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

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

The first contains a general introduction to the work. The goals are defined and the modus operandi is explained.

1.1 chap

Hoofdstuk 2

Intoduction to GameTheory

In the following paragraph an introduction to game theory is given based on the work of [**?**] and [**?**]. For a more detailed and full introduction to game theory, the reader is referred to [**?**].

2.1 Intro Game Theory

Game theory studies the interaction between independent and self-interested agents. It is a mathematical way of modelling the interactions between two or more agents where the outcomes depend on what everybody does and how it should be structured to lead to good outcomes. For this reason it is very important for economics and also for politics, biology, computer science, philosophy and a variety of other disciplines.

One of the assumptions underlying game theory is that the players of the game, the agents, are independent and self-interested. This does not necessarily mean that they want to harm other agents or that they only care about themselves. Instead it means that each agent has preferences about the states of the world he likes. These preferences are mapped to natural numbers and are called the utility function. The numbers are interpreted as a mathematical measure to tell you how much an agent likes or dislikes the states of the world.
It also explains the impact of uncertainty. When an agent is uncertain about a distribution of outcomes, his utility will describe the expected value of the utility function with respect to the probability of the distribution of the outcomes. For example: with 0.7 probability it will be 7 degrees outside and 0.3 probability it will be 10 degrees. The agent can have a different opinion about that distribution versus another distribution. ().
In a decision game theoretic approach an agent will try to act in such a way to maximise his expected or average utility function. It becomes more complicated when two or more agents want to maximise their utility and whose actions can affect each other utilities. This kind of games are referred to as non cooperative game theory, where the basic modelling unit is the group of agents. The individualistic approach, where the basic modelling is only one agent, is referred as cooperative

uitleggen aan de hand van een voorbeeld

game theory.

There are two standard representations for games. The first one is the Normal Form. The second one is the Extensive Form.

In the following list a couple of terms that will be used throughout the paper.

Players: players are referred as the ones who are the decision makers. It can be a person, a company or an animal.

Actions: actions are what the player can do.

Outcomes:

Utility function: the utility function is the mapping of the level of happiness of an agent about the state of the world to natural numbers.

Strategies: A strategy is the combination of different actions. A pure strategy is only one action.

A game in game theory consists of multiple agents and every agent has a set of actions that he can play.

2.2 Virusses

Stealth Regin's developers put considerable effort into making it highly inconspicuous. Its low key nature means it can potentially be used in espionage campaigns lasting several years. Even when its presence is detected, it is very difficult to ascertain what it is doing. Symantec was only able to analyze the payloads after it decrypted sample files.

It has several "stealth" features. These include anti-forensics capabilities, a custom-built encrypted virtual file system (EVFS), and alternative encryption in the form of a variant of RC5, which isn't commonly used. Regin uses multiple sophisticated means to covertly communicate with the attacker including via ICMP/ping, embedding commands in HTTP cookies, and custom TCP and UDP protocols Ways of defending a network:

- Self-defending networks: The next generation of network security
- Honeynet games: a game theoretic approach to defending network monitors

Many network security threats today are spread over the Internet. The most common include:

Viruses, worms, and Trojan horses Spyware and adware Zero-day attacks, also called zero-hour attacks Hacker attacks Denial of service attacks Data interception and theft Identity theft

Computer virus through mail. Though virus spreading through email is an old technique, it is still effective and is widely used by current viruses and worms. Sending viruses through email has some advantages that are attractive to virus

writers: Sending viruses through email does not require any security holes in computer operating systems or software. Almost everyone who uses computers uses email service. A large number of users have little knowledge of email viruses and trust most email they receive, especially email from their friends [28][29]. Email are private properties like post office letters. Thus correspondent laws or policies are required to permit checking email content for detecting viruses before end users receive email [18].

Send a email with malicious attachment. Only again infected if attachment again opened. Thus this is the action of attacking every neighbour node + also can attack again the node where the virus was coming from. There are also email viruses were the malicious program is hidden in the txt and the attachment does not need to be opened.

2.2.1 Malware

Relevant researches:

- How Viruses and worm can be detected. Difference between UDP en TCP worm propagation

2.3 Conclusion

The final section of the chapter gives an overview of the important results of this chapter. This implies that the introductory chapter and the concluding chapter don't need a conclusion.

Hoofdstuk 3

The FlipIt game

3.1 Extensions on FlipIt

There a various possible ways to extend 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 one resource. An example is gaining access to a system and breaking the password. The model is called FlipThem [2]. Two ways of flipping the resources are used: 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. The difference with FlipThem and this paper is that we introduce a Graph Model in the beginning.
Another extension on FlipIt is done by Pham []. Beside the action Flip their is another action Test. The basic idea is to test with an extra action if the resource has been compromised or not. This action involves also an extra cost. This model is useful if somebody wants to know for example if his password has been compromised or wants to assess the periodic security of a system. In [**?**] [**?**] Laszka et al. they also consider non targeted attacks by non-strategic players and .

citatie needed voor Are We Compromised?

verder aanvullen

In this section, we introduce the game FlipIt [4]. FlipIt is a game introduced by .. .. and Rivest. First we explain the framework of FlipIt and after that the formulas and assumptions that we will make for the game for during the whole paper.

3.2 The First Topic of this Chapter

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 players the Attacker and the Defender. To get the control over the resource, players can flip the resource at any given time. Each move will imply a certain cost. The unique feature of FlipIt is that the move will happen in a stealthy way, meaning that the other player has no clue that 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. Players won't move to frequently. 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 move since it does not result in a change of ownership.

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

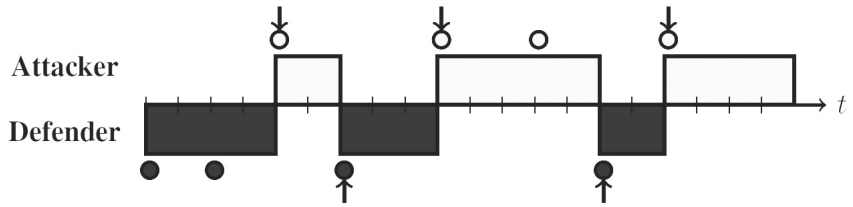
- Non-adaptive (NA): The player does not receive any feedback while flipping.
- Last move (LM): The player finds out the exact time the opponent played the last time.
- Full History (FH): The player finds out the complete history of the opponents move.

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 assumption the strategy of both players will be non-adaptive. None of the players has information of the strategy of the opponent.

3.3 Figures



Figuur 3.1: The FlipIt game where both players are playing periodically

verwijzen naar de figuur 3.1

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

TABEL 3.1: Classes of strategies in FlipIt

3.3.1 Strategies

In this subsection we go through the strategies used in FlipIt and the most important results.

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 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

3.4 Formal definition Game

In this section we provide the formal definition of the game and the notation that we will use throughout the paper.

Players There are two players in the game, one is the defender and the other one is the attacker. They are respectively identified by 0 and 1.

Time The game starts at $t = 0$ and continuous indefinitely as $t \to \infty$. The game is a continuous game.

Game State There is also a time-dependent variable that represents the state of the game. $C = C(t)$ is either 0 if the Game is under control by the defender and 1 if the Game is under control by the attacker. We can also define the state of each resource by C^A and C^D. If $C^A = 1$ then this means that the attacker has control over the resource, and 0 otherwise. For C^D it is visa versa, $C^D = 1 - C^A$.

Graph We represent the company network as a Graph $G =< V, E >$. G is an ordered pair where V denotes the set of resources or nodes in the network and E denotes the set of connections or links, which are a two-element subset of V. We use the notations resources and nodes interleaving in this paper.

aanvullen

We have N resources in the network. $N \in$. This means we can denote the resources by:

$$V \in V_0, V_1, V_2, ..., V_N$$

The set E of connections indicates if there is a link between two resources. We see the links as bidirectional so the total graph is undirected. If there is a link between resource V_n and V_{n+1} then there is also a link between V_{n+1} and V_n.

Moves Both players can make a move in the game. Moves done in a finite numbers of time in any finite time interval. Both players can play at any time they want, they can also play at the same time. If this happens the one that has control over the resource will keep having control over the resource. This makes

beter uitleggen

the game fully symmetric . The sequence of move times are denoted by the following infinite sequence:

$$t = t_1, t_2, t_3, ..$$

Two move times can be the same because we allow players to move at the same time. We can also denote the infinite sequence of times when player i moves. We write this as :

$$t = t_{i,1}, t_{i,2}, t_{i,3}, .. \text{ with } i \in \{0, 1\}$$

The sequences t_1 and t_0 are disjoint subsets of the sequent t. We can also denote who made the kth move by defining a sequence p that denotes the sequence of who played:

$$p = p_1, p_2, p_3, .. \text{ with } p_k \in \{0, 1\}$$

Number of moves $n_i(t$ denotes the number of moves made by player i up to and including time t. This means that

$$n(t) = n_1(t) + n_0(t)$$

is the sum of the number of moves made by the defender and the attacker up to and including time t.

Average move rate We denote $\alpha_i(t)$ as the average move rate by player i:

$$\alpha_i(t) = n_i(t)/t \text{ with } t > 0 \text{ and } i \in \{0, 1\}$$

Period We can also define the period in terms of the average move rate:

$$\delta_i = 1/\alpha_i$$

Cost The cost is an important property of the game. In FlipIt for every player the cost of a move is denoted by k_i. These costs can be very different for every player. In this game we denote the players flipping cost for resource V_N by $c_i^{V_N}$.
For the defender the cost will be either the cost of flipping every resource or the cost of flipping a subgroup of the resources.
For the attacker the cost will be the cost of dropping a virus on a node. The spreading of the virus will not imply an extra cost.

Utility In FlipIt the Gain definition is the utility function. The Gain denotes the total time a player i has gained control over a resource. The Gain G_i denotes players i total gain of a game, wich is the total time the player has gained control over a subset of resources thus controlling the game. If we sum up the total Gain of the attacker and the defender we end up with the time:

nu gain van een resource, moet voor verschillende resources zijn

$$G_1(t) + G_0(t) = t$$

Average gain rate The average gain rate for player i is defined as

$$\gamma_i(t) = G_i(t)/t$$

3.4.1 Our Game parameters

Graph Matrix We represent the graph of the network through a matrix $A = |V| \times |V|$. The (i,j)-entry of the matrix A will have a 1 if there is a connection between node V_i and node V_j. If we are working with an undirected graph the matrix will be symmetric.

Attack Vector We denote $X = 1 \times |V|$ as the attack vector.

Reset vector The reset vector will make sure that the right entries in the matrix become zero.

Cummulative Matrix

State Matrix The State matrix $T(t) = |V| \times |V|$ will keep at every time t the state of the game.

3.5 Conclusion

The final section of the chapter gives an overview of the important results of this chapter. This implies that the introductory chapter and the concluding chapter don't need a conclusion.

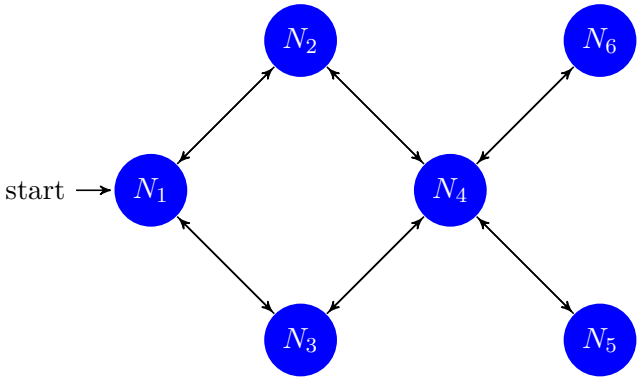
Hoofdstuk 4

Intoduction to GameTheory

4.1 Write down the settings of the game

source: http://en.wikipedia.org/wiki/Adjacency_matrix
We model the network through an undirected Graph $G =< V, E >$ where $|V|$ denotes the number of resources in the network and $|E|$ the number of connections. We can convert this to a adjacent matrix where we can represent which vertices of the graph are neighbours of other vertices.
For our graph we have an $|V| \times |V|$ matrix with on every entry a_{ij} a 1 as value if there is a connection between node V_i and V_j and with zeros its diagonal. Because our graph is undirected we have a symmetric matrix.

"If A is the adjacency matrix of the directed or undirected graph G, then the matrix A^n (i.e., the matrix product of n copies of A) has an interesting interpretation: the entry in row i and column j gives the number of (directed or undirected) walks of length n from vertex i to vertex j. If n is the smallest nonnegative integer, such that for all i ,j , the (i,j)-entry of $A^n > 0$, then n is the distance between vertex i and vertex j." [Wikipedia]



The adjacent matrix becomes this matrix $[A]$:

$$\begin{array}{c} \\ N_1 \\ N_2 \\ N_3 \\ N_4 \\ N_5 \\ N_6 \end{array} \begin{array}{c} \begin{array}{cccccc} N_1 & N_2 & N_3 & N_4 & N_5 & N_6 \end{array} \\ \begin{pmatrix} 0 & 1 & 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 1 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \end{pmatrix} \end{array}$$

Matrix $A \times A = A^2$ becomes the matrix with the number of paths with 2 steps from N_i to N_j: We denote this matrix as matrix *[B]*

$$\begin{array}{c} \\ N_1 \\ N_2 \\ N_3 \\ N_4 \\ N_5 \\ N_6 \end{array} \begin{array}{c} \begin{array}{cccccc} N_1 & N_2 & N_3 & N_4 & N_5 & N_6 \end{array} \\ \begin{pmatrix} 2 & 0 & 0 & 2 & 0 & 0 \\ 0 & 2 & 2 & 0 & 1 & 1 \\ 0 & 2 & 2 & 0 & 1 & 1 \\ 2 & 0 & 0 & 4 & 0 & 0 \\ 0 & 1 & 1 & 0 & 1 & 1 \\ 0 & 1 & 1 & 0 & 1 & 1 \end{pmatrix} \end{array}$$

Matrix $A^2 \times A = A^3$ becomes the matrix with the number of paths with 3 steps from N_i to N_j: We denote this matrix as matrix *[C]*

$$\begin{array}{c} \\ N_1 \\ N_2 \\ N_3 \\ N_4 \\ N_5 \\ N_6 \end{array} \begin{array}{c} \begin{array}{cccccc} N_1 & N_2 & N_3 & N_4 & N_5 & N_6 \end{array} \\ \begin{pmatrix} 0 & 4 & 4 & 0 & 2 & 2 \\ 4 & 0 & 0 & 6 & 0 & 0 \\ 4 & 0 & 0 & 6 & 0 & 0 \\ 0 & 6 & 6 & 0 & 4 & 4 \\ 2 & 0 & 0 & 4 & 0 & 0 \\ 2 & 0 & 0 & 4 & 0 & 0 \end{pmatrix} \end{array}$$

So for A^N every a_{ij} entry gives the number of paths with N steps from N_i to N_j.

With this knowledge we can calculate in how many steps a node is infected. A calculates which nodes are infected after 1 step, A^N calculates which nodes are infected in N steps.. So if we want to know how many nodes are infected after 3 steps we have to add every matrix $(A + A^2 + A^3)$ and see which entry is a non zero entry.

What do we need for an algorithm

Graph network $G =< V, E >$

Graph matrix $[A]$ which is $|V| \times |V|$

Attack vector $[X]$ which is $1 \times |V|$

cummulative matrix $[M]$ which is $|V| \times |V|$

state matrix $[T]$ which is $|V| \times |V|$

Reset vector $[R]$

duration d

time n

rate δ_0 of defender and δ_1 of attacker

Initialisation algorithm:

```
initialisatie
d=0
A=basismatrix
M=A^{0}
n=0
\delta_{0}
\delta_{1}
X
R
controller = defender



Algorithm
n:= n + 1;
Check who is in control? ( through modulo )
if ( defender & controller=defender)
d:= d + 1;

if ( defender & controller=attacker )
G = X \times R  (flippen ten voordele van defender)
d = 0
controller = defender

if ( attacker & controller=defender )
controller=attacker
..

if ( attacker & contoller=attacker )
d:= d + 1
M = M x A
T = T + M
G = X x T
```

Hoofdstuk 5

The Final Chapter

5.1 chap

Hoofdstuk 6

Conclusion

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

6.1 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[1].

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

Bibliografie

[1]

[2] M. F. L. B. Aron Laszka, Gabor Horvath. *5th International Conference, GameSec 2014, Los Angeles, CA, USA, November 6-7, 2014. Proceedings*, page 175.

[3] D. A. Craig and H. T. Nguyen. Wireless Real-Time Head Movement System Using a Personal Digital Assistant (PDA) for Control of a Power Wheelchair. *Conference proceedings : ... Annual International Conference of the IEEE Engineering in Medicine and Biology Society. IEEE Engineering in Medicine and Biology Society. Conference*, 1:772–775, 2005.

[4] R. G. A. J. A. O. R. L. R. Kevin D. Brouwers, Marten van Dijk and N. Triandopoulos. Defending Against the Unkown Enemy: Applying FlipIt to System Security. *Third International Conference, GameSec 2012, Budapest, Hungary, November 5-6, 2012. Proceedings*, 7638:248–263, 2012.

[5] R. G. A. J. A. O. R. L. R. Kevin D. Brouwers, Marten van Dijk and N. Triandopoulos. Defending Against the Unkown Enemy: Applying FlipIt to System Security. *Third International Conference, GameSec 2012, Budapest, Hungary, November 5-6, 2012. Proceedings*, 7638:248–263, 2012.

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