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

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

bib referenties in orde brengen

1.1 Introduction

Security is an important asset in Computer Science. Defending a network of a company is not an easy job. To prevent intruders it can make use of firewalls, routers, IDS systems, virus scans, and other defence mechanisms. Unfortunately technology is growing fast and attacks are getting more sophisticated and the causes of these attacks can be very different. Companies are often the victim of targeted attacks. In a security report of 2014, , states that 80% of the companies are the victims of targeted attacks. Many companies don't see themselves as a target, but sometimes they might be collateral, the target on the way to the real target. This means that everybody can be a target. Corporate networks should continuously defend themselves against outside invaders such as viruses and worms. By doing so the network administrator can keep the network as malware-free as possible. If there is an intruder managed to penetrate the network then the network manager this intruder trying to get out as quickly as possible. This is not always easy. Especially when the intruders secretly sneak and then spread rapidly. In this paper we will work further on the work made by Marten van Dijk, Ari Juels, Alina Oprea and Ronals L. Rivest who wrote a report on the Game FlipIt. FlipIt is a the game of " Stealthy Takeovers". It models a game by means of two players, the attacker and the defender. Both can gain control over a single shared resource by flipping it. The most important property of the game is that the flipping happens stealthy. This means that the players have no clue about when the other player moves. The goal of the game is to maximise the time the player controls the resource minus the average cost of the flipping.

iets tussen nog

verwijzing naar report

verwijzing naar FlipIT

1.1.1 Motivation of the game

1.1.2 Contributions and results

1.1.3 Conclusions

The "I love you"virus is an example of a virus that spreads quickly. This virus propagates via mail systems. If someone opens an email with "I love you"virus in annex this virus spreads itself by sending a mail itself to everyone in your contact list. So the virus can multiply rapidly and eventually a business network shut down by the heavy traffic. In this example, there is a need human interaction to spread the virus to do. If no one opens the virus can not spread the mail. Unfortunately, there are viruses that can spread without human interaction. These viruses are referred to as worms. A worm is also a computer program that replicates itself to spread to other computers so. Via a computer network, copies of the worm forwarded without an intermediary is used for. The worm will use vulnerabilities to infect other computers. Most worms are designed to spread out and just try not to make any changes to the systems that they pass. These worms can still inflict damage by increased network traffic they generate. Worms that contain Harm damage a program to install a backdoor or a rootkit on the infected computers. Backdoors and rootkits ensure that future use can be made of the infected computers. The Stuxnetworm is a very famous worm. Initially this worm spread via infected USB sticks and from then it could spread through the Internet to other computers. The purpose of the Stuxnetworm was broken to run the centrifuges in nuclear reactors. Many reactors have been infected. From the standpoint of the defender, it is very important to respond as quickly as possible so that the worm can not spread quickly.

1.2 introduction number 2

(We live in an era) 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. Since 2009, the number of reported security attacks has increased 66%, year over year. . These numbers only represent the attacks that are detected. In 2014 117,339 attacks where coming in daily. Many of those attacks have a different cause. Some of them can be benign, others can be harmful. Many companies are unaware of all the attacks. Some of them think that they are not a target, but they might be a target on the way to a real target. Recently there where some high profiled targeted attacks which have been revealed. (Belgacom). Targeted attacks are ... The *Kill Chain* is a concept by Lockheed Martin Corporation, explained in the whitepaper . It explains the different phases of a typical attack from the view of an attacker. It also outlines the typical attacker activities on the right. This model is very useful to define the different moments of the life cycle of an attack and when a company should act to defend itself. In this paper we would like to prevent the viruses of spreading into the network system of a company. This means that we have to act in phase Installation, Command and Control and Action on Objectives of the kill chain.

security report van pwc

withepaper toevoegen

Security is an important asset in Computer Science. Defending a network of a company is not an easy job. Malicious people will try to To prevent intruders it can make use of firewalls, routers, IDS systems, virus scans, and other defence mechanisms. Unfortunately technology is growing fast and attacks are getting more sophisticated and the causes of these attacks can be very different. Companies are often the victim of targeted attacks. In a security report of 2014, , states that 80% of the companies are the victims of targeted attacks. Many companies don't see themselves as a target, but sometimes they might be collateral, the target on the way to the real target. This means that everybody can be a target. Corporate networks should continuously defend themselves against outside invaders and targeted attacks. Researchers have already investigated the situations through the FlipIt game in which a system is continuously compromised by an attacker through targeted attacks. FlipIt is a the game of " Stealthy Takeovers". It models a game by means of two players, the attacker and the defender. Both can gain control over a single shared resource by flipping it. The most important property of the game is that the flipping happens stealthy. This means that the players have no clue about when the other player moves and has control over the shared resource. The goal of the game is to maximise the time the player controls the resource minus the average cost of the number of flipping. In this paper we model a company network through multiple shared resources and a flip from the attacker that drops a virus that will spread itself autonomously. We show that ...

iets tussen nog

verwijzing naar report

Hoofdstuk 2

The FlipIt game

2.1 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 [**?**]. 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 [**?**]. 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.

2.2 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 two players the Attacker and the Defender. To get control over the resource, the players can flip the resource at any given time. A flip will be regarded as a move from a player. Each move will imply a certain cost and the cost can vary for each player. Both players will try to minimize their cost. By adding a cost 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 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. 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:

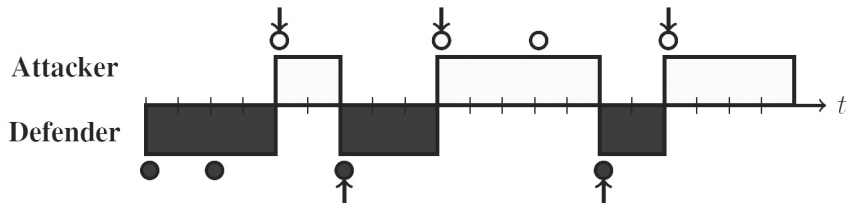
- Non-adaptive (NA): The player does not receive any feedback during the game while flipping.
- Last move (LM): When a player flips it will find out the exact time that the opponent played the last time.
- Full History (FH): When a player flips it will find out the whole history of the opponents move.

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

2.3 Figures



Figuur 2.1: The FlipIt game where both players are playing periodically

verwijzen naar de figuur 2.1

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

2.3.1 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 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

2.3.2 Actions of the attacker

A virus has different kind of ways of making his way through a company network. We will describe the different ways of how the virus can propagate. For start we will say that the virus or worm will be dropped on Node i and that it has k numbers of neighbours.

1. Node i is infected and will spread the virus or worm to every k neighbours and will stop infecting the neighbours in the next step
2. Node i is infected and will spread the virus or worm to every k neighbours and will keep on spreading the virus to the same neighbours in every next step

3. Node i is infected and will spread the virus to only one of the k neighbours and will stop infecting another neighbour in the next step

4. Node i is infected and will spread the virus to only one of the k neighbours and in the next step it will infect another one of the k neighbours

In the game that will be modelled in the paper we will use the settings of the first spreading method. We will not use method 2 because this kind of propagation will float the network. Because we use the settings of a mail system and contact in a mailing list the method of 3 and 4 are not used.
In the first method the node that has been infected can be again infected. If one of the neighbours infects the node again the node will infect his neighbours again. By using this spreading method we have three distinct states in which a node can be situated. An *infected state*, a *clean state* and a *spreading state*. An infected state means that the node is infected and will not spread the virus to its neighbours, a clean state means that the node is not infected on that moment and a spreading state means that the node is infected and that it will spread the virus or worm to its neighbours in the next step. We can argument this kind of propagation through a mail worm.

voorbeeld geven van zo een worm

The Attacker itself has two different ways of attacking the company network. It will only infected one node of the network and will wait for the virus to spread itself through the network. We will model two ways of attacks of an Attacker:

1. The attacker drops the virus on a random node on the network

2. The attacker drops the virus on a targeted node on the network

The attacker in this game will put a virus or worm on one of the nodes in the network. (This will happen at random.) The attacker does not know on which node the virus will be dropped. We will use this randomness because most viruses are spread via a usb stick or a shared resource. If we use this spreading method where we have a targeted attack the attacker will have more information about the network.

feit uit security rapport symantec

The attacker can choose at which rate it will drop a virus on one of the nodes on the network. The cost of dropping a virus will be the same. It will not increase. If it will increase this means that the attacker will eventually drop out of the game because it becomes to expensive.
The attacker is in control over the game if it manages to infect a subset of all the resources of the company network.

2.3.3 Actions of the defender

The attacker wants to protect all the nodes of his network. It can do so by getting back control over the resources. We will assume that the defender of the network has knowledge over his own network. Which is convenient in the real world because a company has to know how his infrastructure looks like.

The defender has two possible ways of defending its network:

1. The defender flips all the nodes of his network
2. The defender will flip a subset of the nodes of his network

The cost of flipping all the nods of the network will be greater than the cost of flipping a subset of nodes. We make this assumption because otherwise it will be beneficial for the defender to always flip all the nodes in the network.

We will also make the assumption that as a defender flips a node the node can get infected again. A flip will not be correlated to a patch but to a clean-up. Another setting of the game can be that the flip of the defender is equal to a patch and that the resource cannot be infected any more. But with this case we deviate from the flipIt game, because the attacker cannot flip the resource any more. Unless we work with different virusses every time the attacker flips. We start with the less complex game of flipping is equal to a clean-up.

waarom geen patch, wormen kunnen veranderen gaandeweg

andere mogelijkheid:

2.3.4 Strategies of both players

We explained what the actions of each player are.

2.4 Formal definition Game

In this section we provide a 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.

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

aanvullen

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

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.

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 start at $t = 0$ with the defender who has control over the game. We do this because we assume that the defender will only put the network online without having a virus or worm in it. The Attacker can gain control over the game when it compromises a subset s of the resources. The subset s is a minimum of 1 resource and a maximum of all the resources N.
We can also define the state of each resource by C_N^A and C_N^D. If $C_N^A = 1$ then this means that the attacker has control over the resource, and 0 otherwise. For C_N^D it is visa versa, $C_N^D = 1 - C_N^A$.

deze variabele nodig ja of nee ? JA

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

beter uitleggen

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

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:

$$\delta_i = 1/\alpha_i$$

Who played last We know who played last by taking the modulo with the period. Z_i represents the time since the last flip of player i. We can also denote the time since the last flip of player i on resource r by Z_i^N. For a non adaptive game, period deterministic: At time $t = n$ is $Z_i = nmodi$. .

er kan nog steeds tegelijk geflipt zijn maar dan hebben ze wel geflipt

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, which is the total time the player has gained control over a subset of resources thus controlling the game. This is denoted by the following:

nu gain van een resource, moet voor verschillende resources zijn

$$G_i(t) = \int_0^t C_i(x)dx$$

If we sum up the total Gain of the attacker and the defender we end up with the time:

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

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

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

2.4.1 Formal definition

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. It will be a vector with only zeros. The attacker will place a virus on a node V. This will be denoted by the Vth entry in the vector that is changed by a 1.

Reset vector The reset vector will make sure that the right entries in the matrix become zero. If the defender flips every node every time it flips then the attack vector will be 0.

Cummulative Matrix This matrix will keep record of the propagation of the virus through the network.

State Matrix The State matrix $T(t) = 1 \times |V|$ will keep at every time t the state of the game and denote which node at time t is infected with the virus. At time $t = 0$ the State Matrix will be the null matrix.

De eerste infectie is de attack vector * Graph matrix .

2.5 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 3

Intoduction to GameTheory

In the following paragraph an introduction to game theory is given based on the work of leyton2008essentials and Coursera. For a more detailed and full introduction to game theory, the reader is referred to leyton2008essentials.

3.1 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

players rationeel en max outcomes

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.

strategien en acties definieren

Best response ook uitleggen?

Voorbeeld ook nog uitleggen?

One of the solution concepts in Game Theory for non-cooperative games is a Nash Equilibrium. A Nash Equilibrium is a subset of outcomes that can be interesting to analyse a game. For a Nash Equilibrium each player has a consist list of actions and each player's action maximizes his or her payoff given the actions of the other players. Nobody has the incentive to change his or her action if an equilibrium profile is played. In general we can say that a Nash Equilibrium is a stable strategy profile: each player is considered to know the equilibrium strategies of the other players and no player would want to change his own strategy if he knows the strategies of the other players.

3.2 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.

Spy vs Spy: Aldrich Ames was a CIA Counter-Intelligence officer. He was also a spy feeding valuable intelligence to the Soviets and compromising US intelligence operations in the Soviet Union. He operated for 9 years before the CIA recognized that they had a spy and began an investigation and determined that he was the leak. This strategic situation is the same one faced by computer networks, drug cartels, intelligence agencies and guerrilla networks.

All such organisations have a reasonable expectation that trusted personal/systems will eventually be recruited/captured by enemy organisations. Therefore such organisations must consume valuable resources to discover such betrayals and thereby regain secrecy. The question is then given the possible threats how often and at what cost should they spend resources on investigations/spy hunts/virus scans. This is where flipIt comes in.

FLIPIT: The Game of "Stealthy Takeover:" FlipIt was created to model these sorts of strategic situations and to study the best courses of action. Specifically flipIt was motivated by the recent interest in and success of Advanced Persistent Threats, or APT.

The basic idea is that given the current experience that perfect protection of trusted resources is unattainable, lets think about how we can optimally manage compromises of the our most trusted systems.

Rules

Two players, player X (blue) and player Y (red) attempt to maintain control over a shared resource. At anytime in the game each player is allowed to play 'flip'. The only way a player can learn the state of the game (who is in control) is when they

play flip. If a player is in control of the resource and they play flip they remain in control of the resource. If a player is not in control of the resource and they play flip they gain control of the resource. Players gain points for the length of time they control the resource. Players lose points every time they play flip. This reflects the situation that the CIA is placed in with regard to moles/enemy spies. They don't know if they have been compromised. They can perform an investigation and determine if they have been compromised, also catching the spy in the act, but this action is very expensive. That is, the CIA has to trade off between remaining"mole free" (a good) and investigations (an expense).

Winning: How do you win a fair game of flipIt against intelligent adaptive human adversaries? I'm not sure.

In the real world what is the best move given that the other players can secretly capture/corrupt your most trusted personal/systems? Rives suggests in his talk that you: Be prepared to deal with repeated total failure (loss of control). Play fast! Aim to make opponent drop out! Arrange game so that your moves cost much less than your opponent's!

3.2.1 Malware

Relevant researches:

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

3.3 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.

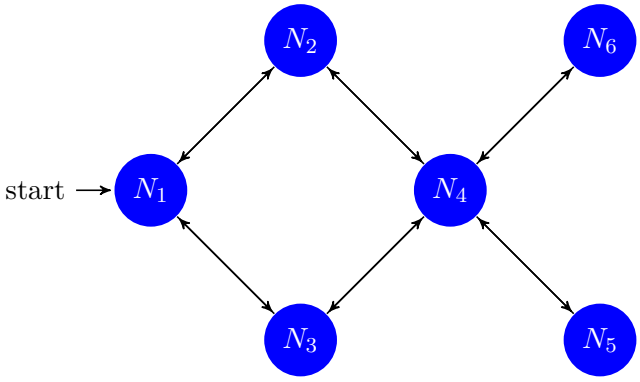
Hoofdstuk 4

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

Intro to virus

5.1 tekst

In this section we are going to elaborate how we are going to model a Flipit game with multiple resources and a virus that propagates and infects the resources. We come up with a formula for the normal FlipIt (normal as in specific parameters and no normalising over the first interval) and then reform it to a FlipIt game with a virus.

5.1.1 FlipIt with a virus

In the previous version the FlipIt game is already been explained. In this section we will introduce the virus propagation. An attacker will drop a virus on one of the resources that is available. The virus will then spread itself to the neighbour resources. The attacker will only gain control over the whole network, in general the game, when it has infected a certain amount of resources. We will call this amount for now ”d”. If we want to measure how many time it takes for the virus to infect ”d” resources, we have to calculate the shortest path to the ”d” th node. If we know how much time it takes to control ”d” resources, we can model a normal FlipIt game but with a delay. The model will not be completely a FlipIt game with a delay, because if the delay is bigger than the period of the attacker, the attacker will gain no control. If it would be with the delay the attacker would gain control after the defender flipt again. This is explained in figure 5.1. In the next section we are going to describe what the setting of the game is.

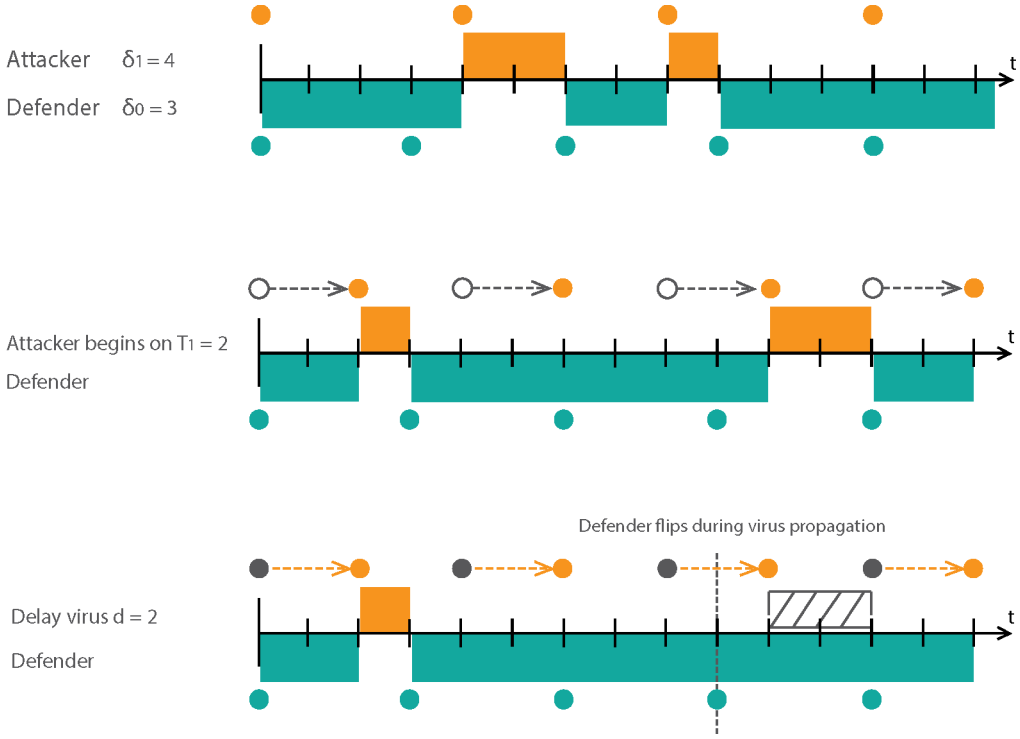
laten zien met een figuur

5.1.2 chap

The setting of the game that we are going to play is one with multiple resources. When the defender flips it will always flip all the resources. The attacker will flip the node in the graph that can infect all the nodes in the shortest time possible. The attacker will gain the control over the resources when all the resources are infected. So ”d” will be the shortest path to the furthest node. We will model ”d” in time units. .

moet misschien niet ?

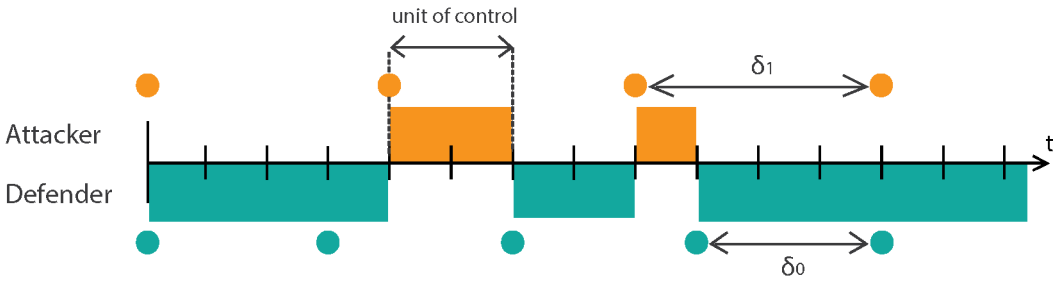
Figuur 5.1: Difference in a FlipIt game between delay caused by a virus and a delay for the Attacker



5.1.3 define formula

Their is a definition given by the writers of the paper FlipIt, but we want to add the property of a virus to the game so we are trying to find a formula that defines a game by counting the amount of time one of the players has control.

Figuur 5.2: Defining unit of control



for $delta_1 > delta_0$ (The defender moves faster than the attacker.)

We will call every continuous time step that one of the players has control over the resource a unit of control. Next we will divide our time line of our FlipIt game

into different intervals of size $delta_1$. So every time the attacker takes control we have the start of a new interval. Because the defender will move faster than the attacker it will at least move 1 time during the interval. Because the attacker only moves at the start of the interval we can say that the defender will always end as being in control of the resource. This brings us to the next forumula to calculate the length of a unit of control of the attacker. For every real number $delta_1$ and $delta_0$ and every n $\in$ of the natural numbers:

$$\Delta A = [(1 - n) \times \delta_1] mod \delta_0 \tag{5.1}$$

where n is the number of the unit of control that you want to calculate of the attacker.

In this formula we multiply the number of unit of control that we want with the period of Attacker (δ_1). The 1 - n is when we count beginning from 1. If we start counting starting at 0 we leave the 1 and the formula becomes:

$$\Delta A = [(-n) \times \delta_1] mod \delta_0 \tag{5.2}$$

We know that each interval ends with the control for the defender. This means that we only need to know how long the defender had control during that interval and take the rest. The rest will be the amount of time the attacker has control in that interval. We take the rest by doing the $mod\delta_0$

This means that if we want to calculate the gain of the attacker we need to calculate the time the attacker has control over the total amount of time that has passed by. For δ_0 and $\delta_1 \in$ Rational numbers we can see that we have a cycle. A pattern that comes back over and over again. That is when the amount of time is a multiple of δ_0 and δ_1 or the largest common multiplier (lcm). At this point the Attacker and the Defender move at the same time what brings us back to the beginning. So to calculate the gain of δ_0 and $\delta_1 \in$ Rational numbers we need to calculate the amount of control units of the attacker that go into the length of time units equal to the lcm of δ_0 and δ_1. After this calculation we divide it by the lcm of δ_0 and δ_1, which is the total amount of time and the amount of time for one cycle. This gives us the following formula:

$$a = \frac{\delta_0}{lcm(\delta_0, \delta_1)} \tag{5.3}$$

$$\frac{\sum_{i=0}^{a} \{[(1 - i) \times \delta_1] mod \delta_0\}}{lcm(\delta_0, \delta_1)} \tag{5.4}$$

We can also define a formula without the greatest common divider. Every δ_0 and δ_1 have to be written in a fraction:

$$\delta_0 = \frac{a}{b} \quad and \quad \delta_1 = \frac{c}{d} \tag{5.5}$$

If δ_0 or δ_1 is a Geheel getal then b or d will be 1. The formula for the gain becomes different:

formule zoek zonder lcm e gcd

If δ_0 and/or δ_1 is an Irrational number: An irrational number $i \neq \frac{a}{b}$ with $b \in Z, a \in N$ Because we cannot write i in a fraction, this means that we won't have a cycle. If we would have a cycle that means that we do have a number that divides i. If we don't have a cycle it goes on forever. Meaning that it goes on to infinity. This also means that no number will be repeated two times. If it does that means that their is repetition, meaning again that their is a cycle. We can conclude that if we have no cycle and no number will be repeated twice, that it will enumerate every number between 0 and the biggest interval (which is δ_0). *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.

interval definie-ren

What we can do is take the limit, count as many control units of time of the attacker and divide it by the greatest amount of time. We can see that this eventually will result to the solution given by the writers of FlipIt. [r/2]. Example delta1 Pi and delta0 1. Grafiek voor maken.

5.1.4 Formula with a virus propagation

Now we can define how we can use the previous formula to calculate the benefit of the attacker with a virus propagation. As mentioned before we have a parameter d that defines the virus propagation. It will take an amount of time d before the attacker gains control over all the resources. We know how to calculate each unit of control of the attacker. If it takes d time before it can take control we have to subtract d form each unit of control. It can be that the unit of control is less than d. This means that we will have a negative number of time. In this case this means that the defender has flipped all the resources before the attacker good gain all the control. So if we want to calculate the benefit we can only take unit of control that are bigger than 0. So the formula becomes:

$$\frac{\sum_{i=0}^{\delta_0}\{[(1-i)\times\delta_1] mod \delta_0 - d\} > 0\}}{delta_0 \times delta_1} \tag{5.6}$$

5.1.5 Random phase

For now we assumed that the first move of both players started at phase $t = 0$. In the FlipIt game the first move is chosen uniformly over the interval [0,δ]. We will call this first move the phase move and denote it by T_1 for the attacker and T_0 for the defender. We will have to integrate these two phases into the formula.

Hoofdstuk 6

APT

6.1 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 7

The Final Chapter

7.1 chap

Hoofdstuk 8

Conclusion

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

8.1 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

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