information, anything that is available about the victim.

2. **Weaponization**: In the second stage the attacker will use an exploit and add a malicious playload to be send to the victim.

3. **Delivery**: The attacker will deliver his malicious code to the victim through different kins of intrusion methods. This can include email, usb stick, cd's, web, applications or other means.

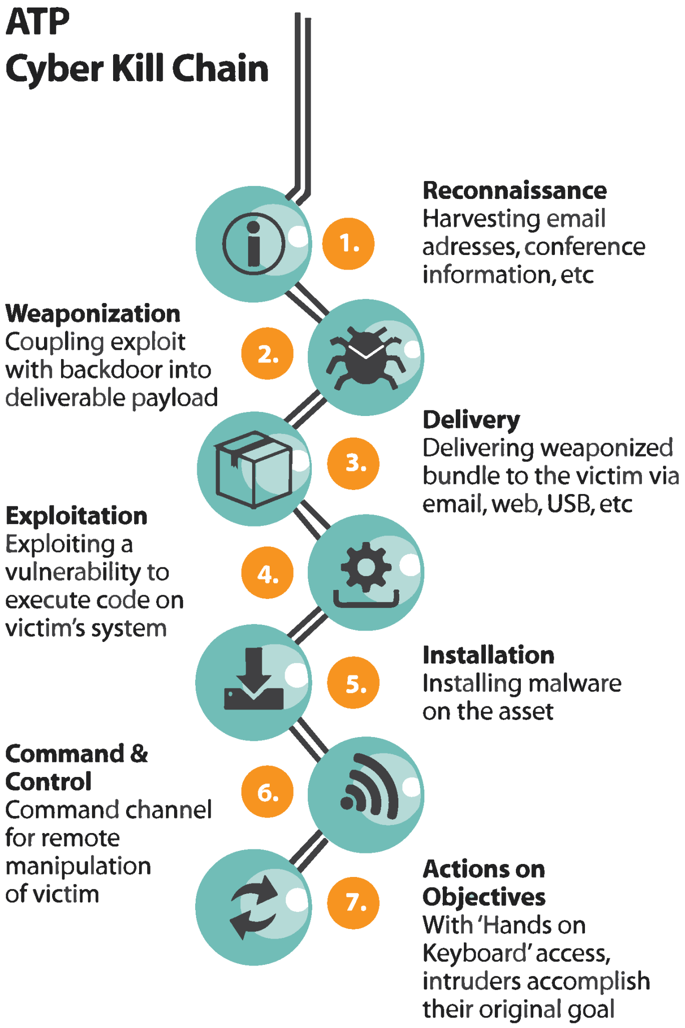
4. **Exploitation**:The attacker executes the exploit, which is only relevant if the attacker uses an exploit.

5. **Installation**: The malware will be installed on the asset. This is only relevant if the attacker uses malware as a part of the attack.

6. **Command and Control**: The attacker will set up a command and control channel for remote manipulation of the victim.

7. **Actions on Objectives**: With "hands on keyboard" access, intruders accomplish their original goal.

nog uitbreiden, toevoegen that attackers will stay unnoticed for as long as possible or leave unnoticed with sensitive information


ATP
Cyber Kill Chain
1.
Reconnaissance
Harvesting email adresses, conference information, etc
2.
Weaponization
Coupling exploit with backdoor into deliverable payload
3.
Delivery
Delivering weaponized bundle to the victim via email, web, USB, etc
4.
Exploitation
Exploiting a vulnerability to execute code on victim's system
5.
Installation
Installing malware on the asset
6.
Command & Control
Command channel for remote manipulation of victim
7.
Actions on Objectives
With 'Hands on Keyboard' access, intruders accomplish their original goal

# Hoofdstuk 5

# FlipIt with virus propagation

## 5.1 FlipIt vs FlipIt with virus propagation

This chapter explains how to model a FlipIt game with a virus that propagates and infects the nodes in a network. First, this section explains the difference between FlipIt with and without a virus. Then, section 5.2 derives the formula to calculate the gain for a FlipIt game without a virus. After that section 5.3 introduces a modification to this formula to achieve an adapted gain formula for a FlipIt game with virus propagation. This allows us to derive the benefit formula.

In chapter 2 the FlipIt game was explained. This chapter starts from the specific case of a non-adaptive continuous FlipIt game where both players play a periodic strategy with a random phase. This choice is motivated by the assumption that in the practical situation of most organisations, the defence strategy is to periodically defend the network. This corresponds to a periodic defender strategy. To simplify the analysis in a first time, a periodic attacker strategy is assumed as well. Further research can investigate the effect of relaxing this assumption.

check reff

A FlipIt game consists of a single resource. To represent the security problem, the game now defines its single resource as a computer network with multiple nodes. One of the players, the defender, will try to defend his network. The defender will do this by flipping all the nodes of the network (i.e. the entire resource) in every move he plays. The attacker, the other player, will try to infect all the nodes in the network. The attacker will do this by flipping the node in the graph that can infect all the nodes in the shortest time possible. After dropping a virus on the first node, it takes a while for the virus to infect the entire network. However, since the original FlipIt game works with a single resource that is always flipped entirely, the assumption is made that the attacker is considered to have gained the control over the resource only when all the nodes of the network have been infected, i.e. the entire resource has been flipped.

## 5.2 Gain formula for a FlipIt game without virus propagation

In their FlipIt paper [], Marten van Dijk et al. , give a definition of the gain of a player $i$ . This definition is however, in the best of our knowledge, not easy to adapt to the situation with virus propagation. Therefore, this section presents an alternative formula that defines a game by quantifying the amount of time each player has control.

The following notations will be used throughout the formal definition (see figure 5.1 for a graphic representation of some of the notations):

$\delta_D$: This is the period of the defender. This denotes the length of the interval between two consecutive moves of the defender ($\delta_D > 0$).

$\delta_A$: This is the period of the attacker. This denotes the length of the interval between two consecutive moves of the attacker ($\delta_A > 0$).

$T_D$: This denotes the phase of the defender that was chosen randomly and uniformly over the interval $[0,\delta_D]$.

$T_A$: This denotes the phase of the attacker that was chosen randomly and uniformly over the interval $[0,\delta_A]$.

*Unit of control*: Defined as the period between gaining (full) control and losing control over the resource.

$n_D$: The n'th interval of the defender, starting from interval 0.

$n_A$: The n'th interval of the attacker, starting from interval 0.

$\Delta Unit_D(n)$ : This is a function that denotes the length of a unit of control of the defender in the n'th interval of the attacker or the defender depending on who is playing faster.

$\Delta Unit_A(n)$ : This is a function that denotes the length of a unit of control of the attacker in the n'th interval of the attacker or the defender depending on who is playing faster.

*lcm(a,b)*: The least common multiple of $a$ and $b$.

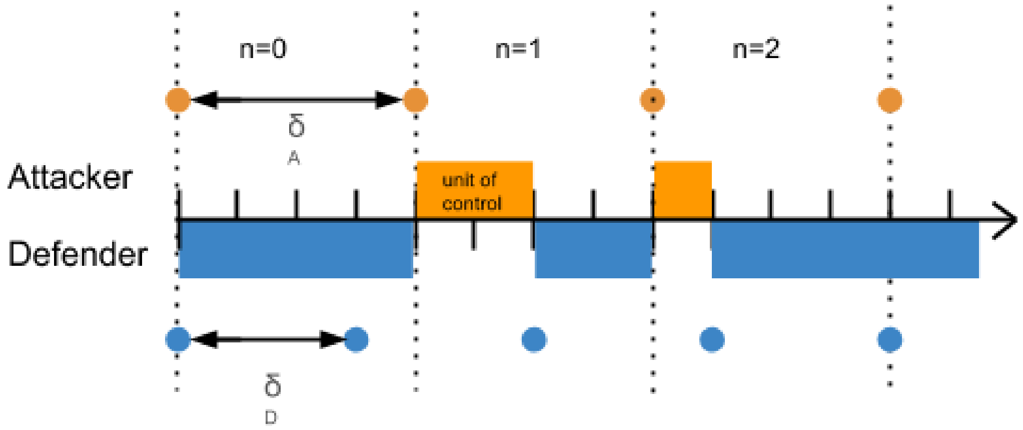
*gcd(a,b)*: The greatest common divider of $a$ and $b$.

$G_i(t)$: Gain of player $i$ at time t. The gain of a player is defined as the total amount of time that a player has owned the resource since the start of the game up to time t. In the context of a FlipIt game with virus propagation, the whole network is seen as one resource. This resource is owned by the attacker if he has control over the entire network.

*Average Gain*: $\gamma_i(t) = G_i(t)/t$ is the average gain rate of player $i$ which is defined as the fraction of time that player $i$ has control over the resource up to time t.

*Benefit*: The average benefit of a player $i$ is denoted by $\beta_i(t) = \gamma_i(t) - k_i/\delta_i$, which is equal to the fraction of time that the resource has been owned by player $i$, minus the cost rate for moving. For now we consider the cost rate equal to 0. In the rest of the paper the 'benefit' of the game for player $i$ will be used as shorthand for 'average benefit'.

FIGUUR 5.1: Graphic representation of some of the notations



To compute the gain of a player in a periodic game without phases, two cases are considered: case 1 where the defender moves at least as fast as the attacker and case 2 where the attacker plays at least as fast as the defender. Next, the formula is enriched by introducing the phases.

### Computing the gain for an attacker of a normal FlipIt game

Consider a game without phases, so in which both players start with a phase $T_D$ and phase $T_A$ equal to zero. Both players start their first move at $t = 0$. As previously stated (in the formal definition of the game and the introduction of different notations used throughout the paper), the defender has control in the beginning of the game at $t = 0$. For the remainder of the game, if the two players move at the same time during the game, the moves cancel each other out and no change of state happens.

**Case 1:**
$\delta_A \geq \delta_D$ (The defender moves at least as fast as the attacker.)

To compute the gain formula for the attacker,the amount of time that the attacker has control over the resource from the start of the game up to time $t$ has to be calculated. This can be done by computing the sum of all the units of control of the attacker up to time $t$.

To calculate a single unit of control of the attacker, the time line of the FlipIt game is divided into intervals of size $\delta_A$. Every time the attacker moves we have the start of a new interval, with the attacker being in control, unless there is a simultaneous move with the defender. Considering that the defender moves at least as fast as the attacker, he or she will at least move one time during the interval of the attacker. Because the attacker only moves at the start of his or her interval we can say that the defender will always end as being in control of the resource at the end of an attacker's interval. .

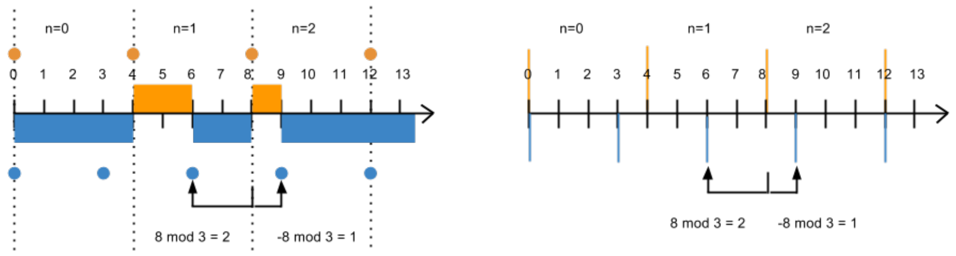
tekening met voorbeeld

To calculate how long the unit of control of the attacker is in the $n$'th interval, we only need to know how long the attacker has control over the resource before the defender moves in that interval. The start time of the $n$'th interval will be a multiple of the period of the attacker. Once the attacker has played, the time he can stay in control until the next move of the defender, depends on the time elapsed since the last time the defender played in the previous interval ( the ($n$-1)th interval).

The time at which the defender plays in an interval is a multiple of its period. Once the defender has played for the last time in the ($n$-1)'th interval, the remaining time until the attacker will play can be calculated as $n \cdot \delta_a$ *modulo* $\delta_D$ which is equal to the remainder of $n \cdot \delta_A$ divided by $\delta_D$. [referentie naar matworks: $http : //nl.mathworks.com/help/matlab/ref/mod.html$ ]The time the attacker will stay in control is then $\delta_D - n \cdot \delta_A$ *modulo* $\delta_D$, which can also be calculated as $(-n \cdot \delta_A)$ *modulo* $\delta_D$.

Figure 5.2 illustrates this graphically, for $\delta_A = 4$, $\delta_D = 3$ and the $n = 2$ interval. We see that in interval 1, the defender will stay 8 *modulo* $3 = 2$ in control, and so, in interval 2, the attacker will stay $1 = 3 - 2 = -(2 * 4)$ *modulo* 3 in control.

Figuur 5.2: Taking the modulo of a negative number



This brings us to the next formula to calculate the length of a unit of control in the $n$'th interval of the attacker.

For every positive and non zero real $\delta_A$ and $\delta_D \in \mathbb{R}$ and every $n \in \mathbb{N}$ (including 0 in the set of natural numbers) :

$$\Delta Unit_A(n_A) = [(-n_A) \cdot \delta_A] mod \delta_D \tag{5.1}$$

where $n_A$ is the number of the $n$'th interval of the attacker starting from interval 0 where the length of the unit of control of the attacker is calculated.

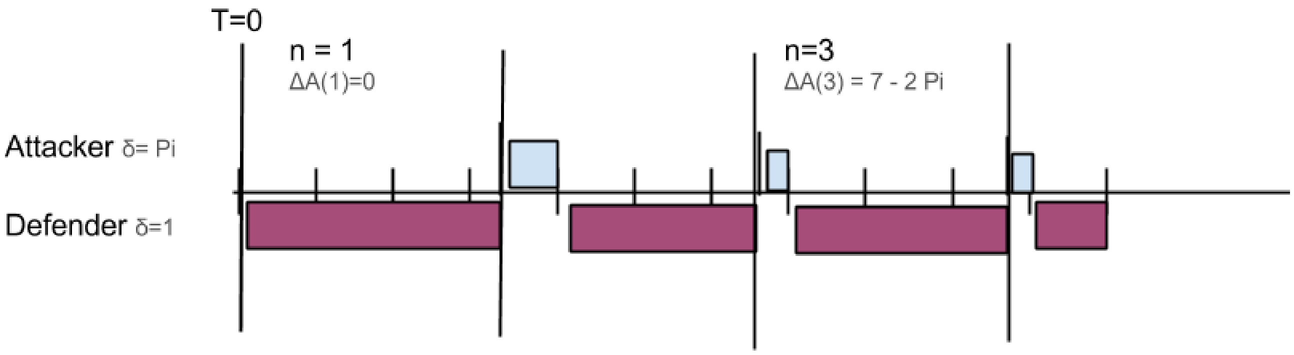
The length of a unit of control of the defender is the remainder of the interval after the attacker loses control over the resource when the defender plays. This can be defines as follows for the $n$'th interval of the attacker:

$$\Delta Unit_D(n_A) = \delta_A - [(-n_A) \cdot \delta_A] mod \delta_D \tag{5.2}$$

An example: Figure 5.2 shows a FlipIt game were the period of the attacker is $\pi$ and the period of the defender is 1. ... On figure 5.3

aanvullen

FIGUUR 5.3: Example for calculating the control unit in interval 3 for a FlipIt game with period defender = 1 and period attacker = pi



The gain formula can be calculated by taking all the units of control of the player up to an amount of $p$ intervals of the attacker. The gain formula for the attacker is stated as follows:

$$Gain_A = \sum_{i=0}^{p} \{[(-i) \cdot \delta_A] mod \delta_D\}\} \tag{5.3}$$

where $p$ is the number of units of control that have to be summed.

The gain of the defender is the sum of the units of control of the defender up to the same amount of $p$ intervals of the attacker:

$$Gain_D = \sum_{i=0}^{p} \{\delta_A - [(-i) \cdot \delta_A] mod \delta_D\}\} \tag{5.4}$$

$$Gain_D = \sum_{i=0}^{p}\{\delta_A \cdot i\} - \sum_{i=0}^{p}\{[(-i) \cdot \delta_A] mod \delta_D\}\} \tag{5.5}$$

$$Gain_D = \delta_A \cdot p - \sum_{i=0}^{p}\{[(-i) \cdot \delta_A] mod \delta_D\}\} \tag{5.6}$$

Note: The gain formula is not in function of time $t$ but the amount of $p$ intervals of the attacker. This approach is chosen because it will result in whole units of control. It is possible to make a gain formula using the time, but this will result in a much more complicated function.

For phases ..

**Case 2:**
$\delta_D \geq \delta_A$ (The attacker moves at least as fast as the defender.)

For this case we use the same approach as in case 1 but with a small difference. To compute the unit of control of both players we divide the time line of the FlipIt game into intervals of size $\delta_D$. The defender moves at the start of each interval, the end of the interval is the beginning of the next interval. Considering that the attacker will move at least as fast as the defender, he or she will move at least one time during the interval of the defender. Because the defender only moves in the beginning of each interval, the attacker will end as being in control of the resource.

If the unit of control of the defender need to be calculated, we only need to know how long it takes for the attacker to move in the interval. This can be done in the same way as in case 1 by taking the modulo of the negative of the beginning of the interval. The big difference with case 1 is when the length of the unit of control in the 0'th interval is calculated. Because the defender always has control in the beginning of the game, the first interval is computed in a different way. The unit of control of the defender in the 0'th interval is equal to the length of the period of the attacker, since from that moment the attacker takes control.

For every positive and non zero real $\delta_A$ and $\delta_D \in \mathbb{R}$ and every n $\in \mathbb{N}$ (including 0 in the set of natural numbers) :

*for* $n_A = 0$

$$\Delta Unit_D(n_D) = \delta_A \tag{5.7}$$

*for* $n_A > 0$

$$\Delta Unit_D(n_D) = [(-n_D) \cdot \delta_D] mod \delta_A \tag{5.8}$$

where $n_D$ is the number of the $n$'th interval of the defender starting form interval 0 where the length of the unit of control of the defender is calculated.

The length of a unit of control of the attacker is the remainder of the interval after the defender loses control over the resource when the attacker plays. This can be defined as follows for the n'th interval of the defender:

$$\Delta Unit_A(n_D) = \delta_D - \Delta Unit_D(n_D) \tag{5.9}$$

The gain formula for $\delta_D \geq \delta_A$ is calculated in the same manner as case 1:

The gain of the defender is the sum of the units of control of the defender up to the same amount of p intervals of the attacker:

$$Gain_D = \sum_{i=1}^{p} \{[(-i) \cdot \delta_D] mod \delta_A\} \quad with \quad i = 0 \quad Gain_D = \delta_A \tag{5.10}$$

and for the attacker:

$$Gain_A = \delta_D \cdot p - \sum_{i=1}^{p} \{[(-i) \cdot \delta_D] mod \delta_A\} \quad with \quad i = 0 \quad Gain_A = \delta_D - \delta_A \tag{5.11}$$

## 5.3 Gain and benefit formula for a FlipIt game with virus propagation

The formulas from the previous section can now be adapted to calculate the gain of both players in a FlipIt game with virus propagation. As mentioned before, the attacker will try to infect all the nodes in the network. He will do this by flipping the node in the graph that can infect all the nodes in the shortest time possible. After dropping a virus on the first node, it takes a while for the virus to infect the entire network. The time that it takes for the virus to infect every node will be denoted as parameter *d*. If we want to measure how long it takes for the virus to infect all the nodes in the network, we have to calculate the shortest path from the first infected node to the farthest node. This can be measured by a method explained in section [matrix berekeningen]. Assume that an attacker attacks at time *t*, then only at time $t + d$ he gains control over the entire network. If the defender flips the network before the period d has elapsed (so, somewhere between $t$ and $t + d$), then the attacker will never gain control over the entire network. Using this parameter *d*, a FlipIt game with virus propagation can be modelled.

The previous section defined a formula to calculate each unit of control of the attacker and the defender for two cases. If the virus propagation takes *d* time before every resource is infected then this *d* has to be subtracted from each unit of control. (see figure 7.4 5.4). It may happen that the unit of control is less than *d*. In that case, the result of the substration will be a negative number, meaning that the defender has flipped all the resources before the attacker could gain control over all the resources. To calculate the gain only the units of control bigger than 0 have to be summed. So the formula becomes:

For $\delta_A \geq \delta_D$ :

$$Gain_A = \sum_{i=0}^{p} \{[(-i) \cdot \delta_A] mod \delta_D - d\} > 0\} \tag{5.12}$$

where $p$ is the number of units of control that have to be summed.

The gain of the defender is equal to the amount of time that the attacker is not in control of the resource. So the formula for the defender becomes:

$$Gain_D = p \cdot \delta_A - \sum_{i=0}^{p} \{[(-i) \cdot \delta_A] mod \delta_D - d\} > 0 \tag{5.13}$$

For $\delta_D \geq \delta_A$ :

$$Gain_A = \delta_D \cdot p - \sum_{i=1}^{p} \{[(-i) \cdot \delta_D] mod \delta_A - d\} > 0 \tag{5.14a}$$

$$with \quad i = 0 \quad Gain_A = \delta_D - (\delta_A - d) > 0 \tag{5.14b}$$

$$\tag{5.14c}$$

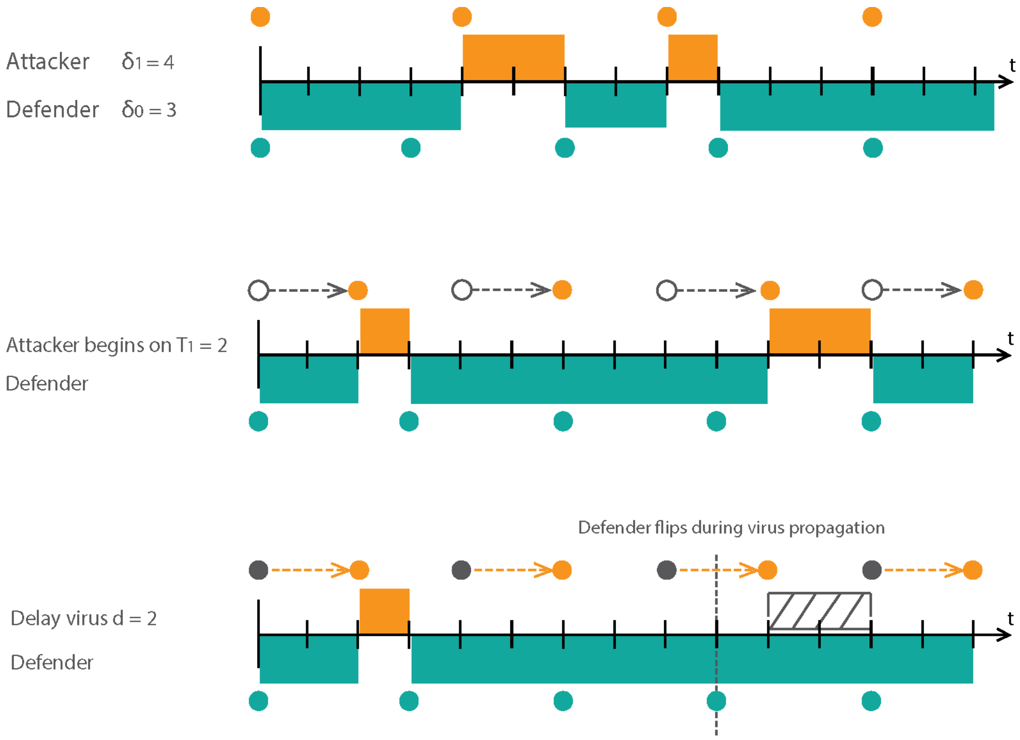
and for the defender:

$$Gain_D = \sum_{i=1}^{p} \{[(-i) \cdot \delta_D] mod \delta_A - d\} > 0 \tag{5.15a}$$

$$with \quad i = 0 \quad Gain_D = \delta_A - d > 0 \tag{5.15b}$$

$$\tag{5.15c}$$

Figuur 5.4: Difference in a FlipIt game between delay caused by a virus and a phase bigger than zero for the Attacker



### Computing the benefit of a FlipIt game with virus propagation

Calculating the benefit of both players, requires calculating the average gain rate of both players. To compute the benefit the value of parameter $p$ needs to be determined. Two cases can be considered: one case where $\delta_D$ and $\delta_A \in \mathbb{Q}$ and the other one where $\delta_D$ and $\delta_A \in \mathbb{I}$. In both cases we first calculate the benefit of the attacker in case that the defender moves at least as fast as the attacker. The benefit of the defender will be BenD = 1 - BenA. The benefit of both players for the case where the defender moves as least as fast as the defender is done in a similar way.

**Rational numbers ($\mathbb{Q}$):** When $\delta_D$ and $\delta_A$ are rational numbers, after a number of intervals (namely their least common multiple), the same pattern of intervals will be repeated over and over again. Why? A rational number is a number that can be expressed as the fraction p/q with p and q $\in \mathbb{Z}$ (integers), with the denominator q not equal to zero, it is possible to find the *lcm* of $\delta_D$ and $\delta_A$. The *lcm* is defined for all rational numbers as: $lcm(\frac{a}{b}, \frac{c}{d}) = \frac{lcm(a,c)}{gcd(b,d)} with$ []. When $t$ is equal to the *lcm* of $\delta_D$ and $\delta_A$, both players will move again at the same time and this can be mapped to the beginning of the game. Because we stated that at the end of the interval of the attacker, the defender is in control and because if two players move at the same time the moves cancel each other out, we can map this to the beginning of the game. Since the game goes on infinitely, to calculate the average gain of the attacker, it is sufficient to calculate the average gain of the attacker only during a period of time equal to the *lcm* of $\delta_D$ and $\delta_A$. Since *lcm* is a multiple of $\delta_D$ and $\delta_A$, there is a number $p$ so that *lcm* =p $\cdot\delta_A$, meaning that the attacker will have played $p$ times. $p$ can be defined as follows:

referentie

$$p = \frac{lcm(\delta_D, \delta_A)}{\delta_A} \tag{5.16}$$

This results in the following formula for the benefit of the attacker with a cost rate equal to zero:

$$\beta_A = \frac{\sum_{i=0}^{p}[[(-i) \cdot \delta_A]mod\delta_D - d] > 0}{lcm(\delta_D, \delta_A)} \tag{5.17}$$

As stated before [], the benefit of the attacker and the benefit of the defender add up to 1 ($\beta_A + \beta_B = 1$). The benefit of the defender can be writen as follows:

$$\beta_D = 1 - \beta_A \tag{5.18}$$

**Irrational numbers ($\mathbb{I}$):** If $\delta_D$ and/or $\delta_A \in \mathbb{I}$: An irrational number $i \neq \frac{a}{b}$ with $b \in \mathbb{Z}$ , a $\in \mathbb{N}$.

Two cases can be distinguished. (A) $\frac{\delta_D}{\delta_A}$ is a rational number a/b with a <= b. In that case, after b intervals, the pattern will repeat itself.

(B)If either $\delta_D$ and $\delta_A$ cannot be written as a fraction, and they are no multiple of each other, the least common multiplier cannot be calculated. Moreover, there will be no repeating pattern. If both players move at one point in the game at the same moment, this point of time has to be a multiple of the period of the attacker and a multiple of the period of the defender. But because there is no least common multiple, no such point of time exists during the game. If both players never play at the same moment, it is not possible to have a repeated pattern because no mapping to the beginning of the game can occur. Additionly two unit of controls with the same length cannot exist. This would mean that the game has a repeated pattern, which is not possible.

The game will go on forever, if no repeating pattern occurs and it would keep on generating units of control with different lengths. This implies that if the game goes on forever, every length between 0 and the smallest interval (which is $\delta_D$ ) will be generated. To calculate the benefit we want to summarize the unit of controls up to a number of interval $p$. Considering that the game goes on forever without repetition we cannot rely on the fact that the benefit can also be calculated only during the repetition. Calculating the benefit of a game without repetition would imply that all the unit of control to infinite have to be calculated. This implicates that all the numbers between 0 and $\delta_D$ have to be summed but this is impossible. *The reals are uncountable; that is: while both the set of all natural numbers and the set of all real numbers are infinite sets, there can be no one-to-one function from the real numbers to the natural numbers* [WikiPedia: real numbers] If they are uncountable that means that we cannot calculate the sum of all the numbers between 0 and the biggest interval. This is proved by the Cantor diagonalisation argument. Uncountable does not mean that we cannot order it. The Field of the real numbers is ordered.

nog mooie tekst schrijven in engels: hier essentie

We kunnen de benefit wel benaderen door een zo groot mogelijke som te nemen van de unit of controls. Uit deze benadering is af te leiden waar de verhouding naartoe zou gaan als de limiet zou genomen worden. -> laten zien met een voorbeeld van Pi en 1.

# Hoofdstuk 6

# Formula

## Playing periodically with virus propagation

This chapter explains how to model a FlipIt game with a virus propagation that infects a network. The first section explains the difference between a normal FlipIt game and a FlipIt game with virus propagation. The next section derives a formula to calculate the benefit for a FlipIt game with a virus propagation. In the last section we calculate the Nash equilibrium for the benefit formula.

## 6.1 Explaining difference between FlipIt with and without virus propagation

zelfde als in vorige chapter:

In chapter 2 the FlipIt game was explained. This chapter starts from the specific case of a non-adaptive continuous FlipIt game where both players play a periodic strategy with a random phase. This choice is motivated by the assumption that in the practical situation of most organisations, the defence strategy is to periodically defend the network. This corresponds to a periodic defender strategy. To simplify the analysis in a first time, a periodic attacker strategy is assumed as well. Further research can investigate the effect of relaxing this assumption.

A FlipIt game consists of a single resource. To represent the security problem, the game now defines its single resource as a computer network with multiple nodes. One of the players, the defender, will try to defend his network. The defender will do this by flipping all the nodes of the network (i.e. the entire resource) in every move he plays. The attacker, the other player, will try to infect all the nodes in the network. The attacker will do this by flipping the node in the graph that can infect all the nodes in the shortest time possible. After dropping a virus on the first node, it takes a while for the virus to infect the entire network. However, since the original FlipIt game works with a single resource that is always flipped entirely, the assumption is made that the attacker is considered to have gained the control over the resource only when all the nodes of the network have been infected, i.e. the entire resource has been flipped.

After dropping a virus on the first resource, it takes a while for the virus to infect the entire network. The time that it takes for the virus to infect every node will be denoted as parameter d. If we want to measure how long it takes for the virus to infect all the nodes in the network, we have to calculate the shortest path from the first infected node to the farthest node. This can be measured by a method that we will explain in section []. Assume that an attacker attacks at time t, then only at time t + d he gains control over the entire network. If the defender flips the network before the period d has elapsed (so, somewhere between t and t+ d), then the attacker will never gain control over the entire network. Using this parameter d, a FlipIt game with virus propagation can be modelled.

## 6.2 Benefit for FlipIt game with virus propagation

Periodic Game with delay for the attacker:

**Case 1:** $\delta_D \leq \delta_A$(The defender plays at least as fast as the attacker.)

Let $r = \frac{\delta_D}{\delta_A}$. The intervals between two consecutive defender's moves have length $\delta_D$. Consider a given defender move interval. The probability over the attacker's phase selection that the attacker moves in this interval is r. Given that the attacker moves within the interval, he moves exactly once within the interval (since $\delta_D \leq \delta_A$) and his move is distributed uniformly at random.

The expected period of attacker control within the interval would be r/2, without considering the delay.

However, because of the delay, the maximal time of control is reduced to $\delta_D - d$. There is a probability of $r$ that the attacker will move in the interval of the defender. The attacker has to play soon enough to gain control, meaning that the attacker has to play during the period of $\delta_D - d$ during the interval of the defender. There is $\frac{\delta_D - d}{\delta_D}$ probability that the attacker will move soon enough which gives the attacker a gain of $\frac{\delta_D - d}{2}$. If the attacker moves after the period of $\delta_D - d$, the gain of the attacker will be zero. The average gain rate of the attacker can be expressed as follows if we look at one interval of the defender:

$$\beta_A(\alpha_D, \alpha_A) = \frac{1}{\delta_D}[\frac{\delta_D}{\delta_A} \cdot \frac{\delta_D - d}{\delta_D} \cdot \frac{\delta_D - d}{2} + \frac{\delta_D}{\delta_A} \cdot \frac{d}{\delta_D} \cdot 0] \tag{6.1}$$

To complete the formula to derive the benefit function, the cost of moving is added. In the second formula we can see the formula of the original FlipIt game.

$$\beta_A(\alpha_D, \alpha_A) = \frac{(\delta_D - d)^2}{2 \cdot \delta_D \delta_A} - k_A \alpha_A \tag{6.2}$$

$$\beta_A(\alpha_D, \alpha_A) = \frac{\delta_D}{2 \cdot \delta_A} - k_A\alpha_A + \frac{d}{\delta_A} + \frac{d^2}{2 \cdot \delta_A\delta_D} \tag{6.3}$$

The benefit of the defender is expressed as follows:

$$\beta_D(\alpha_D, \alpha_A) = 1 - \frac{(\delta_D - d)^2}{2 \cdot \delta_D\delta_A} - k_D\alpha_D \tag{6.4}$$

**Case 2:** $\delta_A \leq \delta_D$ (The attacker plays at least as fast as the defender.)

Let $r = \frac{\delta_D}{\delta_A}$. The intervals between two consecutive attacker's moves have length $\delta_A$. Consider a given attackers move interval. The probability over the attacker's phase selection that the defender moves in this interval is $\frac{\delta_D}{\delta_A} = (1/r)$. Given that the defender moves within the interval, he moves exactly once within the interval (since$\delta_A \leq \delta_D$) and his move is distributed uniformly at random.

## 6.3 something

Periodic Game with delay for the attacker:

**Case 1:** $\delta_D \leq \delta_A$(The defender plays at least as fast as the attacker.)

Let $r = \frac{\delta_D}{\delta_A}$. The intervals between two consecutive defender's moves have length $\delta_D$. Consider a given defender move interval. The probability over the attacker's phase selection that the attacker moves in this interval is r. Given that the attacker moves within the interval, he moves exactly once within the interval (since $\delta_D \leq \delta_A$) and his move is distributed uniformly at random.

The expected period of attacker control within the interval would be r/2, without considering the delay.

However, because of the delay, the maximal time of control is reduced to $\delta_D - d$. If we consider a duration of $\delta_D \cdot \delta_A$ the attacker will play $\delta_D$ times. If the attacker plays soon enough it will get a gain of $\frac{\delta_D - d}{2}$ in $\delta_D - d$ of the cases. In $d$ cases it will receive a gain of zero. This is the case were the duration of the delay causes the defender to play before the attacker can get control over the resource. So the gain of the attacker can be expressed as follows:

$$Gain = \frac{\delta_D - d}{2} \cdot (\delta_D - d) + 0 \cdot d = \frac{\delta_D - d}{2} \cdot (\delta_D - d) \tag{6.5}$$

The benefit of the attacker can be expressed as follows

$$\beta_A(\alpha_D, \alpha_A) = \frac{(\delta_D - d)^2}{2 \cdot \delta_D\delta_A} + k_A \cdot \alpha_A \tag{6.6}$$

$$\beta_A(\alpha_D, \alpha_A) = \frac{\delta_D}{2 \cdot \delta_A} + k_A \cdot \alpha_A + \frac{d}{\delta_A} + \frac{d^2}{2 \cdot \delta_A \delta_D} \tag{6.7}$$

The benefit of the defender is then:

$$\beta_D(\alpha_D, \alpha_A) = 1 - \frac{(\delta_D - d)^2}{2 \cdot \delta_D \cdot \delta_A} + k_D \cdot \alpha_D \tag{6.8}$$

$$\beta_D(\alpha_D, \alpha_A) = 1 - \frac{\delta_D}{2 \cdot \delta_A} + k_D \cdot \alpha_D - \frac{d}{\delta_A} - \frac{d^2}{2 \cdot \delta_A \delta_D} \tag{6.9}$$

**Case 2:** $\delta_A \leq \delta_D$ (The attacker plays at least as fast as the defender.)

Let $r = \frac{\delta_D}{\delta_A}$. The intervals between two consecutive attacker's moves have length $\delta_A$. Consider a given attackers move interval. The probability over the attacker's phase selection that the defender moves in this interval is $\frac{\delta_D}{\delta_A} = (1/r)$. Given that the defender moves within the interval, he moves exactly once within the interval (since$\delta_A \leq \delta_D$) and his move is distributed uniformly at random.

If we consider a duration of $\delta_A \cdot \delta_D$ there is a probability of $\frac{\delta_A}{\delta_D}$ that the defender moves within the interval of the attacker. The defender will then receive an average gain of $\frac{\delta_A}{2}$. There is $1 - \frac{\delta_A}{\delta_D}$ probability that the defender will not move in the interval of the attacker and so the defender will receive no gain. The benefit can be expressed as follows when the defender plays $\delta_D$ times during a duration of $\delta_A \cdot \delta_D$:

$$\beta_D(\alpha_D, \alpha_A) = \frac{1}{\delta_A \delta_D} \cdot \delta_D \cdot [\frac{\delta_A}{\delta_D} \cdot \frac{\delta_A}{2} + [1 - \frac{\delta_A}{\delta_D}] \cdot 0] + k_D \cdot \alpha_D \tag{6.10}$$

$$\beta_D(\alpha_D, \alpha_A) = \frac{\delta_A}{2 \cdot \delta_D} + k_D \cdot \alpha_D$$

same as the FlipIt solution

(6.11)

However, because of the delay, the maximal time of control of the defender is increased by d. In other words, the defender has some benefit time of d before the attacker really gains control over the resource, meaning that the attacker gains control only after $\delta_A + d$ instead of after $\delta_A$. So, when the defender plays, with a probability of $\frac{\delta_A}{\delta_D}$, the expected gain of the defender's control in this interval would be more than half of the period $\delta_A$: it is $\frac{\delta_A + d}{2}$. There is $1 - \frac{\delta_A}{\delta_D}$ probability that the defender will not move in the interval of the attacker but because of the delay the defender will receive a gain of d. So the benefit of the defender can be expressed as:

$$\beta_D(\alpha_D, \alpha_A) = \frac{1}{\delta_A \delta_D} \cdot \delta_D \cdot [\frac{\delta_A}{\delta_D} \cdot \frac{\delta_A + d}{2} + [1 - \frac{\delta_A}{\delta_D}] \cdot d] + k_D \cdot \alpha_D \tag{6.12}$$

$$\beta_D(\alpha_D, \alpha_A) = \frac{\delta_A - d}{2 \cdot \delta_D} + \frac{d}{\delta_A} + k_D \cdot \alpha_D \tag{6.13}$$

The benefit of the attacker is expressed as follows:

$$\beta_A(\alpha_D, \alpha_A) = 1 - [\frac{\delta_A - d}{2 \cdot \delta_D} + \frac{d}{\delta_A}] + k_A \cdot \alpha_A \tag{6.14}$$

# Hoofdstuk 7

# Conclusion

The final chapter contains the overall conclusion. It also contains suggestions for future work and industrial applications.

## 7.1 trala

# Bijlagen

# Bijlage A

# The First Appendix

Appendices hold useful data which is not essential to understand the work done in the master thesis. An example is a (program) source. An appendix can also have sections as well as figures and references[**?** ].

## A.1 More Lorem

# Bijlage B

# The Last Appendix

Appendices are numbered with letters, but the sections and subsections use arabic numerals, as can be seen below.

## B.1 Lorem 20-24

KU Leuven Faculteit Ingenieurswetenschappen 2014 – 2015

# Fiche masterproef

*Student*: Sophie Marien

*Titel*: Gametheory and Cybersecurity: a study FlipIt and multiple resources

*Engelse titel*: Beste masterproef ooit al geschreven

*UDC*: 621.3

*Korte inhoud*:

Hier komt een heel bondig abstract van hooguit 500 woorden. LaTeX commando's mogen hier gebruikt worden. Blanco lijnen (of het commando `\par`) zijn wel niet toegelaten!

Nam dui ligula, fringilla a, euismod sodales, sollicitudin vel, wisi. Morbi auctor lorem non justo. Nam lacus libero, pretium at, lobortis vitae, ultricies et, tellus. Donec aliquet, tortor sed accumsan bibendum, erat ligula aliquet magna, vitae ornare odio metus a mi. Morbi ac orci et nisl hendrerit mollis. Suspendisse ut massa. Cras nec ante. Pellentesque a nulla. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Aliquam tincidunt urna. Nulla ullamcorper vestibulum turpis. Pellentesque cursus luctus mauris.

Thesis voorgedragen tot het behalen van de graad van Master of Science in de ingenieurswetenschappen: computerwetenschappen, hoofdspecialisatie Veilige software

*Promotor*: Prof. dr. ir. Tom Holvoet

*Assessoren*: Ir. W. Eetveel
W. Eetrest

*Begeleider*: Ir. Jonathan Merlevede, Ir. Kristof Coninx