|  |  |
| --- | --- |
| D:\Dropbox\HoneyJar Business\Referentie\Afbeeldingen\hive.jpg  End Report  Honeypot and Sandbox for Malware studies | Authors: A. Pluimers, M. el Ouahbi  Student numbers: 314831, 423819  Study programme: HBO-ICT Business  Educational Institute: Saxion  Date: June 2018  First supervisor: E. Salomons  Second supervisor: J.M. Pedersen |

# Preface

The existence of this end report stems from our common interest in Malware. When we signed up for this project, we were not aware of the international aspect that was involved, which turned out to be a nice surprise once we got admitted to following the course. We met for the first time as we come from different universities, and we flew to Riga to attend the first seminar regarding this project and to meet the rest of the group. This end report is the result of all the work we put in as a group during the course of roughly six months.

This end report is mainly aimed towards our supervisors, secondly we hope this end report will be used for the further development of the Honeyjar project in forthcoming iterations, as the current model we designed is still a fairly conceptual one. This document is also relevant to the stakeholders we interviewed as part of our field research, and will thus be shared with them.

As far as division of tasks go amongst us Business students, we tried to work together on the same aspects of the project as much as possible, meeting frequently in person as well as through online collaboration. A lot of online collaboration was inevitable due to the international nature of this project.

We’d like to thank our supervisors from Saxion for their continuous support and enthousiasm, E. Salomons, N. van Hattum-Janssen, C. Mensink, as well as our international supervisor J.M. Pedersen for keeping the scope of the project realistic and being a nice guy in general.

# Management summary

## Clients

Saxion University of Applied Sciences

Erasmus-EPIC

## Problem definition

Companies as well as regular consumers have a lot of sensitive data stored on their smartphones. This in itself is worrisome, however the popularity of smartphones is rising as well. At this point in time smartphones outsell PC’s. Computers have been susceptible to Malware since they were first introduced on the market, however it seems that Android phones, while vastly gaining popularity over PC’s, are still relatively underprotected to Malware attacks.

To exploit the fact that the market hasn’t really focused on security on Android phones yet, a basic Honeypot system for Android was made. A Honeypot, as to be derived from its name, is a virtual machine that emulates a system, in our case an Android Phone, presenting it in such a way that it becomes interesting for Malware to attack.

As Business students, it was our task to analyse this Honeyjar-system, and decide whether and how such a system can be made profitable.

## Method of research

Through a combination of desk-and field research, research was conducted. This information was then transformed to usable data using a coding system. Also during the course of the project there were two meetings with the group in Riga and Barcelona, to facilitate interviews and promote group work. There was weekly contact through Discord, to share knowledge and keep the group updated.

## Notable conclusions

The research made clear that in our current time the sales of Android phones is increasing, as well as the looming threat of Malware, most notably Ransomware, on mobile devices. Research also points out that this is fairly new ground, as businesses are starting to become aware of the importance of security in their infrastructure, but there is no solid solution aimed at Ransomware on Android phones at this point in time.

This is where the Honeyjar shines, as it technically is a trendsetting product with no competitors that offer the same functionalities.

# Contents

[1 Preface 1](#_Toc517616557)

[2 Management summary 2](#_Toc517616558)

[3 Contents 3](#_Toc517616563)

[4 Figures 3](#_Toc517616564)

[5 Organization 1](#_Toc517616565)

[6 Theoretical framework 3](#_Toc517616579)

[7 Study design and execution 5](#_Toc517616590)

[8 Subquestion I 8](#_Toc517616603)

[9 Subquestion II 16](#_Toc517616630)

[10 Subquestion III 27](#_Toc517616652)

[11 Conclusion 31](#_Toc517616678)

[12 Recommendations 33](#_Toc517616686)

[13 Bibliography 34](#_Toc517616687)

[14 Appendixes 37](#_Toc517616688)

# Figures

[Figure 1: Petya victim 10](file:///D:\Dropbox\HoneyJar%20Business\Eindverslag\Eindverslag%20HoneyJar%20v0.9.docx#_Toc517616697)

[Figure 2: Android/-FakeAV.E victim 10](file:///D:\Dropbox\HoneyJar%20Business\Eindverslag\Eindverslag%20HoneyJar%20v0.9.docx#_Toc517616698)

[Figure 3: DoubleLocker victim 11](file:///D:\Dropbox\HoneyJar%20Business\Eindverslag\Eindverslag%20HoneyJar%20v0.9.docx#_Toc517616699)

[Figure 4: The HTC Dream 12](file:///D:\Dropbox\HoneyJar%20Business\Eindverslag\Eindverslag%20HoneyJar%20v0.9.docx#_Toc517616700)

[Figure 5: Mobile infections per OS 15](file:///D:\Dropbox\HoneyJar%20Business\Eindverslag\Eindverslag%20HoneyJar%20v0.9.docx#_Toc517616701)

[Figure 6: Phone OS Distribution in Europe 16](file:///D:\Dropbox\HoneyJar%20Business\Eindverslag\Eindverslag%20HoneyJar%20v0.9.docx#_Toc517616702)

[Figure 7: Global market share held by RIM 17](file:///D:\Dropbox\HoneyJar%20Business\Eindverslag\Eindverslag%20HoneyJar%20v0.9.docx#_Toc517616703)

[Figure 8: Phone OS distribution in businesses 17](file:///D:\Dropbox\HoneyJar%20Business\Eindverslag\Eindverslag%20HoneyJar%20v0.9.docx#_Toc517616704)

[Figure 9: Beta version of the Honeyjar GUI 19](file:///D:\Dropbox\HoneyJar%20Business\Eindverslag\Eindverslag%20HoneyJar%20v0.9.docx#_Toc517616705)

[Figure 10: A Shadow Honeypot 22](#_Toc517616706)

[Figure 11: Overview per VM 23](#_Toc517616707)

[Figure 12: Different plans for different customers 25](file:///D:\Dropbox\HoneyJar%20Business\Eindverslag\Eindverslag%20HoneyJar%20v0.9.docx#_Toc517616708)

[Figure 13: Honeyjar designed in BPMN 26](#_Toc517616709)

[Figure 14: Stakeholder Points of Interest 29](#_Toc517616710)

# Organization

## Group composition

The Honeyjar group consists of the following members, each respectively with their occupation and university.

|  |  |  |
| --- | --- | --- |
| Name | Occupation | University |
| Magnus Stensli | Programmer | Aalborg University, Denmark |
| Daniel Britze | Programmer | Aalborg University, Denmark |
| Jacob Vejlin Jensen | Programmer | Aalborg University, Denmark |
| Peter Moller | Programmer | Aalborg University, Denmark |
| Robert Nielsen | Programmer | Aalborg University, Denmark |
| Ahmet Turkmen | Network analyst | Abdullah Gul University, Turkey |
| Alexander Pluimers | Business | Saxion, The Netherlands |
| Morcel el Ouahbi | Business | Saxion, The Netherlands |

Table 1: Group Composition

### Aalborg University

By far the largest amount of students in this project hail from Denmark. These students are in their second year of their studies, and take care of all programming duties related to this project.

### Abdullah Gul University

Ahmet Turkmen is the network specialist in this project. His focus is the Machine Learning algorithm and the connectivity of our system.

### Saxion University of Applied Sciences

Last but not least, our role in the group is to make the Honeyjar sellable to a target audience, i.e. the business-aspect.

### Communication

Initially, a seminar took place in Riga, Latvia, where the group met for the first time. During this first week a lot of base rules about communication where established in a contract. Also during this week, a general idea about what the Honeyjar project should be was decided by the end of the week.

Halfway through the project, the group met once again in Barcelona, which turned out to be very refreshing to get a clear goal of the project, and to align all the expectations.

As this project takes place among different countries, it is essential that a clear method of communication is used. Keeping that in mind, the group decided on having weekly virtual meetings at 15:00 PM EST. Apart from one instance where Daylight Savings caused some confusion with one student showing up one hour late, this turned out to be an excellent way of communicating.

For every meeting, an agenda was made beforehand with a list of attendees, date, and subjects to discuss. As stated in an agreement that was made in Riga, attendees of the meeting were open for discussion, gave others the room to talk, and treated each other respectfully in a fairly laidback atmosphere. Morcel, as the group leader, directed the meetings, while Alexander made minutes.

## Supervision

In this project there is supervision on multiple levels.

### Erasmus+

Starting off from a macro level, the main “client” for this project is Erasmus+, specifically the EPIC division. Erasmus+ is a European government-funded program aimed at creating international job opportunities, and bringing educational institutes together. Erasmus+ did not directly supervise the results of the project, but rather if the requirements were met on an organizational level, i.e.:

* Attendance lists that need to be signed
* Financial aspects
* Feedback forms about EPIC

### J. M. Pedersen

Besides being a representative of EPIC and leading the Danish part of the Honeyjar group, J.M. Pedersen also functions as a general supervisor of the Honeyjar group. Mr. Pedersen has been actively giving feedback, participating in virtual meetings, and guiding the group during our meetings in Barcelona and Riga whenever the scope was about to be changed for the hundredth time.

### E. Salomons

Then finally, Mr. Salomons is our direct supervisor from Saxion. Weekly meetings were had between the Business students and Mr. Salomons, regarding feedback on documents, and the direction of the project in the big scheme of things. Three presentations about the Honeyjar group’s findings were presented during the course of this project, which were also attended by N. van Hattum-Janssen and C. Mensink, who both are familiar with Erasmus+ as well, giving the Honeyjar group valuable feedback.

## Tools

To facilitate communication and a working environment, the following tools were used.

### Discord

For our main communication channels, Discord was used. This was a logical choice as it featured easy file transfer, has improved sound quality during calls as opposed to Skype, and everyone in the project is familiar with it.

### GitHub

Seeing as a lot of programming takes place in the project, GitHub was a logical choice. Apart from coding and programming, GitHub is also used for meeting documents and research items for the Danish group and Ahmet.

### Dropbox

As far as us Business students goes, we decided on using Dropbox for file transfer between the two of us. The reason for this is because Dropbox is very easy and hassle-free, whereas GitHub tends to prompt unlogical errors which cost time to repair.

# Theoretical framework

In this chapter we will define the majority of concepts that are presented in this report.

## Android OS

The native Operating System that comes pre-installed on all Android-based smartphones. Android OS has relatively many modifying possibilities as opposed to its Apple counterpart OS.

## Apple iOS

Comes preinstalled on all Apple phones. While this OS looks very sleek, a lot of the functionalities that it’s Android counterpart has are very limited. This is a blessing and a curse, as these limitations can be annoying – they also guarantee a level of security and quality.

## BPMN

Stands for Business Process Modelling. This is a method of designing flowchart-type processes inside a business, with one of the main goals it being understandable by any kind of employee within said organization. BPMN is our tool of preference for showing complex processes because of this – it makes complex systems clear.

## GUI

Stands for Graphical User Interface and essentially is the visual aspect of a system with which a user has to interact. A GUI is broadly implementable, for example a GUI can be used in a web-setting as well as a stand-alone program.

## Honeyjar

The name of this international project. Not to be confused with “Honeypot”.

## Honeypot

A system that pretends to be a “real” user, aimed at luring Malware for analysis.

## Machine Learning

An algorithm aimed at self-improving through the analysis of data. Whenever data is fed into this algorithm, it gets more intelligent and able to detect specific types of data using this knowledge.

## Malware

Software with the intention to inflict harm upon other systems.

## Packet

A cluster of metadata. A packet describes the specifics of data, such as version, length, protocol, source, and destination.

## Ransomware

A type of Malware that locks a victims’ system, threatening to publish the data on the system unless a fee is paid.

# Study design and execution

## Goal

The goal of the project is to create a system that is able to protect phones from malicious attacks. This system should be able to detect and prevent malicious activity on phones and thus be able to separate good traffic from bad traffic.

Since this goal is very broad it has been narrowed down to the following goal: Detect and prevent Ransomware on Android phones.

While this is narrowed down a lot more it is still too big of a project to finish in five months. So instead of creating a full-on solution the project group will be delivering a blue print for a product. With this blueprint, potential further projects can continue the work that has been done in this project and work towards the goal to protect all type of phones from all sorts of malicious attacks.

## Problem definition

Smartphones have become more and more embedded into the daily lives of everyone around the world. Everyone is able to stay connected with anyone around the globe. However this brings a lot of risks with it as well. Cybersecurity is becoming more and more of a hot topic lately and large companies are investing more and more money in it since they are becoming aware that good cybersecurity is fundamental for a stable business.

Phones is something that has been gaining popularity only recently really. Smartphones are something of the last ten years but are here to stay for a long time to come. However cybersecurity on phones is not something that is self-explanatory, yet. This project is set to create a solution to this problem and the following research questions have been made to answer:

*“How can a Honeyjar-system for Android be made relevant for our target audience and turned into a* *profitable product?”*

And this main question is divided in three more sub questions.

1. What is a Honeyjar system and what is it’s relevance in current times?
2. Why and how should our targeted audience use a Honeyjar system?
3. What can a Honeyjar mean for our target audience?

## **Accountability method**

In this paragraph we will elaborate on the methods we chose, and why we chose them.

### Desk research

As all project members are located in different countries, some necessary research has to be done by collecting information through the internet, books, and other sources. This includes research such as (but not limited to):

* Tutorials on Virtual Machines, servers, Android, etc.
* Manuals for the server
* Documents describing Honeyjar systems
* Articles
* Forums
* Troubleshooting

Desk research is part of any project where a subject has to be researched.

### Field research

For information that needs to be actual and from a unique perspective, field research was performed. Information was collected through interviews with companies focused on security, as well as other relevant people.

Through the conducting of interviews information was be collected that was summarized through axial coding, making the information usable and measurable.

We chose for interviews as in both our experiences it was a good way of gathering information from individuals as opposed to using forms, for example.

### Sharing of knowledge

To make sure everyone in the group was up-to-date with the progress that was made, during every virtual meeting all project members shared what they have been researching the week prior. This kept every project member informed, and allowed for room for discussion.

Weekly communication was necessary to keep in touch, but also to keep all members updated.

### Axial coding

To make sure the information we gathered from interviews was valuable, it had to be transformed from raw text from the interviews to information that could be used to generate statistics. We’ve chosen for this method to give better insight into what’s deemed relevant by the stakeholders we interviewed.

## Accountability results

In this subchapter, we will reflect on the results from aforementioned methods we used during the course of this project.

### Desk research

Especially in the beginning of the project, a lot of desk research proved to be helpful in making us understand the material we needed to study for this project. For further details on the literature we studied, we refer to the Literature list at the end of this report.

### Field research

As our field research primarily consisted of interviews, we can say that these went according to our expectations. We gathered a variety of stakeholders that provided us a broad spectrum of information. In the way we did the interviews in hindsight we will ask more direct questions and record everything on tape instead of transcribing ad-hoc.

### Sharing of knowledge

The weekly meetings were proven to be essential, mostly to keep the group on track and giving each other feedback.

### Axial coding

The axial coding turned out to be essential as well in making longwinded interviews simple and relevant for statistics.

# Subquestion I

*“What is a Honeyjar-system and what is its relevance in current times?”*

## Honeypot

A Honeypot can be either a computer, or a network of computers with the purpose of “luring” malicious data towards themselves by seeming vulnerable and/or human. The Honeypot allegory is based on the attraction of bees to a pot of honey.

However, building such a system is complex since Malware can detect whether a system is really operated by a human, or is in fact a dummy computer functioning as a Honeypot. Therefore it is imperative that a Honeypot-system seems as ‘’human’’ as possible, i.e. emulating human behavior.

A containment system can be used to control traffic within the Honeypot to prevent and guard against Malware spreading throughout the network.

A Honeypot in its basic state can be seen as a lightning rod, meant to divert malicious data away from, say, a server. However, by installing additional modules into the Honeypot-system, it can be turned into a tool for data analysis. Additionally, there’s a variety of types of Honeypots:

1. **Production Honeypots**

Mostly used by corporations, these Honeypots improve the overall security and are frequently deployed in a production network.

1. **Research Honeypots**

Gather information about Black Hat[[1]](#footnote-1) attacks.

1. **Pure Honeypots**

Monitors all activities of the attacker using a bug tap.

1. **High-interaction Honeypots**

Imitates various activities using multiple Virtual Machines – should one VM be compromised, the VM can quickly be recuperated.

1. **Low-interaction Honeypots**

Imitates only the most frequently requested services by attackers.

### Purposes

As derived from the list in the previous paragraph, there are multiple purposes for a Honeyjar. Summarizing, the purpose of a Honeyjar is to analyse information by luring malicious data using the emulation of a system. Additionally, a popular usage of a Honeypot is to extract vital information from the hacker attacking the Honeypot, which can result in evidence against the hacker, which in turn can be used in court.

### Malware

Currently there are several kinds of Malware, i.e. malicious software that has the intent to obtain unauthorized data/cause damage.

1. **Phishing**

Malware that poses as something legitimate. For example, a popular phishing method is for a hacker to emulate an e-mail sent by a bank requiring the victim to enter his credentials, thus giving the hacker access to the victim’s bank account.

1. **Spyware**

Also known as Adware. Follows the victim’s movement on the internet, by sending the gathered information back to the hackers’ computer, which in turn generates spam for the victim. This is a particularly tedious Malware to remove as it tends to nest itself in the system registry.

1. **Keyloggers**

A Keylogger, as the name implies, logs all the keystrokes a victim makes. This may seem harmless, until the victim enters sensitive information like passwords for social media, banking, and e-mail.

## Current state

### Advantages

### High- interaction/Low-interaction

This entails the complexity of the Honeypot. High-interaction encourage the hacker to waste a lot of time messing around in a fake environment, all the while leaving behind valuable information about the hacker. Low-interaction Honeypots collect only the basic info, however setting up a low-interaction Honeypot costs less resources.

### Insights

Using a Honeypot gives the organization insight in who is attacking, and the methods that are being used. This also is a good indicator of what security measures should be installed on the real systems to prevent them from being attacked in a similar fashion.

### Frustration

Once a hacker discovers he’s operating in a Honeypot, he will most likely be frustrated and abandon the session. The other possibility will be elaborated on in the paragraph “Weaknesses”.

### Weaknesses

### SQL

In current times, a popular way of attacking a Honeypot is by using SQL Injections. SQL Injections are a code injection technique that may have fatal results for SQL-based databases. This happens via the placement of malicious code in SQL Statements via web page input, for example by entering SQL statements in a username/password prompt.

### Honeypot Detection Systems

Another weakness that naturally evolved at the hands of increasing popularity of Honeypots, are Honeypot detection systems. Such a system analyses specific characteristics of existing Honeypots, to identify other Honeypots, alerting the hacker that they are in fact not hacking a ‘’real’’ system.

### Legal repercussions

Yet another weakness of sorts is that a Honeypot isn’t simply a question of installing it and letting the software do its job automatically, like for example a firewall or antivirus. With a Honeypot, the person responsible has to know what he’s doing, i.e. have expertise on the workings of a Honeypot and know how to handle different situations.

Once a hacker is trapped in a Honeypot, the administrator has to be able to contain the hacker and prevent him from, say, retaliating once the hacker finds out he’s in a Honeypot. Retaliation can be in the form of the hacker jumping off to another point in the network, and causing damage there. The administrator or company using the Honeypot could be held liable for damages done to other networks through their own.

## Social consequences

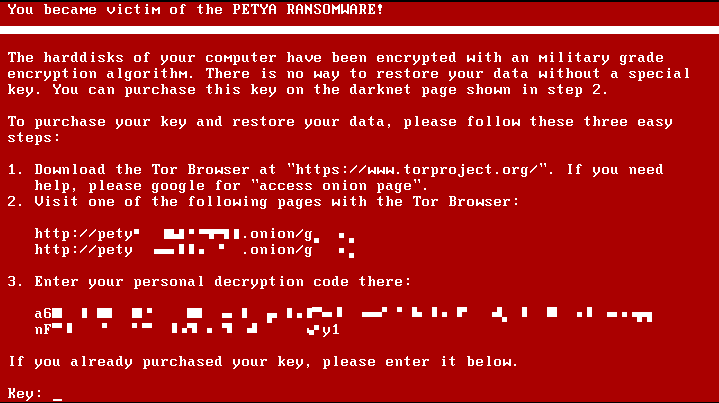
One perfect and very recent example of Malware causing major problems in recent history, is the Malware called “Petya”. Petya was Ransomware that locked down the victim’s computer, demanding a sum of money to be transmitted to the Bitcoin-address of the hacker. Due to this Malware spreading around the world, Ukrainian nuclear power plants, banks, and metro systems were affected, as well as one of the worlds’ largest shipping companies, Maersk, who is estimated to have lost between two hundred to three hundred million dollars in revenue due to this Malware-infection.

Figure 1: Petya victim

## Technical consequences

Ransomware does not only target operating systems like Windows, but also Android phones. With the rising sales of Android devices, this problem only seems to be growing bigger.



Figure 2: Android/-FakeAV.E victim

### Pornhub

One example of this is the Ransomware called Android/-FakeAV.E. This is Ransomware pretending to be an app for the adult video website Pornhub. When the user opens this app, the user is prompted an error message as shown in the figure. After this error message, the user is prompted to run a fake Avast Antivirus scan, and when this scan is done the phone is locked, with a message saying “device is in danger and is now blocked for security reasons”, prompting the user to buy the Avast “Pro” version for the price of a hundred dollars.

### Adobe Flash Player

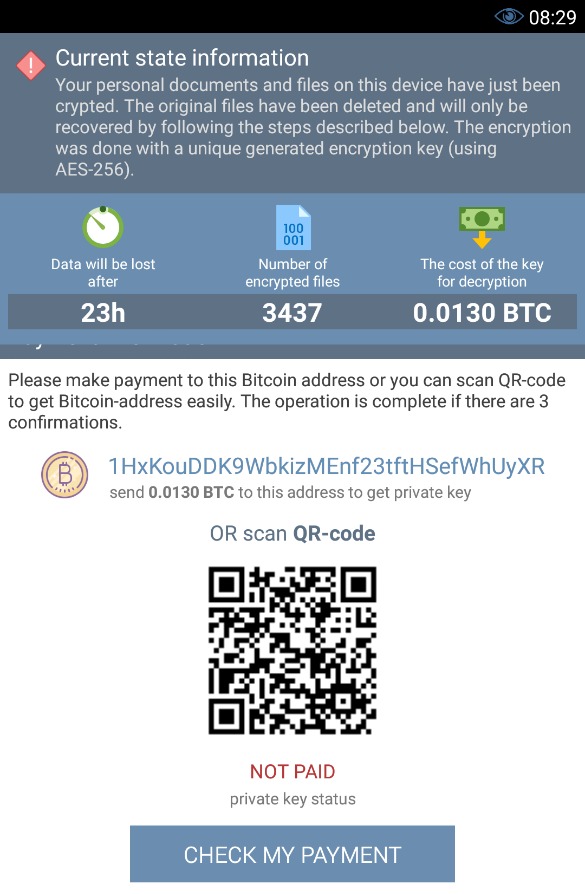
Another example of a very annoying type of Ransomware is DoubleLocker, discovered by Slowakian security company ESET. Doublelocker pretends to be an update for Adobe Flash Player, a technology that allows websites to play video content. Through hacked websites, Doublelocker sends out notifications to its victims saying they should update their Adobe Flash Player.

Figure 3: DoubleLocker victim

Of course, once the user downloads this file, the victim is downloading Ransomware. The file in question is an \*.apk-file, which is an executable file that has not been approved by the Google Play Store.

After installation, the Ransomware is activated as soon as the user hits the “Home” key of his phone, showing a prompt on the locked home screen as shown in the figure. Having locked the home screen, Doublelocker also changed the PIN-code of the phone. Once again, just like Petya, the money can be transferred through Bitcoins.

Once a phone is hit by Doublelocker, the only way to get the phone back to normal, is to reset it to factory mode, thus losing all files on the phone if no backup is made.

## Honeyjar

In this project we will use a Honeypot inside a Honeyjar-system. What this means is that, besides the core Honeypot functionality, the end product will have several features that make the system unique, namely Machine Learning.

### Machine Learning

Machine Learning is a broad term centered around artificial intelligence. In essence, a Machine Learning-algorithm initially gets fed a base amount of data. In turn, additional data gets fed into the Machine Learning algorithm, making the algorithm more accurate with each iteration. Thus, the machine is “learning”.

To further illustrate this process, Machine Learning is essential to retail advertising in current times. Whenever a person is browsing the internet, he is bound to receive advertisements that are within his field of interest, making him more tempted to click on the advertisement and buy said product. If it was a random product, it can be said that the person would be statistically less prone to click on the advertisement. This is happening because the Machine Learning-algorithm has analysed the person’s online behaviour, generating more specific information about the user with each page he visits, thus giving him ads relevant to his interests in the end. This example shows one of the benefits that Machine Learning can create: generating more revenue.

## Implementation

What the project group envisions for the Honeyjar-system is a core Honeypot, with added functionality of Machine Learning. This means that whenever Malware enters the Honeypot, the Machine Learning algorithm will start analyzing the data that has been edited by the Malware (i.e. infected), thus improving the recognition of Malware over the course of time.

## Android

Android officially launched in the USA in October of 2008 with the T-Mobile G1 handset phone. Everywhere else in the world the phone was called the HTC Dream. As seen in the picture below it had a keyboard built into the phone. It was revolutionary at the time and caused the start of what we would know today as the Android operating system.

The Android operating system is open source and developed by the Open Handset Alliance (OHA). The OHA is an alliance between manufacturers and developers and was established on November 5th 2007, (Industry Leaders Announce Open Platform for Mobile Devices, 2007) the first Android phone would be released a year later. The goal of the alliance is to create an open source standard for smartphones. One of the outcomes of this alliance is the Android OS. The OHA has been founded by Google and they are also responsible for the Android OS.

Figure 4: The HTC Dream

(BHATT, 2009)

Furthermore, the Android system can be chopped up into four parts:

1. The base operating system
2. An application middleware
3. A Java SDK
4. A collection of system applications

These four parts are necessary for a basic Android system.

### Appeal

One of the appeals Android had was the seamless Google integration. Things like Gmail, Calendar and Contacts were all built into the phone. This made setting up and using the phone much easier. The only thing necessary was a Google account, after that the phone would synchronize with the Google servers.

### Security risks

However, even with the release of the HTC Dream there were still holes in the software, security breaches that could be exploited by anyone. The HTC Dream had a security flaw that was discovered not long after the release of the phone. Anyone was able to take control via outside software (Hildenbrand, 2017).

So even the first phone with Android brought along security issues that could be exploited.

The exploit with the HTC Dream was that anything being typed by the keyboard was interpreted as a command in a privileged shell. This would mean that if something (Malware) took over your phone and sent commands using the keyboard it could do anything it wanted. Even get to your personal data.

This exploit was quickly fixed by Google, however a signed image of the old software leaked making it available to anyone (Sridhar, 2015).

Since then smartphones have taken a root into society. Everyone uses smartphones in their everyday life. The person reading this probably has one lying on the table or in their pockets. These devices are used for everyday tasks such as:

* Browsing the internet
* Gaming
* Vehicle guidance or other GPS applicatiosn
* Voice calls
* Text messages

With all these tasks, sensitive data may be sent through them, as well as simply stored on the device. This data could then be to malicious. In a study conducted in 2012 titled *“Measuring User confidence in Smartphones Security and Privacy*”, a group of researchers questioned sixty smartphone users about security on their smartphone devices.

The researchers first asked the user’s willingness to perform certain tasks on their devices. This was asked to test the hypothesis that users avoid using their phones for privacy and security concerns.

Secondly the researchers analyzed why and how users select applications. For instance, how do they trust certain applications with their personal data?

### Computers

There of course are certain drawbacks to using a smartphone when compared to a PC. On the other hand a smartphone is small and easily accessible. If someone quickly wants to look something up they usually grab their smartphone. If someone is lost they can quickly use the GPS system on their phone to get to where they need to go. However, multiple studies have shown that people finish most tasks on their computers (Chin, Felt, Sekar, & Wagner, 2012). So, they quickly start something on their smartphone and then later finish the task on a computer. Researchers believe this is because of a couple of reasons:

1. Screen size
2. Network performance
3. Typing difficulties
4. Security reasons

Now the last point is perhaps the most interesting. Why do people not trust their phone’s security? In another study titled *“No smartphone is an island: the impact of places, situations, and other devices on smart phone use”* researchers revealed that users who perform shopping activities on their phone usually wait with payment until they are on a computer. The interviewed people blamed this on the fact that they didn’t think their phone browser was able to handle the requests.

### Statistics

Now that its clear security isn’t great on Android let’s get into numbers. How many phones are infected? And do people know this?

Nokia releases a report every year in which they asses the security on different platforms. One of the tested platform is Android. In the following table is shown which Malware is the most common, it shows the top 20 of most common Malware on Android.

|  |  |  |  |
| --- | --- | --- | --- |
| Name | Threat | % | Previous |
| Android.Adware.Uapush.A | Moderate | 14.59 | 5 |
| Android.RansomWare.Jisut.BT | High | 9.93 | 34 |
| Android.Trojan.Axent.EH | High | 4.89 | New |
| Android.BankingTrojan.Marcher.A | High | 4.6 | 6 |
| Android.Trojan.Sivu.C | High | 3.96 | 16 |
| Android.Trojan.HiddenApp | High | 3.57 | 60 |
| Android.Trojan.Clicker.HA | High | 3.36 | New |
| Android.Trackware.AndrClicker.D | Moderate | 3.06 | New |
| Android.Backdoor.Godless | High | 2.82 | 17 |
| Android.Trojan.Rootnik.i | High | 2.6 | New |
| Android.Trojan.Xiny.19.origin | High | 2.54 | 14 |
| Indep.MobileSpyware.mSpy | High | 2.42 | 74 |
| Android.InfoStealer.Adups | High | 2.2 | 84 |
| Android.MobileSpyware.SmsTracker | High | 2.12 | 11 |
| Android.Trojan.Xgen.FH | High | 2.11 | New |
| Android.Trojan.Rootnik.q | High | 2.11 | New |
| Android.Trojan.Qysly.Q | High | 2.02 | 12 |
| Android.BankingTrojan.Acecard.m | High | 1.93 | 13 |
| Android.Backdoor.Xgen.CD | High | 1.73 | New |
| Android.Trojan.Leech.d | High | 1.47 | 43 |

Table 2: Most common Android Malware

It has to be noted that this changes constantly. In 2016 Nokia released a completely different table for their report.

Android is responsible for 68% of the malicious activity on mobile platforms. Despite the fact that Google is trying to prevent this with countermeasures, there still are a lot of third party app markets where users download their apps from.

China for example is responsible for a baffling 96% of third party market usage, only 4% in China use the legitimate App Store. Third party App Stores have become the norm in China and the official Google Play Store is in the 10th(!) place. One of the reasons for this is the fact that Google doesn’t support the purchase of paid apps in China. There are a more countries that are unable to purchase apps from the Google Play Store. A couple of examples are Sudan, Myanmar, Iran and Cuba (Google, 2018).

### Third-party stores

The reason that third-party stores are so popular is quite simple: Google’s support is lacking. To take one example: it is now possible in every countries to buy apps, however selling apps is a whole different story. From all the countries that a consumer is able to buy their apps from, 45,39% of those countries are unable to sell their apps and put them on the Google Play Store. This leads to developers selling their apps on different platforms (Google, Supported locations for developer & merchant registration, 2018).

Another reason that Android is the most infected platform is manufacturers. There is a lot of fragmentation on the Android markets. There a thousands of different kind of Android phones from different manufacturers, and Google isn’t responsible for updating each and everyone’s device. Google develops the Android OS and distributes it to the different manufactures, but at the end of the day it’s up to them if they are going to push to update to their phones.

Figure 5: Mobile infections per OS

What this means is for example: a Samsung phone has a security issue because of a bug in the latest version of Android. Google proceeds to fix the bug in Android and makes it accessible for everyone. It is then up to Samsung to push the new version to their phones. If they don’t do this, then this bug will keep affecting their phones. They are not forced to update their phones, it’s their choice.

## Conclusion

*“What is a Honeyjar system and what is its relevance in current times?”*

The Honeyjar-project group intends to focus on Ransomware on Android phones. The choice for this specification was made because of:

1. Increasing cases of Ransomware-problematics around the world, for example Petya
2. Increasing use of Android phones, especially in professional environments

Concluding, we can define a Honeypot as tool to gather information about Malware, as well as the hackers deploying the Malware. In the case of the Honeyjar-system however, this Honeypot will be enhanced by adding the functionality of Machine Learning. This way once Malware enters the system, it gets recognized, defined, and added to the database.

As of now there is much unknown about the combination of Ransomware on Android-phones, as it’s a relatively “new” phenomenon. This makes it all the more interesting for the Honeyjar project.

# Subquestion II

*“Why and how should our targeted audience use a Honeyjar-system?”*

## Targeted audience

First of all, a targeted audience has to be chosen to answer this question, with the goal to answer who would benefit the most from Android-based security. At this point, there can be made cases for two candidates; Business and Personal Use.

### Personal Use

It is no secret that Android phones are extremely popular among consumers. Their price range is significantly broader than their Apple counterpart, ranging from $100 to +$1000. The minimal price for an Apple Phone sets the user back $400, ranging up to models of $1200, not considering second hand purchases.

Figure 6: Phone OS Distribution in Europe

As shown in the graph, it’s obvious that the Android OS is the most popular in Europe.

The criteria that have been used to create the figure are:

1. Only operating systems that are on mobile phones have been taken into account, so tablets have not been accounted for
2. The region is Europe
3. The sample size is from March 2017 – March 2018 (Statcounter, 2018)

The Google Play Store, Android’s in-house app store, has a vast amount of apps, more specifically reaching 4,000,000 at the time of writing.

The Apple App Store on the other hand has ”only” 2,100,000 apps. The crucial thing to note here though, is that anyone can submit their home-made app to the Google Play Store with ease, whereas the Apple App store is strictly monitored, and only allows few apps that strictly abide Apple’s policies.

This fact already gives away a problem, what happens if an App gets through the Google Play Store, that for example logs all keystrokes without the consumer knowing it? This is cause for concern, as important data can be retrieved from the consumer.

One example of Ransomware on Android phones is LeakerLocker. LeakerLocker was distributed through the Google Play Store, hidden away in an app called “Wallpapers Blur HD”. Once the app was downloaded and the Ransomware kicked in, the user was warned that if he didn’t pay the $50 fee, all his private data would be sent to his contacts.

A Honeyjar-system for personal use could partake in a network of other users, gathering data about the data on their phones, helping to identify and perhaps prevent Malware. However, for the average user, such a system might be too complex and expensive for the small risk it might prevent. Consumers these days store their sensitive data on their phones anyhow, in general there is not enough fear for losing sensitive data instilled within the average consumer. On that note, for financial gain, hackers and Malware might have more results from hacking businesses and governments.

### Business

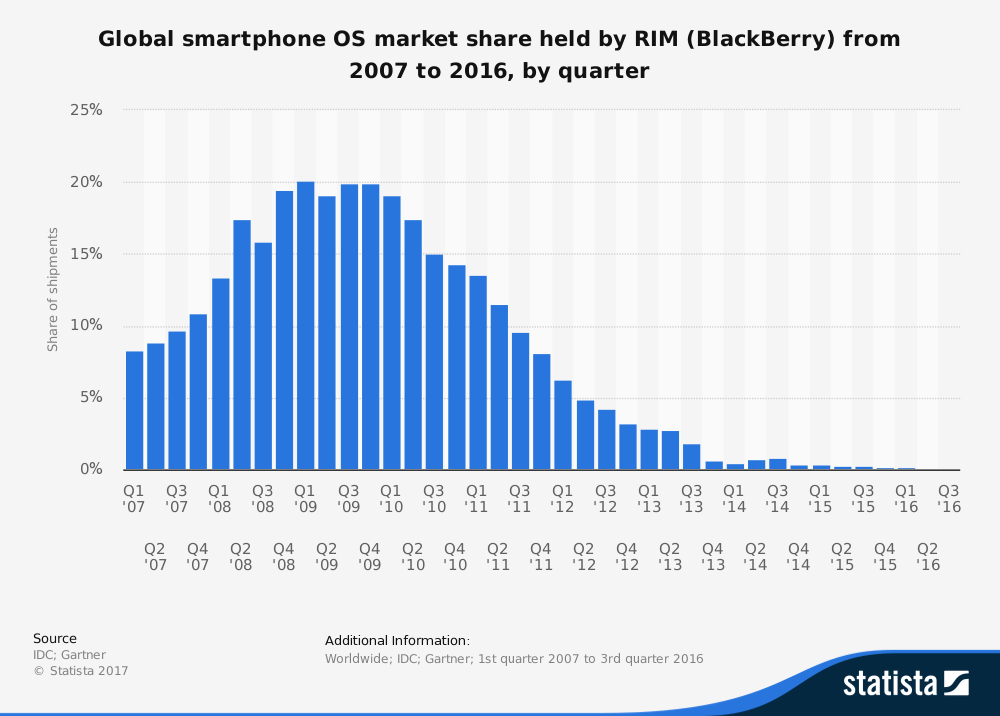
For a long time, whenever the words “business”and “smartphones” were uttered in the same sentence, the first brand to come to mind was Blackberry.

Figure 7: Global market share held by RIM

However, in a world where IT-improvements rise at cut-throat speed, it is essential to keep up. This, among other reasons such as the introduction of the Apple iPhone, caused the downfall of Blackberry. Soon enough, Apple, Samsung and OnePlus were vying for this piece of market share.

Research has recently been conducted in the Netherlands, pointing out how Android regained traction in the Business field, as seen in the graph below.

So even in businesses Android is the most popular OS. Not by a lot but it’s still a majority. The reason for this is most likely because of the fact that Android comes in a lot of different shapes and sizes.

Figure 8: Phone OS distribution in businesses

There are cheap phones which can be competitive versus the low-end devices, while on the other hand there are high-end phones who can compete with the likes of Apple.

In order to verify this graph the project group contacted the website owner and asked them how they did it. They responded as follows:

(Profile, 2017)

*“Doing research like this requires users to fill in answers which they might not know. They might give contradictory answers which in turn invalidates the research. We tried to accommodate for this in our answers.”*

When the project group followed up with a question whether there have been done other researches on the same problem the owner responded: “

*“Not that I’m aware of, the only people that would know this are companies such as Vodafone and T-mobile, the ones who actually provide the businesses with phones.”*

So the percentages show that Android is more popular among individuals then businesses. But what if someone, be it a business or not, is attacked. How much damage will there be done on average?

To answer this question, it is needed to research an attack in recent history.

### Case study: Maersk

A company called Maersk had been hit by NotPetya. Maersk is a company that has segments in transport, logistics and energy sectors. They are the largest container ship and supply vessel operator in the world since 1996. Maersk is a Danish business which is a collection of different companies which together form Maersk. (Maersk Q2 2017 report, 2017)

On June 27th 2016, Maersk fell victim to the NotPetya Ransomware attack. In a report from Q2 of 2017 Maersk released a report in which they stated the Ransomware entered through software which was used to fill in taxes in Ukraine. The software is called M.E.Doc, it was later revealed that this software hadn’t received and update since 2013 (Cimpanu, 2017). The attack made the company unable to use applications as well as making data unavailable to use. Basically everything from the company went down.

When Maersk found out they were infected they took immediate action to contain the Malware.

Infected networks were closed down to contain the attack. What ended up happening was that only the container-related business of Maersk was affected. This is 33% of Maersk’s business.

It took a total of ten days to completely recover from the attack. In those ten days Maersk had some of the following issues:

1. Formerly automatic tasks has to be done manually.
2. Shipments had to be manually identified since IT systems were down and employees didn’t know what was in the containers.
3. Maersk was unable to receive new order since the portal was down (McCreanor, 2017).

Maersk had to reinstall their entire infrastructure. This consisted of:

1. 45000 new PC’s,
2. 4000, new servers
3. 2500 applications

The estimated loss is around 250 – 300 million euros.

During the time of the reinstallation of the infrastructure, business had to continue. Maersk managed to do this with 80% of their normal workload, so they had a 20% drop in total volume which is impressive when you have to process 20.000 ship containers every fifteen minutes with little to no IT. Once the email services went down, WhatsApp had to be used in order to communicate with each other.

In a cybersecurity conference after the attack, Chairman Jim Hagemann Snabe called the attack a *“very expensive wake-up call”* and followed up on saying the company has learned three primary things from the attack:

1. They want their cybersecurity to become a competitive advantage.
2. Keeping an open dialogue to the world so that cybersecurity is something that is accessible to everyone and an attack can happen to any company.
3. 90% of all the orders from Maersk are digital and come through the internet, the next step is that everything becomes digital (Møller-Maersk, 2018).

The reason Maersk has been taken as an example is because it was one of the largest companies to be hit by the attack and it managed to handle the attack fairly well. There were different companies hit by the attack such as FedEx and Merck. They had similar losses in revenue but the aftermath of the attack was worse for them. (Forrest, 2017).

## Choice of audience

Suffice to say that following up on the aforementioned cases, a business is most sensitive to the intrusion of Malware through Android phones. This is because there is an increased chance of Malware since there are a lot of Android phones used by employees for the same business, as well as the possibility of leaking much more valuable data.

## Relevant factors

Now that the target audience is chosen, it must be made clear what the most important features are in such a Honeyjar-system. As a result of the Desk-and Field Research that was done for this project, the following has been established[[2]](#footnote-2). In other words, these are the Must-haves for this project:

### GUI

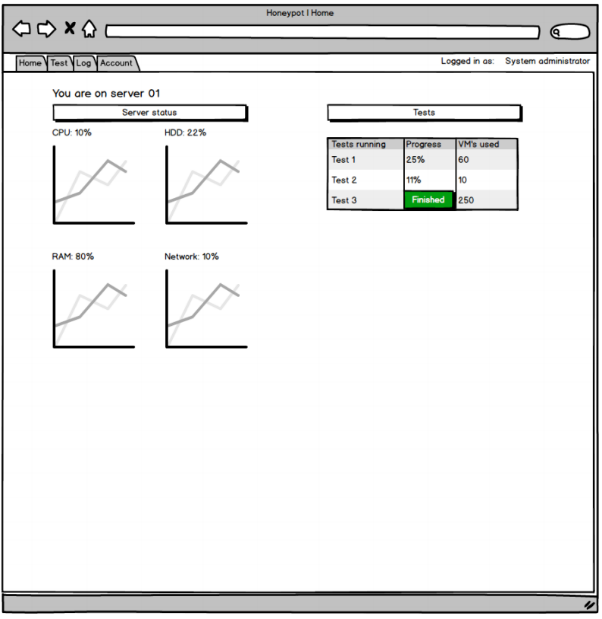
One thing that became clear early on during development, is that the envisioned model should be made into a Web UI. The reasoning behind this is that a Web UI is available from anywhere as long as there’s an internet connection, usable on computers as well as handheld devices. Another advantage is that a Web UI is easier to create and maintain than a desktop application (for example, a Desktop App would have to be developed for several OS’es).

Figure 9: Beta version of the Honeyjar GUI

The GUI on the left is one of the early designs that has been made with the entire group in Barcelona.

This design was made with the idea in mind that it could be finished by the end of the project. There is an overview for the server with the following:

1. CPU Load
2. HDD Load
3. RAM Load
4. Network Load

Furthermore the user, in this version, is able to inject the system with a piece of code or file.

The reason the project group decided on showing this particular information is quite simple.

The product that could be delivered only had to be able to take a file, inject it into the system and then monitor what happens. The monitoring can be done with the graphs that have been described earlier. If twenty virtual machines are running the same type of injected program and one of the virtual machines is showing a lot of HDD-activity then something suspicious is going on. With the graphs, it will be possible to spot such an anomaly, then the user will be able to take a look at that specific virtual machine and see what’s going on.

Since the previously mentioned GUI will not be the end product, the Saxion students made three designs that could be the designs for potential end products. These three will now be showcased.

### The free user experience

Since the end product will be functioning in conjunction with Machine Learning, it’s vital to obtain a viable dataset from which the Machine Learning will be able to learn, expand and predict new malicious activity.

This however will require a substantial amount of data. In order to obtain this data the following plan has been thought out:

Hobbyists, network enthusiast, universities and anyone else that is interested will be able to download a free version of the system. This system will be comprised of two parts:

1. Part one will allow users to put in data, let the Honeypot system algorithm take a look at it and then tell the users whether the file is malicious or not (See Appendix 14.1.1).
2. Part two will allow the users to do the Machine Learning themselves.

Part one is self-explanatory, part two however will be elaborated on right now.

Part two will be CMD (command line based). What users will do is inject a file into our system through the command line, they will then be able to see the file and all of its properties, i.e. packet length, hash values etc. The free users will then be able to connect certain parts of the file together and say: If this part of the file has value X and this part of the file has value Y: Then it is malicious. Another example is weight and height, these two variables say something about someone. They are always connected. The taller you are, the more you (most likely) will weigh and vice versa. Keyword here is most likely since you can never be 100% sure.

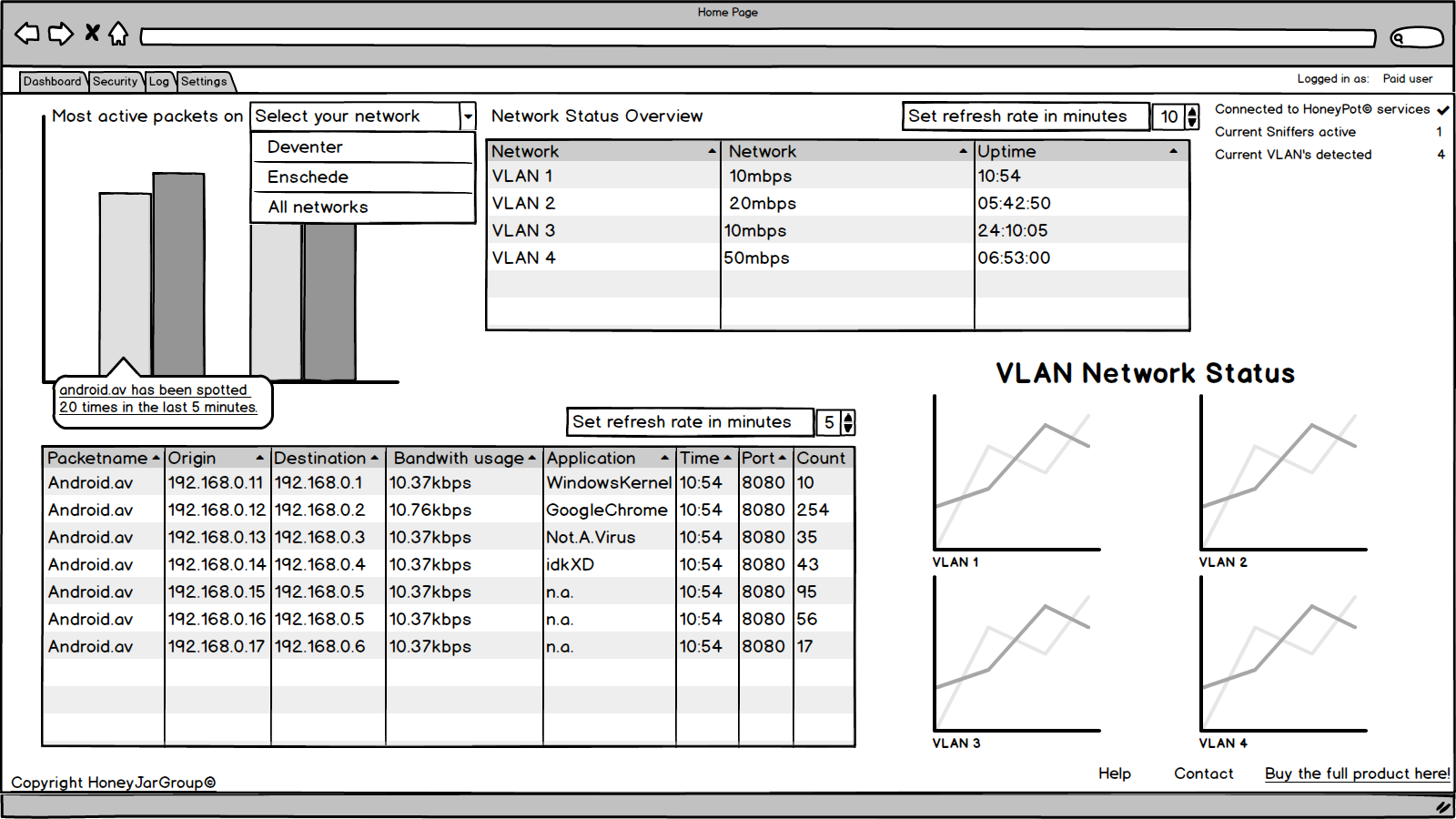
Once the free users have determined the connections, the Honeypot algorithm will see if it is usable and upholds the threshold. This threshold will be put in place to verify whether or not the suggested relation is legit or not. If the threshold is met, the relation will be saved and added to the algorithm. This way users contribute to bettering the algorithm.

A BPMN scheme has been made and placed in the Appendix to showcase how this solution will work.

### The Network Sniffer Solution

The mid-priced product will be made possible with a network sniffer. A network sniffer is a device that gets placed in-between the customer’s network and the ISP. This way the network sniffer will be able to catch all the traffic coming in and out of the network. In the interviews it was said that a GUI will be mandatory, as long as it’s logical and not randomly put together.

The solution is the following GUI.



A quick rundown from the GUI shown above:

Starting from the top left corner of the screen. The users will be able to select their network. Once a network has been selected the GUI will change according to the network that has been chosen.

Then a graph is shown with the most active packets on the selected network. The user will be able to mouse over the different beams which will then show more details about the beam.

Going down there is a table that is showing what packets have gone through the network since the last refresh. The refresh rate will be able to be set on the top right corner of the table.   
The table shows the following data:

1. **Packetname**

This will show the name of the packet.

1. **Origin**

The origins will show from what IP Adress the packet is coming from.

1. **Destination**

The destination will show the destination of the packet. This is so that the user knows what client the packet was send to.

1. **Bandwidth** **usage**

This will show the bandwidth that has been used by the packet. If a packet is using more bandwidth than it should than that might be cause of concern

1. **Application**

This will show the application that requested the packet to be send over the network. This is interesting because the user can look directly into the application if something is out of order.

1. **Time**

The time of the logging. This will allow precision pinpointing and troubleshooting. Maybe the same malicious activity is happening on the same time each day.

1. **Port**

The port that is being used by the packet. If malicious activity is coming through only one specific port, it might be a good idea to shut down the port.

1. **Count**

This has been set in place to see how many times the same packet has been through the network. If there is a lot of the same packet coming through but the user is unfamiliar with it, something might be wrong.

With this table a network administrator will be able to closely monitor their network and see if there is any anomalies going on.

The next part of the GUI shows an overview of the network status. This is displayed in a table with the following columns:

1. **Network**

If there is only one network selected this column will show the VLAN’s on the network, otherwise it’ll show each individual network.

This will make it possible to look at each individual part of the network and give a summary of the activities.

1. **Bandwidth usage**

The same as in the previous explanation of bandwidth usage, this will show how much bandwidth has been used. In this case however it’s over a whole network.

1. **Uptime**

This is the uptime of a specific network. The longer a network is up the more reason a user might have to reboot it if there are issues.

In the diagram on the bottom right there is a visualization of the previously mentioned table.

### The Shadow Honeypot Solution

This is the ultimate end-product. This will require good hardware because the idea is that it will create a shadowed version of the actual network. Before anything gets on the network it will run through the shadow or simulated version. If something happens on the shadowed version that should not happen, it will not be allowed on the network.

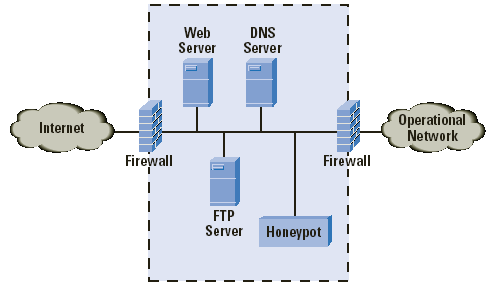


Figure 10: A Shadow Honeypot

(Pessotti, 2018)

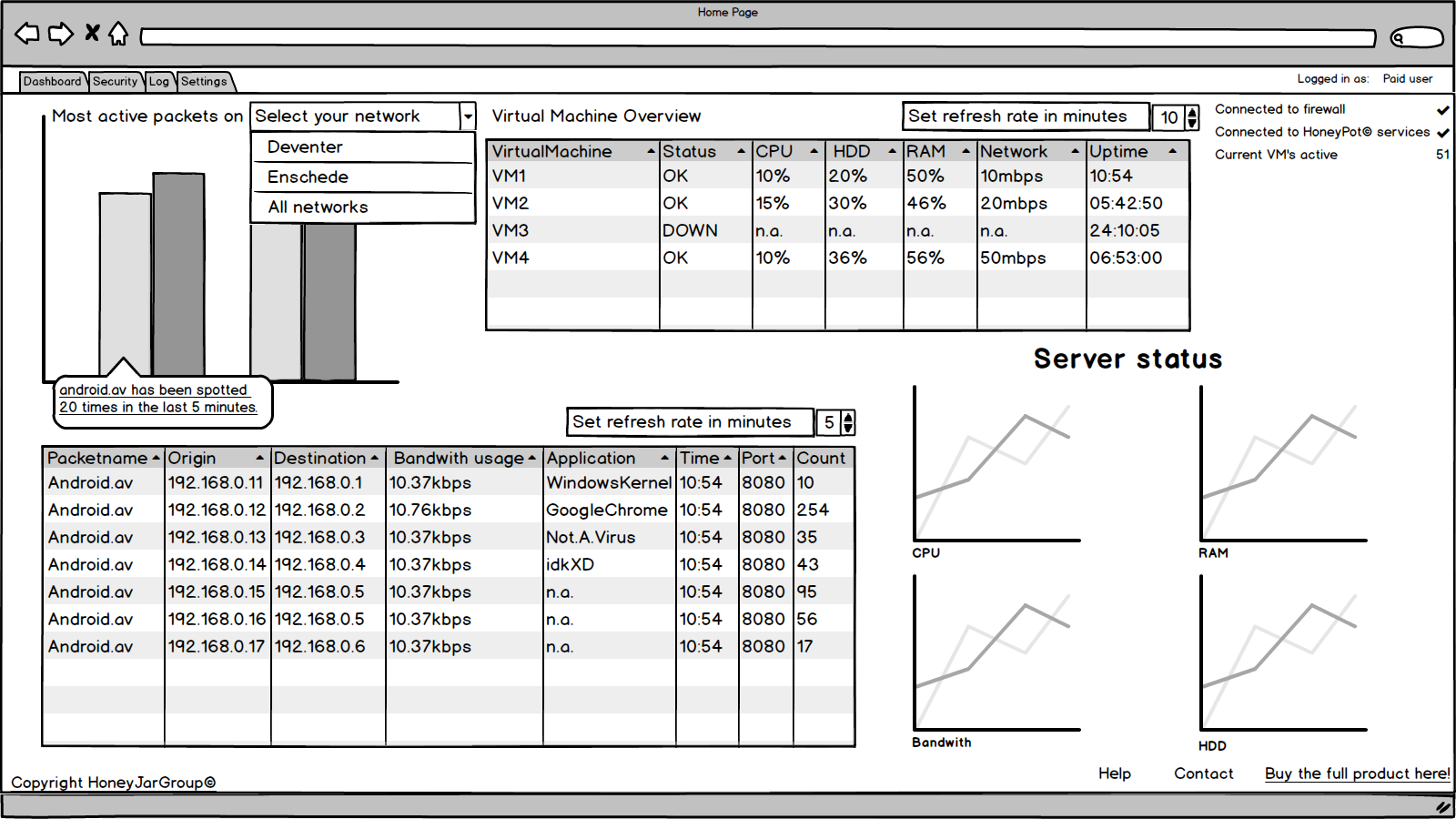


Figure 11: Overview per VM

This GUI looks very similar to the previous one, there are however a couple of key differences.

There now is a virtual machine overview and a server status overview.

Since the Shadow Honeypot will be using virtual machines to emulate the network an overview per virtual machine is mandatory. This will show whether or not a virtual machine is acting up, if it is a user might want to look into it.

The server status is there to make sure the physical server, that is running all the virtual machines, is not being overloaded. If it’s constantly at its peak performance it might be needed to upgrade the server’s hardware.

A BPMN Scheme has been made and placed in Appendix I to showcase how this system works.

## Functionalities

In this subchapter we will elaborate on the different functionalities that should be in the system.

### Virtual Machines

Virtual Machines are what makes this Honeyjar-system possible. These are systems that can emulate other systems on a server, so in the case of this project, Android phones. The Virtual Machines must be lifelike in the sense that a hacker wouldn’t be able to differentiate the Virtual Machine from an actual Android phone.

### Account policies

Aforementioned GUI should contain several different types of accounts, to enable different permissions and show different data relevant to the person that is browsing the GUI. For example, a CEO isn’t interested in watching the network usage of a Virtual Machine, whereas a System Administrator arguably is. It can be said that the CEO only wants to see simple indicators such as lost/gained revenue due to Malware attacks. For this reason, the system should have an account management system in place.

### Machine Learning

One of the core features of this project is the Machine Learning process. The main goal of this algorithm in the scope of this project is to show a percentual increase in the recognition of Malware. When Malware enters the Virtual Machine, it is considered “Unlabelled Data”, until the Machine Learning algorithm recognizes and sorts it as “Labelled Data”. This simply means that de algorithm was triggered, and recognized the data as malicious, giving it a label.

### Local Hardware

Another key factor to make this system work, is to install local hardware, i.e. a network scanner on the location of the business in question, to send all relevant data to the Honeyjar Project. The reasoning behind installing this hardware is that it won’t interfere with the company’s bandwith, as there is a constant stream of data being sent. If there weren’t any hardware, it would be a serious strain on the company’s bandwidth.

### Law Compliancy

As Internet security and privacy becomes a recurring subject in the news these days, the Honeyjar group must be aware of the rules that apply to the gathering of user data, and be mindful of not breaking any rules.

## Business Model

The Honeyjar will generate labelled data through the local hardware located at the business, as well as individuals who use the open source Honeypot to contribute data. This open source Honeypot is free of charge, on the condition that the individual shares the data with the Honeyjar Project. All this labeled data, from businesses and individuals combined, will be sent to the main Honeyjar server, allowing thet Machine Learning algorithm to improve.

Access to this database can be granted in two ways:

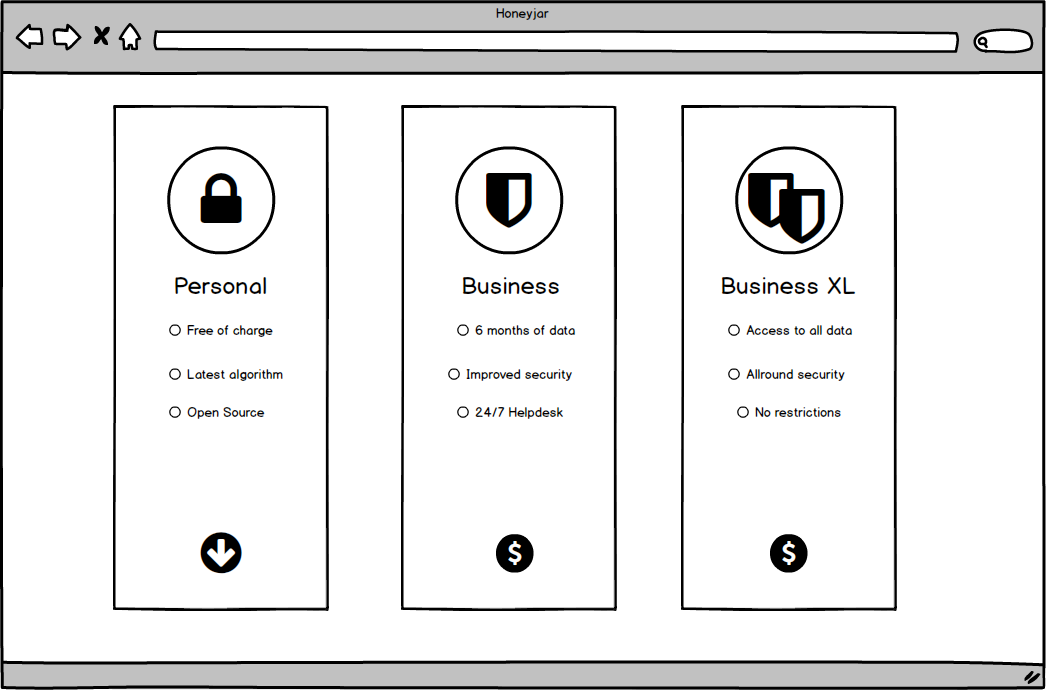
1: For Personal Use  
With this subscription aimed at individuals, the latest week of the results of the Machine Learning algorithm are accessible, granting up-to-date recognition of Malware, however lacking the base data.

Figure 12: Different plans for different customers

2: For Businesses  
The main source of income should derive from businesses. They are able to buy a standard dataset, i.e. the resultst of the last 6 months of the Machine Learning algorithm, or for full security, a subscription where access to the entire dataset from the Machine Learning is granted.

This results in 3 ways to gain access to the Honeyjar-system. In other words, a Freemium model: The base functionalities of the application are available for free, however when additional functionalities are required, the user is required to pay.

## Blueprint

To further illustrate the workings of this Honeyjar-system, the following illustration has been made.



Figure 13: Honeyjar designed in BPMN

First off in the process the client wants to buy one of the three Honeyjar systems. Payment follows. Once this has been completed, a Honeyjar mechanic will go on-site to install a local network scanner as well as a Honeypot. These will send data every 5 minutes to the Honeyjar server, automatically. The Machine Learning scan is executed, taking note of any Malware that has been detected. If this is the case, the attack is prevented. If there is no malicious activity, the system is on hold until there is.

## Conclusion

*Why and how should our targeted audience use a Honeyjar System?*

Concluding, it can for one be said that for the sake of securing their sensible data, businesses should implement a Honeyjar system. Secondly, by doing so, the companies will contribute to the Machine Learning Database – the more businesses that join, the better the system will be.

Businesses have several ways of implementing the Honeyjar system, 2 plans can be chosen, each with different attributes. Once a plan is chosen, enabling security for the business should be effortless, the only hindrance that should be experienced by the business in question is the one-time visit by the Honeyjar Mechanic, who will install the local server. Once that has happened, the company will enjoy the degree of security that is promised in the plan they bought.

Another contributing factor to the Machine Learning algorithm is the open source, free version of the Honeypot which enables hobbyists and security enthousiasts to contribute Malware data to the Machine Learning algorithm.

# Subquestion III

*“What are the results of the Honeyjar-system and what does this mean from a business perspective?”*

## Disclaimer

It is important to mention that the scope of the project has been revised many times during the course of the project. After some time it became clear that the Honeyjar System cannot be completed during the project, as the system has become quite complex on a technical level, and thus would take a substantial amount of time to realize – time the group does not have.

The good news however is that the Honeyjar system will be passed on to following project groups. This means that in the scope of a few years, the system will be realized by multiple different groups. The basic layout of the system made during this project, will serve as the blueprint for following projects.

## Prognosis

So as the end product of the project won’t physically delivered, to answer the subquestion “What are the results of the Honeyjar-system and what does this mean from a business perspective”, assumptions have to be made. Logically these assumptions have to be as accurate as possible, so to back these assumptions up, the interviews made during the field-research phase will be used.

## Encoded interviews

### Approach

During the field-research phase of the project, three different potential buyers of the Honeyjar system were interviewed. The stakeholders were:

1. Bram Wenting, a CEO at SST Software. SST software creates websolutions.
2. Wesley van Wieringen, Network Administrator at Tedas
3. Valentin Carela, product manager at Talaia Security.

These stakeholders were chosen particularly for their different roles. The Honeyjar group thinks this will give a broad overview of who finds what relevant in the Honeyjar system.

### Example

However, interviews alone do not provide information in a constructive or clear way. That’s why the Honeyjar group chose to use the Axial Encoding technique. This is a fairly straightforward technique that works as follows.

1. During the interview, a transcript of the conversation will be made.
2. The sentences deemed most relevant in this transcript will be highlighted.
3. These highlighted sentences will be assigned to a maximum of 3 subjects related to the sentence.
4. The final coding part is assigning one subject to the previously made subjects.

This looks something like this:

|  |  |  |
| --- | --- | --- |
| Sentence | Open coding | Axial coding |
| “I’d like to use earplugs, but they just look so silly.” | Intention. Appearance. | Appearance. |
| “I always wear earplugs when I go to a festival, that way my ears are protected.” | Protecting ears. | Protecting ears. |
| “I don’t want to be the only one wearing earplugs, I’ll make a fool of myself.” | What do others think? Behaviour of others. | Behaviour of others. |

Table 3: A simple example of coding

So now that the sentences are coded, they can be used in graphs for example, i.e. “What prevents consumers from buying earplugs?”. So to conclude this example, coding makes a seemingly uninteresting bulk of information clear and insightful.

## Interview context

Through coding, the project group generated the following subjects:

### Dashboard

The Dashboard subject entails what the stakeholder deems relevant when using the Honeyjar-system – in other words, what statistics should be shown, what should the layout look like, and what functionalities should be present in the dashboard.

### Ease of Use

Defines how accessible the Honeyjar should be for the users that have access to it.

### Staff Knowledge

For the business that wants to implement the Honeyjar its relevant to define what the level of IT-knowledge is from the staff – do they treat their smartphones carelessly or are they aware of their privacy and security?

### Profitability

Ways in which to make the Honeyjar profitable.

### Malware

This subject entails anything related to Malware, not only Ransomware.

### Law Compliance

The level of importance given to adhering to the law by using IT.

### Infrastructure

Defines the ways in which infrastructure, being physical or digital, is relevant to the stakeholder.

### Client Knowledge

The level of knowledge the client of the stakeholder is expected to have, and whether it’s necessary to “teach” the client.

## Coding results

Now that the subjects are clarified, the results can be presented in the *“Stakeholder Points of Interest-graph*”.

Figure 14: Stakeholder Points of Interest

## Results context

### Dashboard

Stakeholders had a reasonable amount of input regarding what should be apparent in the Dashboard. Requests that were made for instance was a Management Dashboard, a top 10 of incoming packets, and a means to track individuals based on packet data.

### Ease of Use

Not a reasonable amount of feedback was given here, most notably stakeholders implied that Google should treat their App store the same (strict) way Apple does to guarantee a level of safety and ease in their apps.

### Staff Knowledge

The stakeholders we interviewed put a lot of trust in their colleagues. The colleagues are aware of security and privacy on their phones, and in some cases even discourage using a phone for work. Where common sense amongst colleagues fail, the stakeholders keep their personnel updated on everything security-related through trusted sources like partners in security, and security forums/news.

### Profitability

Perhaps surprisingly enough the stakeholders didn’t delve deeply into Profitability. However one recurring advice was to make the Honeyjar subscription-based to make it appealing for customers.

### Malware

Malware was a big point of interest. Most notably, the Machine Learning algorithm was advised to make as accurate as possible, being one step ahead of Malware trends, and the advice to implement features in the system that are not dependent of other features, so if one feature fails, all other features should still work.

### Law Compliance

Did not get much attention, however one recurring theme was compliance to the recently applied AVG-laws, that forces websites to adhere to specific rules about the data they collect from visitors. Companies have to be law compliant these days, otherwise customers will choose for a competitor that is law compliant.

### Infrastructure

To the surprise of the researchers, Infrastructure turned out to be the biggest point of interest amongst stakeholders. Most notably it was advised to make the Honeyjar simple hardware-wise, easily deployable, with the least amount of on-site maintenance required. It’s also very relevant to keep all systems up-to-date.

### Client Knowledge

Finally, stakeholders would like to see functions in the Honeyjar that remind the clients of the customer about recent Malware-attacks, and educate them on what not to store on their phones – essentially educating them.

## Conclusion

*“What are the results of the Honeyjar-system and what does this mean from a business perspective?”*

The results show that the system needs to adhere to a specific set of factors:

1. **Infrastructure**

The system should be easily deployable for installment in businesses, and sparingly need adjustment or on-site maintenance.

1. **Malware**

The Machine Learning Algorithm should be as accurate as possible, and functions within the Honeyjar should be independent from each other to prevent chain infections of features.

1. **Dashboard**

Should be insightful yet simple, and contain at least a top 10 of incoming packets, and a means to track individuals based on packet data.

Concluding, the project group realized during the process of interviewing that the Honeyjar-system has a potential place in the market. This is evident in the positive feedback from the stakeholders as well as the amount of input the group received. This input can be found in the Coding sheet in the Appendixes chapter of this thesis.

# Conclusion

## Summary

During the writing of this document, the sub questions have provided the necessary preparation to answer this question. To refresh our memory, the subquestions will be summarized once more:

### “What is a Honeyjar-system and what is its relevance in current times?”

A Honeypot is a tool to gather information about Malware. In the case of the Honeyjar-system however, this Honeypot will be enhanced by adding the functionality of Machine Learning. This way once Malware enters the system, it gets recognized, defined, and added to the database. It’s relevant to analyze in our current time as Ransomware on Android phones is on the rise, without much security available to prevent it.

### “Why and how should our targeted audience use a Honeyjar-system?”

Businesses should implement a Honeyjar system. Secondly, by doing so, the companies will contribute to the Machine Learning Database – the more businesses that join, the better the system will be.

Businesses have several ways of implementing the Honeyjar system, 2 plans can be chosen, each with different attributes. Once a plan is chosen, enabling security for the business should be effortless, the only hindrance that should be experienced by the business in question is the one-time visit by the Honeyjar Mechanic, who will install the local server. Once that has happened, the company will enjoy the degree of security that is promised in the plan they bought.

### “What are the results of the Honeyjar-system and what does this mean from a business perspective?”

Our interviews have clearly determined that in this day and age where security and privacy are at stake, a Honeyjar-system as the group envisioned it is very much wanted.

## Main research question

The main research question is as follows:

*“How can a Honeyjar-system for Android be made relevant for businesses and turned into a profitable product?”*

To give a detailed answer to this question, the question will be dissected:

### Relevancy

The research made clear that in our current time the sales of Android phones is increasing, as well as the looming threat of Malware, most notably Ransomware, on mobile devices. Research also points out that this is fairly new ground, as businesses are starting to become aware of the importance of security in their infrastructure, but there is no solid solution aimed at Ransomware on Android phones at this point in time.

### Profitability

As stated, there is an increasing awareness of the importance of security among businesses, but with no clear solution in sight. This is where the Honeyjar shines, as it technically is a trendsetting product with no competitors that offer the same functionalities.

Concluding, research points out that our Honeyjar system can offer vital information for businesses in an untapped market.

# Recommendations

In the following paragraph recommendations will be made for any future projects as well as things we were not able to finish due to the lack of time or manpower.

First it should be stated that the product defined in the report is a very early sketch of a potential end product. A lot of things are in such an early phase that it may be possible that things should change along the way or will turn out impossible for future development.

That being said, the Saxion group recommends that at least the following will be done in a future product:

1. Look into the legal issues concerning collecting data.
   * Privacy is currently a hot topic and something that should be taken seriously. Because of time constraints it has been impossible to focus on this matter so it’s not covered in the project. Since this could be a project in itself this is something that should be looked into.
2. Further develop the product into an alpha version
   * In order the keep continuing the product it will be needed to start building toward an actual product. The Danish group has started this but it is still in a very early phase. This will need to be developed further.
3. The designs that have been made are still something that is in an alpha phase, this will need to be developed further.

If these three recommendations will be fulfilled, the project will be able to continue on in the future and create a product that will truly be something that the market will be able to use.

# Bibliography

BHATT, N. (2009, May 17). *HTC DREAM (GOOGLE ANDROID) SMART PHONE REVIEW*. Retrieved from bhatt.id: http://www.bhatt.id.au/blog/htc-dream-Google-android-smart-phone-review/

Chin, E., Felt, A., Sekar, V., & Wagner, D. (2012). *Measuring User Confidence in.* Washington.

Cimpanu, C. (2017, July 6). *M.E.Doc Software Was Backdoored 3 Times, Servers Left Without Updates Since 2013*. Retrieved from Bleeping Computer: https://www.bleepingcomputer.com/news/security/m-e-doc-software-was-backdoored-3-times-servers-left-without-updates-since-2013/

Forrest, C. (2017, October 30). *NotPetya ransomware outbreak cost Merck more than $300M per quarter*. Retrieved from techrepublic: https://www.techrepublic.com/article/notpetya-ransomware-outbreak-cost-merck-more-than-300m-per-quarter/

Google. (2018, June 21). *Supported locations for developer & merchant registration*. Retrieved from Support.Google.com: https://support.Google.com/Googleplay/android-developer/table/3539140?visit\_id=1-636585251809564044-2576914025&hl=en&rd=1

Google. (2018, June 21). *Supported locations for distribution to Google Play users*. Retrieved from Support.Google.com: https://support.Google.com/Googleplay/android-developer/table/3541286

Hildenbrand, J. (2017, July 29). *It's been 9 years and Android still has a bad reputation when it comes to security*. Retrieved from AndroidCentral: https://www.androidcentral.com/its-been-9-years-and-android-still-has-bad-reputation-when-it-comes-security

*Industry Leaders Announce Open Platform for Mobile Devices*. (2007, November 5). Retrieved from Open Handset Alliance: http://www.openhandsetalliance.com/press\_110507.html

*Maersk Q2 2017 report.* (2017, April). Retrieved from Mearsk: http://files.shareholder.com/downloads/ABEA-3GG91Y/5003725988x0x954063/4803DE66-B269-4730-8A92-C0D59BD7EE28/Presentation\_Q2\_2017.pdf

McCreanor, N. (2017, July 3). *Danish Shipping Giant Maersk Hit by NotPetya Ransomware*. Retrieved from Itgovernance: https://www.itgovernance.eu/blog/en/danish-shipping-giant-maersk-hit-by-notpetya-ransomware

Møller-Maersk, A. (2018, January 24). Securing a Common Future in Cyberspace. (A. D. Cohn, Interviewer)

Profile, C. (2017, January). *APPLE AND SAMSUNG MOST POPULAR BUSINESS SMARTPHONES IN THE NETHERLANDS*. Retrieved from Computer Profile: https://www.computerprofile.com/analytics-papers/apple-and-samsung-most-popular-business-smartphones-in-the-netherlands/

Sridhar, G. S. (2015). *Android (a Deep Knowledge).* March: 25.

Statcounter. (2018, May 1). *Operating System Market Share Worldwide - May 2018*. Retrieved from Statcounter: http://gs.statcounter.com/os-market-share

CGP Grey. (2017). How Machines Learn [Video file]. Retrieved from https://www.youtube.com/watch?v=R9OHn5ZF4Uo

Dong, Y. (2015). *Android Malware Detection Tool using Dynamic Taint Analysis*. Retrieved from http://library.iugaza.edu.ps/thesis/116141.pdf

Enck, W., Octeau, D., McDaniel, P., & Chauduri, S. (n.d.). *A Study of Android Application Security*. Retrieved from https://www.cs.vu.nl/~herbertb/papers/pa\_acsac10.pdf

Google. (2017). The 7 steps of Machine Learning [Video file]. Retrieved from https://www.youtube.com/watch?v=nKW8Ndu7Mjw

Lubbad, M. (2017). *Android Malware Prediction by Permission Analysis and Data Mining*. Retrieved from https://deepblue.lib.umich.edu/bitstream/handle/2027.42/136197/YouchaoDong\_Thesis\_0327.pdf?sequence=1&isAllowed=y

MalwareAnalysisForHedgehogs. (2017). Malware Analysis - Getting Started with High-Level Petna / Petya [Video file]. Retrieved from https://www.youtube.com/watch?v=uWzQr8PFbfA

McAfee. (2018). *Mobile Threat Report*. Retrieved from https://www.mcafee.com/enterprise/en-us/assets/reports/rp-mobile-threat-report-2018.pdf

Meng, G. (2017). *A Semantic-based Analysis of Android Malware for Detection, Generation, and Trend Analysis*. Retrieved from http://www.ntu.edu.sg/home/yangliu/publications/guozhuthesis.pdf

Mulliner, C., Liebergeld, S., & Lang, M. (2017). HoneyDroid - Creating a Smartphone Honeypot. Retrieved from https://pdfs.semanticscholar.org/20a6/822db127593af37cd6b2a01bcb9663f904f1.pdf

Nokia. (2016). *Nokia Threat Intelligence Report – 2016*. Retrieved from http://nokiamob.net/wp-content/uploads/2016/09/Nokia-TI-Report-H1-2016.pdf

Nokia. (2017). *Nokia Threat Intelligence Report – 2017*. Retrieved from http://nokiamob.net/wp-content/uploads/2017/11/Nokia-Security-Report.pdf

Pedersen, J., & Stevanovic, M. (2015). *AAU-Star and AAU Honeyjar*. Retrieved from http://vbn.aau.dk/files/222852089/AAU\_Star\_and\_AAU\_Honeyjar\_JMP.pdf

Peter, E. (2017). A Practical guide to Honeypots. Retrieved from https://www.cse.wustl.edu/~jain/cse571-09/ftp/honey/index.html

Portokalidis, G., Homburg, P., Anagnostakis, K., & Bos, H. (2010). *Paranoid Android: Versatile Protection For Smartphones*. Retrieved from https://www.cs.vu.nl/~herbertb/papers/pa\_acsac10.pdf

Ransomware Past Present and Future. (2017). Retrieved from https://documents.trendmicro.com/assets/wp/wp-ransomware-past-present-and-future.pdf

Rong, H. (2017). *Evaluation of the Android Third-party stores*. Retrieved from http://www.diva-portal.se/smash/get/diva2:972682/FULLTEXT01.pdf

Savage, K., Coogan, P., & Lau, H. (2017). The Evolution of Ransomware. Retrieved from http://www.symantec.com/content/en/us/enterprise/media/security\_response/whitepapers/the-evolution-of-ransomware.pdf

# Appendixes

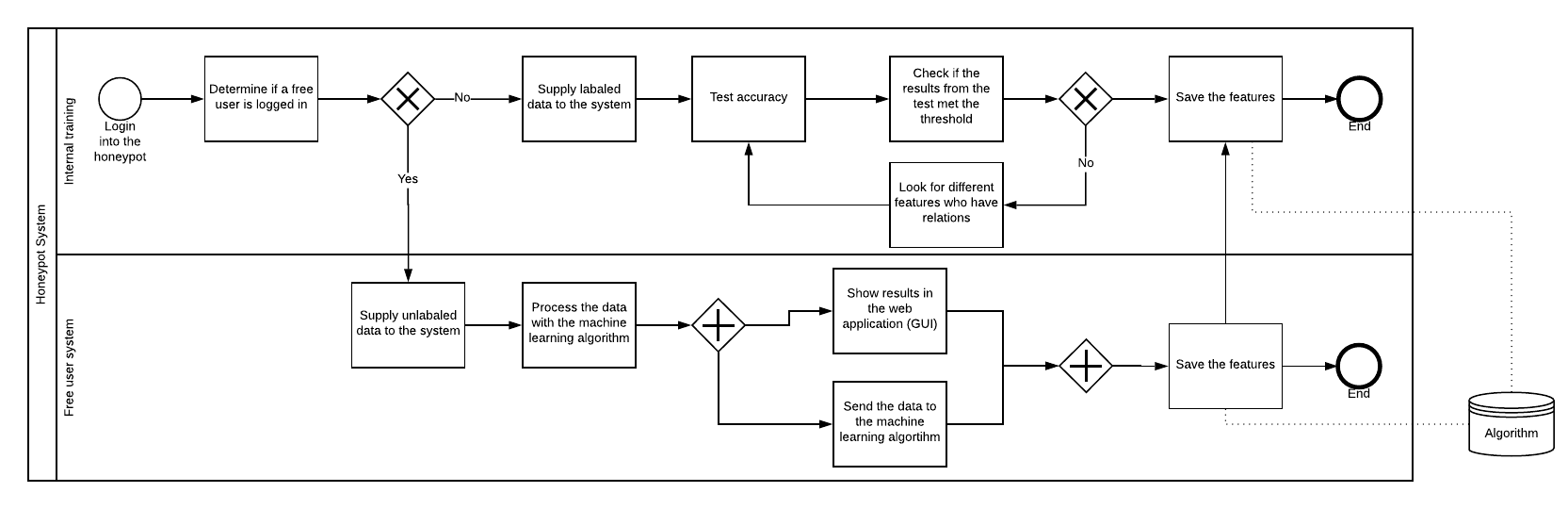
I BPMN

II GUI

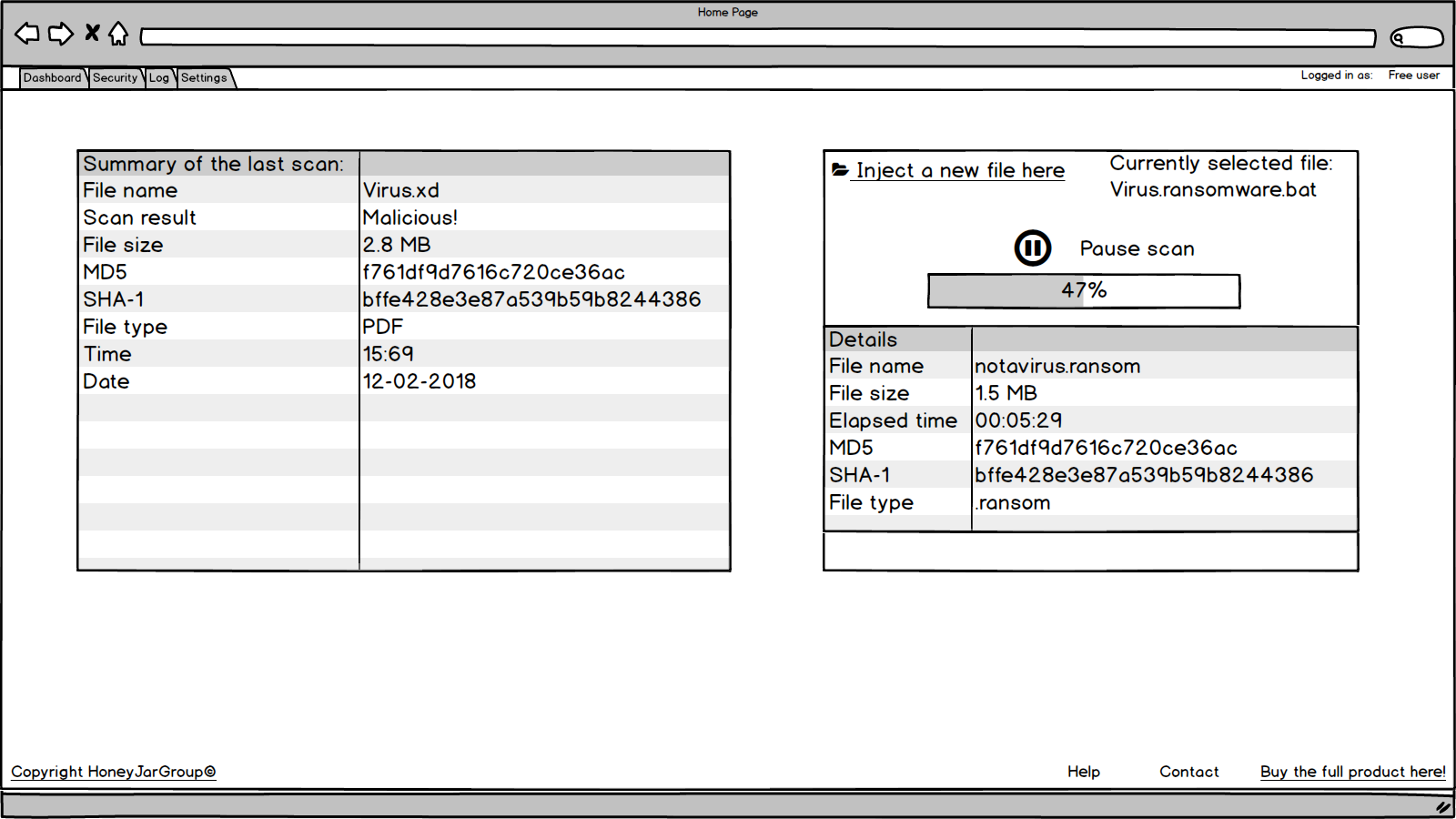
III Transcripts

IV Coding

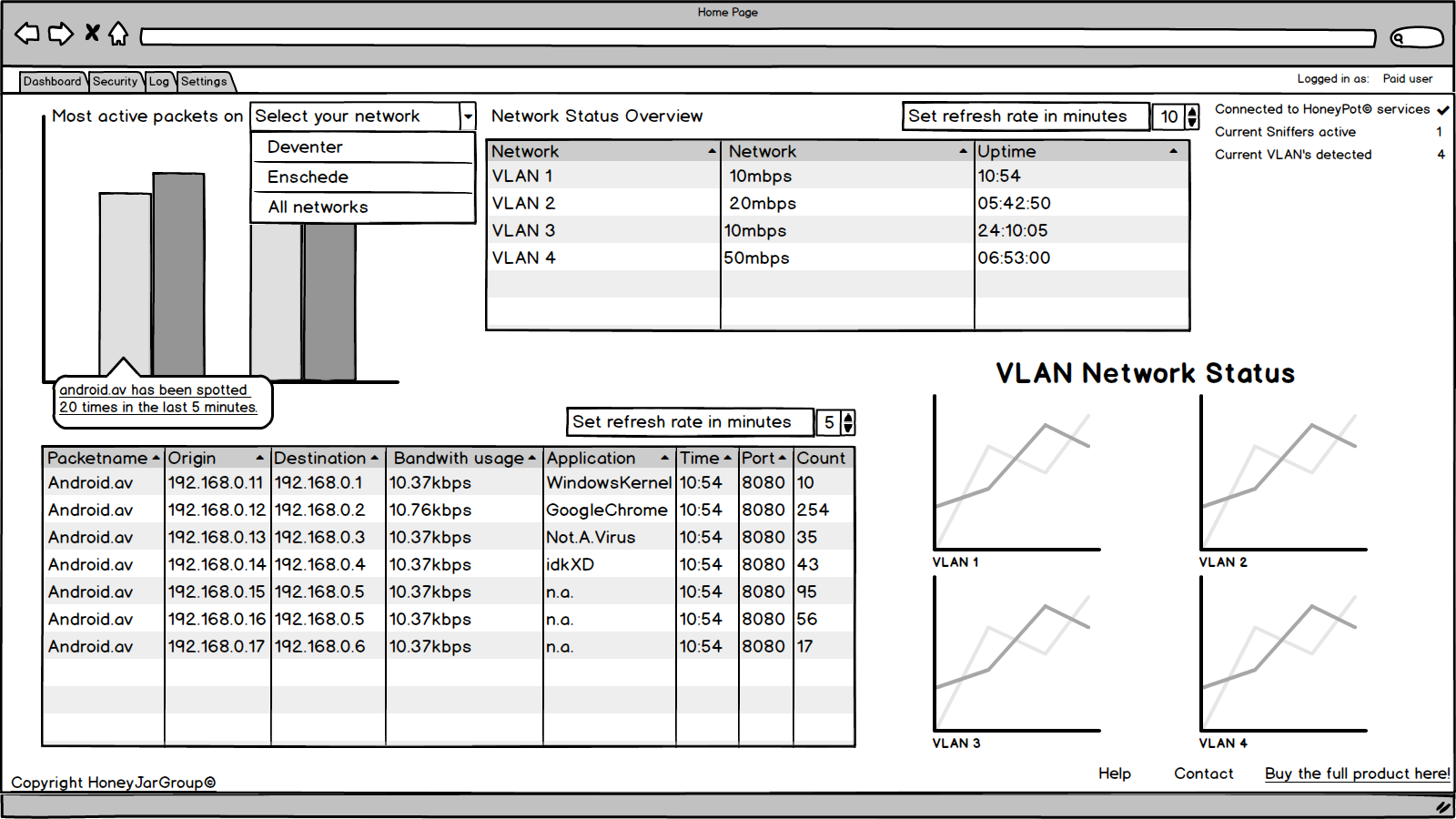
## BPMN Shadow HoneypotC:\Users\Morcel\Downloads\Shadow Honeypot.jpg

**Both free user options**

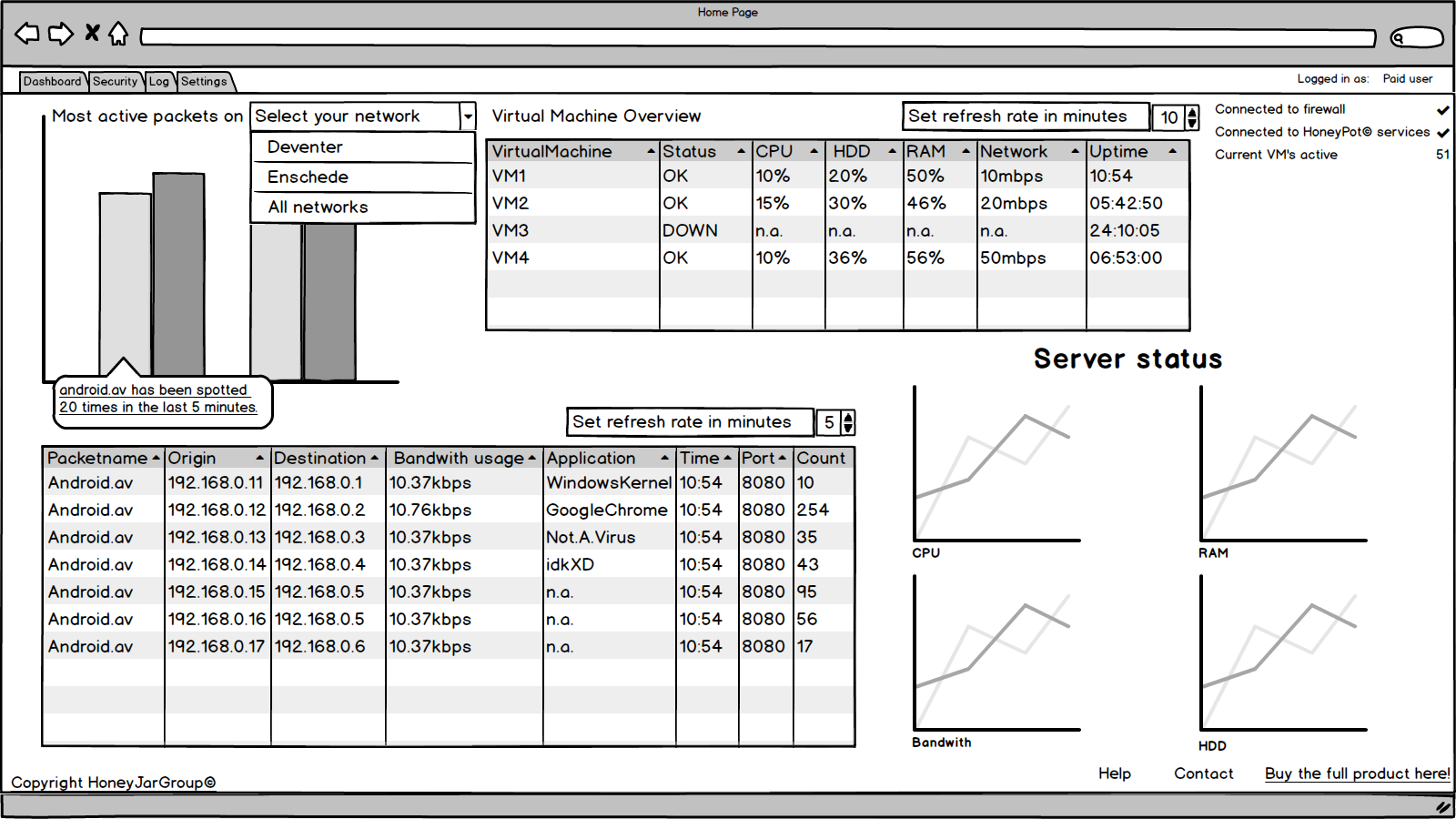
### FREE USER DASHBOARD



### NETWORK SNIFFER DASHBOARD



### SHADOW HONEYPOT DASHBOARD



## Transcripts

Interview Valentin Carela

Security

How big of an issue has cybersecurity become in the last couple of years? How do you notice that the most?

Nowadays security is basic. Before it was only basic for big companies. Nowadays even the smaller companies have noticed it’s crucial for their services. Now the governments and the states are looking after the data of the company and users. It was a hot topic, but nowadays it’s even more important. A single attack was able to destroy big companies at once, like amazon, twitter, facebook.

Before someone attacked Playstation, irrelevant services, but nowadays also governments etc.

What type of Malware is currently in your experience the most popular?

I would say there are 2 different branches. DDOS. And the other one is related to the exploit data from users and/or companies. We’ve seen cases that focus on a specific company. It was a tool designed specifically to hack that company. I think those are the most common ones. Denial of Service is a problem because everyone is aware of it, but the data that leaks Malware is sneaky. It goes unnoticed. It moves very fast. So I would say that the most popular one is Spyware. However it’s not the most dangerous one. I’m going to give you some categories of Malware. On the one side you have viruses that you can combine with software, i.e. a torrent client.

Another category is worms. It’s a piece of software that goes over the network, infecting other machines on the same network. You can steal any data, credentials. Another category would be Trojan Horse that is installed manually by the user. This enables you to reach a computer outside your network, like a virtual machine with windows but actually with a real computer. This goes unnoticed.

Then you have spyware, that enables addons to other software, what they do is extract information from your devices and share it with someone else, again credentials, and data. Now we have Grey worms, its technically not Malware, for example Google and Facebook, they ask you for permission to use your data. The difference between this and spyware is that they give you a better ‘’service’’ like ads, attuned to the user. Also botnets. Ransomware is probably related to worms, but might be it’s own category really.

How do you make sure you keep up to date in an environment that moves very quickly.

Usually for companies it’s being one step behind, because we’re preventing. Sometimes we don’t say as we try to stay on the same level. What we do is usually do research about what the universities are doing, what’s relevant in conferences, always up to date, always implementing new proposals that are state of the art.

In talaia’s case we try to be agile, we can modify things fast and it’s easy for us to be on the edge. Sometimes companies willingly don’t keep up because of finances.

SDM software defined networks. Separate the routers and switches, instead you have one machine that forwards packets, and then there’s a brain that manages all these packages. What happened here, was that they already spent all their money building super expensive machines. If you do a Honeyjar like this you want to keep hardware the same, the system should be modular. Better even, use software. Much easier to manage and implement. Plug and play. So instead of paying thousands of euro’s for machines in the network use software, this can be implemented anywhere. Hardware is unneccesary in this case. It’s a complexity you want to avoid.

What makes a Honeypot attractive?

Make it easily deployable, isolated. I.e. there are solutions like enefzen, that cost 2 weeks only for installing the tool. It doesn’t always work. Very unreliable. It would also be useful if you could introduce some Malware in the Honeypot. Something like a demo. Imagine that you can install different VM’s with Malware inside, you can try any combinations. This makes it very attractive. One of the key points is to keep it simple. Other functionalities that should be in the Honeypot is to check what have been the most important Malware attacks in the last 3 years for android, and ID them. Install these in the specific VM’s. This is actual. “Remember this recent big attack?” This triggers the company.

Android

How embedded have mobile phones become in your company?

In our case it’s not important. Because we focus in a completely different area. Our target is usually ISP’s and company and we focus on the monitoring of them. We don’t focus on differentiating between phone and computer traffic. However, we use our phone for operational things, like SLACK, Whatsapp, Email. I have a console with a VPN to connect with the machine in Talaia. It’s not comfortable but it’s useful for emergencies. I don’t recommend to use your cellphone for work. I could do my work without my phone. Nowadays I don’t trust anyone so I try to keep as little personal information on my phone.

How big of a risk have mobile phones become concerning cybersecurity, in your experience.

I’m cynical. Because people don’t realize how much information is stored in their phone. For my mother that’s receiving a Whatsapp message with like a discount code, having her log in to facebook, allowing permissions. It happens very quickly. For example Facebook was finding out what the best place for a political meeting was. Depending on the information. Facebook takes advantage of this, and as well as fake news. News presented in a very specific way. There are a lot of subliminal controlling factors in facebook.

How much manpower, realistically, would be needed to operate a Honeyjar system like ours? (compare with talaia here)

It depends. If you just focus on Malware just 1 guy is fine. If you allow the user to install whatever they want so they can evaluate, it can be useful for others too. Maybe 2 or 3. Companies are not employing people for this specific topic. Maybe in a company that develops games for Mobile phones.

Network Security

Q about research:

First I’m not an expert, but I’m going to explain you what we know. So you were talking before about using deep learning. The good thing is that when there is a Machine Learning problem, try to extract all the features that you and from the data that you have, even if it’s not important for you. The Machine Learning model will tell what information is relevant to the ML system. Don’t try to understand be relation between the distribution size of the Malware because that’s going to be tough. See it is a black box, input and output. Rest is not relevant.

What is important is to try to insert features that are not dependant of the rest. i.e. Fashion and female would be dependant probably. With the data you have show a list of all the features and apply models for output. Compare these.

Q about Labeled data: There are 2 approaches. Unsupervised ML and supervised. In our case what would work best is supervised ML. Honeyjar is going to provide us data. The supervised ML will collect data from the internet, you put it in the model and the model creates some clusters. Manually inspect these clusters. If you see common parameters you can tell where its coming. Try to find similar parameters in the non-common data.

Q about encrypted traffic: Should we consider encrypted traffic or not?

The trend is that everything is moving to encrypted, but it’s more difficult to detect. Maybe not worth to do.

Q about Android: What is the point of Android in this case? It’s the same as Android. What’s the reason behind Android phones? Depends on what you want and what you can do. The start is the Honeyjar. Android is not as common as other systems. It’s a hot topic so it’s a good environment to test. This makes work easier, cos it focuses on 1 point.

Cloning Joy on local server is a good idea. Importing multiple PCAP files, if something is wrong it will make an analysis of that part. Taking those and modifying them and combining them in an effeicient way, is this a good idea? (Straffor)

If there is no real labeled and unlabeled data then should we consider deep learning on the packets that we are working on it.

Should we consider encrypted traffic directly or not ?

Do you think that we are at right direction by checking http requests and getting some responses from API? (-there will be 15 sec delay and which is huge delay -)

Is it ok to use frameworks and modify some functions of existing open source tools according to our needs ?

The extracted features are related with DdoS attacks and we want to identify malicious activity but the features are NOT so much related with the identfying malicious activity.

Feedback on GUI

Danish guys:

Server security. All the same VM’s are going to show the exact same thing because they’re all on the same server.

Saxion:

Maybe you can present data that is related. Detection model is made in Machine Learning. I don’t think it’s the right approach. You cannot differentiate traffic inside the Honeypot. This is a second step. First you need to get the data and use it for learning process. If you use ML to ID what is malicious and what is not.

Should monitor HDD, network. What are the processes that are running, what’s being used. Once you enable Malware and differences appear in the logs you can differentiate the Malware. Focus on the process.

Geographical location/Type of company. An idea is to use information gathered from customers. Or we can directly sell it. Subscription based for example. But then you have to get the permission from the users. Giving them access to all the data from the customers is valuable. Give a price for the deployment of the Honeyjar and allow the customers to join using subscriptions.

One of the challenges is good accuracy with Machine Learning. We’re getting too many False positives, this is a drawback – and it’s specific to the company. We’re closing the gap the main channels of the seminar project today. A system can be trained for a specific kind of company to deal with false positives.

## Interview Wesley

Morcel: Ik heb je net kort al even uitgelgd wat ik ongeveer doe voor mijn project, maar dat zal ik nu nog een keer even uitleggen.

Het gaat voornamelijk om cyber security. Onze groep heft onderzoek gedaan naar cybersecurity en hoe dat op dit moment is. Wij hebben ervoor gekozen om…. We werken in een groep van negen mensen. Eentje uit polen, maar die is zo goed als weggevallen, eentje uit Turkije, vijf uit denemarken en twee uit nederland.

De deense jongens hebben vorig half jaar een Honeypot gebouw, een android Honeypot. Het was heel basic en nog niet helemaal af, daar zijn ze nu dus mee verder gegaan. Nu zijn wij er dus bij gekomen.

Hun opdracht was dus om eenn Honeypot te bouwen, Android Honeypot dus. Toen zijn wij erbij gekomen, Alex en Ik saxion in dit geval en toen was de vraag hoe kunnen we dit profitable maken? Hoe kunnen we dit sellen, hoe kunnen we dit verkopen?

Toen hebben we onderzoek gedaan naar cybersecurity in het algemeen, is dat een issue op het moment en toen zijn we erachter gekomen dat het een redelijk hot topic is.

Wesley: Zeker weten. Ja absoluut. Dat verkoopt zoiezo wel lekker.

Morcel: Ja, ja ja. Laatst had je natuurlijk de WannaCry attacks en de Petya/NotPetya. Dus het is wel een redelijk hot topic.

Toen dachten we van: Oke ze hebben dus al een Android Honeypot gebouwd, wat kunnen we hier verder mee?

Toen zijn we tot de conclucie gekomen dat ransomware wel redelijk on the rise is. Dat wordt steeds meer gebruikt. Files encrypten en betalen en dan betaal en krijg je nog je bestanden niet terug. Maar het zorgt er in ieder geval voor dat je bedrijf een tijde plat ligt.

Wesley:

Heb je dat wel eens gehoord? Dat je dan alsnog wel naar de helpdesk van die Ransomware kan bellen?

Morcel:

Ik heb het weleens gehoord maar…

Wesley: 02:14

Dan krijg je gewoon iemand uit rusland aan de telefoon en die gaan je dan legit proberen te helpen om die betaling over te maken en dan gaan ze het voor je regelen.

Morcel:

En dan wordt alles gewoon gedecrypt?

Wesley:

Ja, ja. Normaal gesproken wel. Er is wel een tijd geweest waarin dat niet gebeurde maar dat kwam omdat er een fout in de software van de ransomware zat.

Morcel:

Ja, ik had van die NotPetya gelezen dat je wel kon betalen, maar dat die code zo buggy was dat ze niet konden zien dat je had betaald dus krijg je het nog niet niet terug. En die hele decrypt service is ook niet zo goed dus eigenlijk betaal je dan voor niks en kan je nog steeds alles restoren.

Er was ook toen een Deens bedrijf, Maersk heette dat. Het was een havenbdrijf en ze deden logistiek met die zeecontainers. Die was ook gehit.

Wesley:

Ooh ja daar heb ik wel van gehoord ja.

Morcel:

Dat was super massive, toen konden ze voor een week hadden ze geen IT. Ze moesten alles handmatig doen, via whatsapp enzo. Daarna hebben ze natuurlijk cybersecurity serieus opgepakt. Maar dat gaat dus altijd zo.

Maar Android en Ransomware, het is op het moment niet heel populair maar het wordt steeds meer. Er zijn varianten zoals doublelocker en die locked je phone en gaat dan heel irritant lopen doen en dat gaat pas weg als je hebt betaalt.

Ransomware en telefoon is dus….

Wesley:

Heb ik nog niet gezien in ieder geval.

Morcel:

….nog niet heel populair. Het bestaat wel maar het is dus nog heel weinig inderdaad.

Wesley:

Maar dat zal inderdaad wel gaan groeien, het zal me niks verbazen inderdaad.

Morcel:

Een andere reden dat er voor Android is gekozen is het feit dat iedereen van alles doet met een telefoon. Iedereen denkt het staat op mijn telefoon dus het is veilig.

Wesley:

Alles is veilig. (sarcasme)

Morcel:

Paspoorten, bankgegevens. Alles staat erop en als je dan toegang hebt…

Wesley:

Iedereen gebruikt het dus het zal wel veilig zijn. (sarcasme)

Morcel:

Ja nee dat dus. Wel grappig dat mensen dat denken.

Ja, we hebben dus een android based Honeypot systeem, android is de focus.

Eerst even wat standaard basic vraagjes.

Wat is je functie bij je huidige werk, wat doe je precies?

Wesley:

Ik werk bij Tedas in brummen, daar doe ik netwerkbeheer. Voornamelijk op het gebied van security, netwerk infrstructuur ontwerpen en zoeken naar verbeteringen.

Morcel:

En wat voor verbeteringen, gewoon qua beveiliging of infrastructuur?

Wesley:

Op het moment nog niet echt beveiliging op het niveau om andere netwerken te verbeteren dan ze al zijn is nog te hoog gegrepen, met de ervaring die ik nu heb. Het gaat er nu nog voornamelijk om netwerkjes die basic zijn opgebouwd en steeds verder zijn gaan groeien en daar niet echt goeie richtlijnen op zijn toegepast waardoor alles er heel rommelig uit kwam. Denk maar aan hele netwerken in vlan01 gooien. NAT op NAT. Je kan het zo gek niet bedenken. Dat is allemaal 1 grote chaos daarbij moet ik verkeersstromen in beeld gaan brengen kijken welke firewalls er overal op staan. Dat uit gaan kammen, structuur in brengen en weer opnieuw in elkaar gaan zetten.

Je ziet eigenlijk ook nergens dat een netwerk van scratch gelijk goed wordt neergezet, er komen overal wel fouten in. En daar probeer ik dan structuur in te brengen.

Voornamelijk om te kijjken naar verkeer dat door bepaalde devices gaat dat eigenlijk niet hoort dus om dat een beetje dicht te timmeren. Of bijvoorbeeld een heel subnet dat toegang heeft tot een bepaalde server zonder porrtconfiguratie of wat dan ook. Kijken wat er nodig is en dan daar specifiek op gaan filteren.

Morcel:

En als je dan diep in zo’n netwerk zit zie ja natuurlijk ook wel zwakheden

Wesley

Ja

Morcel:

En hoe is het naar jou mening gesteld met de netwerk security?

Wesley: 07:45

Ik heb dan in het verleden die ethical hacking cusus gedaan. En bij mij is er dan als conclusie uitgekomen dat als je iets gewild wilt aanvallen, ergens binnen wilt komen, bijna geen enkel netwerk is veilig. Je kan overal uiteindelijk wel binnen komen als je zou willen.

Morcel:

Is dat met of zonder social enginieering, want social engineering….

Wesley:

Social enginieering doet al heel veel, daar is het grootste issue bij de meeste netwerken. Maar grotendeels is alles op puur hacking niveau ook wel te doen.

Morcel:

Dus er is geen enkel netwerk zo betrouwbaar/stabiel dat je niet binnenkomt?

Wesley:

Nee, zeker niet.

Morcel:

En welke oplossing worden daarvoor geïmplementeerd als ze worden geïmplementeerd? Of wordt het gewoon een beetje weggevegen?

Wesley:

Haha, vaak is het zo dat als het een keer fout gaat en alles komt boven water gaan ze het dicht proberen te timmeren als het dus eigenlijk al te laat is. Dat soort dingen zijn heel moeilijk om van te voren dicht te gaan zetten.

Morcel:

Dat is zeker waar

Wesley:

De meeste fouten gebeuren naar mijn mening door systemen niet te gaan patchen, verouderede systemen. Dan hoor je in het nieuws opeens, als je securitynl bijvoorbeeld volgt, hoor je in een keer dat er een nieuwe vulnirability is gevonden in een bepaalde software. De hele wereld gaat dan vervolgens dat proberen te scannen en kijken waar dat lek inderdaad zit en er dan massaal op gaan aanvallen. Dan kan je daar heel snel een netwerk mee platgooien als ze hun zaken niet op orde hebben.

Morcel:

Ja, is het dan ook gelijk… Vindt je het dan ook moeilijk om proactief te handelen met netwerk security? Je bent natuurlijk altijd een stapje achter.

Wesley 09:48

Jazeker, haha. We hadden laatst op mijn werk heel veel microtik routers staan, zo’n 1500 en zoals ik eerder ook al zei is het slordig opgebouwd maar wel enorm gegroeid. Ze hebben bijvoorbeeld geen centrale update server. Nou wat gebeurde er, paar weken geleden was er een kwetsbaarheid gevonden, in een microtik router os versie waarbij bepaalde managementport; WinBox benaderbaar was en je er gebruikersnamen en wachtwoorden uit kon slopen. Dat was in een aantal software versies. Nou wij hadden ik denk wel meer dan 1000 routers, waarvan sommige aan het internet hingen, die hadden die verkeerde software versies. Wij hadden dan geen centrale software server dus wij konden alles 1 voor 1 updaten. Dan denk je toch van; Goh, hier had misschien wel eerder over nagedacht moeten worden. Maar goed dan gebeurt zo iets en wij zijn nu bezig met het implementeren van een centrale patch server voor de volgende keer als er weer een brakke software is.

Morcel:

En hoe waren jullie hierachter gekomen?

Wesley: 11:05

Door de forums van microtik te volgen, dus nog voordat het in het nieuws kwam. Je hebt vaak wel sites zoals securityNL, die wachten een paar dagen voordat ze het erop zetten om beheerder nog even de kans te geven te gaan patchen voordat het naar het publiek wordt gegooid. Want zodra het naar het publiek wordt gegooid gaat iedereen natuurlijk proberen om er misbruik van te maken.

Morcel: 11:29

Merk je dat ook?

Wesley:

Ja, zeker.

Morcel:

Zie je dan ook veel meer traffic?

Wesley:

Veel meer traffic op bepaalde poorten zie je overal naartoe gaan, ja.

Morcel:

Haha, dat is wel grappig

Wesley:

Hehehe ja dat is zeker wel grappig.

Morcel:

Ik neem aan dat je ze allemaal wel kan blocken. Of ze kan tegenhouden. Komt er ook weleens wat tussen?

Wesley:

Het probleem heel vaak bij dat soort dingen is; Je kan het wel tegenhouden maar je moet ook verkeer ook wel toe kunnen laten dat wel ernaartoe hoort te gaan.

Morcel:

Ja

Wesley:

En dat is wel heel lastig, voornamelijk als je netwerken hebt dat je door het hele land heen hebt zitten.

Morcel:

En hoe filter je dan tussen slecht en goed?

Wesley:

Uuhm, Ja dan moet je het goed hebben opgezet. Dan moet je VPN overal hebben geïmplementeerd. Maar dit is niet overal het geval dus sommige system hingen nog aan het netwerk dat wel kwetsbaar was en dan maar hopen dat ze niet gehit worden. En je in dat geval kon je de management poorten wel dicht gaan zetten en wanneer je erbij moest komen moest je iemand opbellen die het met de hand even open ging zetten.

Morcel:

Ik had altijd het idee dat je wel kan onderscheiden wanneer iets kwaadaardig is en niet…

Wesley:

Het is moeilijk

Morcel:

Maar dat is dus wel iets lastiger. Doe je dan ook deep traffic analysis of doe je dat niet? Kost dat teveel tijd?

Wesley:

Ja het kan wel, maar bij dit netwerk is dat niet echt mogelijk geweest. Bij saxion in het verleden kon dat wel. We hadden daar ook een next generation firewall ziten en daar kan je ook veel dieper in pakketjes pakken wat er nou aan de hand is en waar het vandaan komt, maar bij dit netwerk ij tedas is dat nog niet zo netjes geregeld maar dat komt er allemaal nog.

Morcel:

Nou onze focus is dan op android telefoons zoals ik al zei. Merk je ook dat telefoons malicious packets doorsturen of..?

Wesley:

Telefoons kunnen wel heel veel nare dingen met zich mee brengen vooral met het concept tegenwoordig van Bring Your Own Device. Dat was bij Saxion ook een hot topic. Want je hebt al die studenten, deocenten en mensen van overal. Die komen dan binnen en willen allemaal zo makkelijk mogelijk gebruik maken van het netwerk en wat kan hun het nou schelen wat voor beveiliging er allemaal achter zit, daar hebben ze toch geen verstand van. Het is dus weleens voorgekomen dat laptops of telefoons waar dan wel malicious software op zat dat dat problemen ging veroorzaken.

Morcel:

Zoals? Wat zie je dan?

Wesley:

Grotendeels werdt het al wel tegengehouden. Op het moment dat ze het netwerk op wilden komen werdt het al afgekaatst. Maar het gebeurde wel eens dat mensen op het netwerk zelf zaten en op een verkeerd mailtje klikten waardoor inderdaad bestanden geencrypt werden.

Bijvoorbeeld ik vertelde je toen ook nog van die collega, die was niet zo heel erg snugger. Die werkte bij de eerste-lijn support en we hadden toen heel erg last van phishing mailtjes, ransomware. Maar zijn zoontjes deed een of andere ICT Opleiding MBO 2 of 3. En zei; Hey Pap, ik heb hier software gevonden die bestanden kan decrypten zonder probleem.

Hij had natuurlijk het licht uitgevonden want daar hadden wij nog nooit aan gedacht om bestanden gewoon te decrypten, probleem opgelost toch?!

Dus hij zegt tegen z’n zoontje: stuur maar naar mij toe, ooh dat kan niet via de mail? Doe maar via wetransfer.

Nou hij heeft die troep op z’n laptop gezet, op z’n werk waar hij dus ook meer toegang had omdat hij bij support werkte en toen werd een groot deel van het netwerk plotseling geencrypt. Rara hoe kan dat.

Morcel: 15:44

Was dat een werk laptop of z’n eigen laptop?

Wesley:

Z’n werk laptop.

Morcel:

Dat is helemaal fijn. En toen lag alles plat?

Wesley:

Toen moesten we een back-up uitrollen.

Morcel:

En dat kost natuurlijk tijd en tijd is geld.

Wesley:

Ja, tijd en geld. Je moet ook bij dat soort verhalen ook met ransomware rekening mee gaan houden van hoeveel bestanden er geraakt zijn en hoeveel werk uren daarin zitten. Is het, het dan wel waard om alles terug te gaan zetten in plaats van het bedrag te betalen.

Morcel:

Hebben jullie dat toen ook gedaan?

Wesley:

Nee, dat hebben wij toen niet gedaan maar dat gebeurt soms wel. Daar houden de developers van die ransomware rekening mee. Hoeveel zou dat ongeveer gaan kosten (om het terug te zetten) en daar handen ze dan een prijskaartje aan zodat bedrijven gaan overwegen om te betalen.

Morcel:

Mits je de garantie hebt dat het wordt gedecrypt natuurlijk.

Wesley: 16:32

Ja, maar dat gebeurd over het algemeen wel.

Morcel:

Daar verdienen ze dan wel flink geld aan.

Wesley:

Ja zeker, dat is een mooi stukje software haha.

Morcel:

En het voorbeeld waar je het net over had. Hoe heb je ervoor gezeorgd dat het niet verder ging spreiden over het netwerk? Heb je he direct geblocked? Heb je het afgesloten?

Wesley:

Het is bijna niet te blocken. Ik heb gewoon alle medewerkers gevraagd, dat was toen ook een projectje van mij, als zij mailtjes van… Ze hebben eerst ook cursussen awareness training gehad; wanneer lijkt het inderdaad op een nepmailtje wanneer is het een phishing mailtjes en dat soort dingetjes.

Als ze zo’n soort mailtje kregen, vroeg ik of ze het wouden doorsturen naar support en dat zou dan naar mij doorgestuurd worden.

Ik ga dan in het mailtje kijken wat voor website erachter hangt als je op het linkje zou klikken en die websites heb ik allemaal op een blacklist gezet, die IP adressen.

Morcel:

Handmatig?

Wesley: 17:38

Dat werd op een gegeven moment een mooie lijst toen dacht ik van; Goh er zullen vast wel andere mensen zijn die dat ook doen. Dus ik heb op internet gezocht naar hele lijsten van IP adressen waarvan men dacht inderdaad dat het daarvan daar kwam, dus die heb ik er allemaal bij in gegooid en op een gegeven moment na een aantal maanden kreeg ik dan weer een mailtje: Deze is ook malicious. Als ik er dan op klikte gebeurde er niks omdat het toch allemaal al geblocked was. Dus het hielp uiteindelijk wel heel veel maar dat heeft gewoon tijd nodig en dat moet je blijven bijhouden en dat is gewoon niet handig.

Morcel:

En stel je voor dat zoiets automatisch zou zijn?

Wesley: 18:07

Daar waren wel oplossingen voor en hebben we naar gekeken. Toen dachten we dat gaan we kopen maar, dat is uiteindelijk niet gedaan vanwege beleid en bla bla bla. Mensen zagen het nut er niet van in.

Morcel:

Bureaucratie.

Wesley:

Want het was maar een paar keer fout gegaan.

Morcel:

Maar een paar keer, totdat het in een keer goed fout gaat natuurlijk.

Wesley:

Maar ja mensen kijken natuurlijk alleen maar hoe vaak is het fout gegaan maar niet hoe vaak het is goed gegaan.

Morcel:

Dat is ook wel een stuk moeilijker te meten natuurlijk

Wesley:

Ja, Absoluut.

Morcel:

Je zei dat net dat als er iets binnenkomt op een telefoon dat het meeste malicous traffic werd afgekaatst. Hoe wordt dat afgekaatst, heb je een netwerk sniffer staan, firewall?

Wesley:

Dat was… Hoe heet dat ook al weer? Misschien weet jij dat nog wel, als je bij saxion in wou loggen op het netwerk werden sommige devices direct geblocked door een of andere service, maar daar ben ik de naam even van kwijt. Ik weet niet precies hoe dat werkte maar die keek als een soort virusscan op je device en op basis daarvan kreeg je dan wel of geen toegang tot het netwerk.

Morcel:

Dus hij kijkt meer naar je specifieke….

Wesley:

Gewoon lokale bestanden die erop staan

Morcel:

Hij kijkt niet naar je netwerk traffic of iets dergelijks?

Wesley:

Nee volgens mij niet maar dat weet ik niet 100% zeker

Morcel:

Dat maakt niet uit. Wij willen dus dat Honeypot systeem gaan bouwen, we hebben een paar ideeën van hoe we dit kunnen implementeren. Ik zal ze even kort toelichten en dan kijken wat je ervan vindt.

We hebben het eerste idee, dat is eigenlijk een netwerk sniffer. Dus die zet ja aan het begin van je netwerk neer, dus die zit precies wat er naar binnen en buiten gaat. Hij kijkt hoe die traffic is en komt ie iets tegen dan kaatst ie het af en dat wouden we doen door middel van Machine Learning.

Wat Machine Learning inhoudt is dat je een machine of iets gaat trainen door middel van AI. Dus als je bijvoorbeeld een machine hebt heb je Bees en Threes. Een drie en een bij. Die kan hij niet onderscheiden. Ik kan een drie schrijven in deze bank of ik kan hier een drie schrijven. Ze betekenen allebei hetzelfde maar dat snapt die machine niet. Als je hem genoeg samples geeft kan hij ze uiteindelijk wel onderscheiden. Als je de machine dus genoeg voorbeeld geeft kan hij uiteindelijk zelf nieuwe voorbeeld onderscheiden zonder dat wij hier iets voor hoeven te doen.

Wesley:

Ja, dat soort software is hartstikke mooi.

Morcel:

Zou zoiets kunnen werken denk je?

Wesley:

Ja, dat zou zeker kunnen werken. Het zijn ook wel services en pakketen waar men op het moment mee bezig is om neer te kunnen zetten maar dat soort dingen is gewoon heel lastig om te bouwen.

Morcel:

Het enige… Nou ja probleem…. We kunnen het natuurlijk niet bouwen dit project maar het enige probleem wat we op dit moment kunnen voorzien is de hoeveelheid data dat we krijgen om te trainen. Want je hebt ontiegelijk veel data nodig om zoiets te kunnen trainen. Heel veel voorbeelden en uiteindelijk moet hij het zelf kunnen. Hij moet de data packets gaan bekijken, dat is het deep packet analyses waar ik het over had.

Je hebt tock ook altijd gewicht en lengte die met elkaar verbonden zijn, die hebben altijd iets met elkaar te maken.

Wesley:

Ja, ja ja ik ken het wel ja.

Morcel:

Nou dat is dus het eerste idee, een netwerk sniffer. Die gaat kijken naar het netwerk en naar het algoritme om te kijken of er iets aan de hand is.

We hebben nog een idee alleen dit is iets meer uitgebreid en dat is een shadowHoneypot. Ik weet niet of je dat kent.

Wesley:

Nee

Morcel: 22:31

Je hebt je gewone Honeypot, dat is een netwerk die zich probeert voor te doen als een normaal echt netwerk. Dan denkt de ransomware; ow dit is een echte omgeving hier kan ik op deployen. Als hij doorheeft van oke, dit klopt niet dan klapt hij dicht en doet ie niks.

Als je een shadowHoneypot hebt heb je je normale netwerk, alleen net daarvoor dus voordat je op het normale netwerk komt kom je op een soort nep netwerk. Maar die is precies hetzelfde ook qua behaviour enzo. Als er dan ransomware binnenkomt denk hij dat het een echt netwerk is en gaat ie van alles encrypten en dan kapt onze oplossing het af en wordt het daadwerkelijke netwerk niet aangetast.

Dit is dus meer dan sniffen dit is echt proactief tegengaan.

Dit zijn dus de twee oplossingen die we in gedachten hadden.

Wesley:

Maar dat verkeer dat dan.. Je ziet al het verkeer van een IP-adres vandaan komen die die actie op die Honeypot doet. Wat doe je daar vervolgens mee? Zet je die op een blacklist of,,?

Morcel:

Daar is dat algoritme dus voor, daar wordt het in opgeslagen en hij leert terwijl hij bezig is.

Nu zijn we op het moment met een GUI ontwerp bezig en ik weet dat je heel erg van de command line houdt…

Wesley:

Nou ik werk nu met microtik en dat is ook allemaal GUI gebaseerd en dat werkt gewoon logisch. In het verleden heb ik met GUI’s gewerkt en dat is gewoon compleet onlogisch en dan was het gewoon kut maar als het werkt is het zeker fijn.

Morcel:

Ik heb hier een plaatje van een shadow Honeypot. Hier heb je internet en daarna zitten al je services.

Wesley:

Ow dat is een cisco template

Morcel:

Ja, dat klopt haha. Dan heb je dus voor je echte netwerk je Shadow Honeypot zitten en die is exact hetzelfde als je daadwerkelijke netwerk. Dus voordat het naar je daadwerkelijke netwerk gaat komt het eerst in de Honeypot, als daarin allemaal gekke dingen gebeuren komt het niet binnen in het echte netwerk.

Wesley:

Moet ik dat dan zien als een soort sandbox waarin het daadwerkelijk wordt uitgeveord om er achter te komen of het malicious is of niet?

Morcel:

Ja

Wesley:

En op die manier leert hij dan?

Morcel:

Ja, op die manier zier hij dan dat het malicious is en dan gaat ie daar onderzoek op doen

Wesley:

En als hij het zeker weet gooit hij dat vervolgens naar de firewall aan de externe kant en dan gaat hij het vanaf daar blokkeren?

Morcel:

Ja, dat is de bedoeling.

Wesley:

Dat is wel mooi, dat zou cool zijn. Het zou super handig zijn als dit bestaat.

Morcel:

Stel je hebt zo’n systeem, en dit is natuurlijk heel lastig om te bedenken, maar wat zou je in je dashboard willen zien?

Wesley:

Voornamelijk als er malicious pakketen binnen komen dat we kunnen zien waar het vandaan komt wat het probeert uit te voeren, wat het doet inderdaad, dat hij dat in het vervolg gaat blokkeren en proactief tegen gaat houden vanaf de andere kant.

En dat je daarna ook echt verschil gaat zien in load op je firewalls.

Morcel:

Denk je dat zo’n systeem veel onderhoud nodig heeft?

Wesley:

Het zal een service zijn die je in principe naast je beveiliging neer zet, ik denk niet dat het heel veel nodig heeft, het zal vast wel keiharde hardware nodig hebben.

Morcel:

Hoe groter je netwerk hoe meer powerful de oplossing moet zijn

Wesley:

Gaan jullie dan als uiteindelijk toekomstplan een hardware oplossing zijn of zal het een softwarepakket zijn die je op je VM’s zal installeren?

Morcel:

We hebben dit ook overlegd met Valentin in Spanje en hij zei dat bedrijven er een hekel aan hebben als je met een server aan komt zetten van 3 bij 3 en een hele server kast moet hebben en dit en dat.

Wesley:

Ja maar, dit zijn wel devices die gigantische CPU gaan slurpen.

Morcel:

Ja zeker, de shadowHoneypot is het ultime product. Het simuleren van je netwerk en alles wat erin gebeurt maar, zo’n netwerk sniffer hebben we nu het idee dat we dat tussen je ISP en je bedrijfnetwerk kan zetten en daar data vandaan kan halen en door kan sturen naar het systeem.

Het zou plug and play moeten zijn, maar we weten nog niet hoe realistisch dat is.

Wesley:

Dat verschilt per netwerk denk ik.

Morcel:

Als je je netwerk pakt van je werk?

Wesley: 28:40

Hmmm, het verschil bij ons netwerk is dat wij heel publiek hangen op het internet. Wij routeren zelf via BGP wij zijn zelf ook een BGP endpoint. Dus bij ons zou dat lastig gaan.

Maar als ik zou gaan vergelijken bij Saxion bijvoorbeeld, daar zou je dat heel makkelijk kunnen implementeren.

Morcel:

Daar heb je niet veel voor nodig, zoiets zou je er gewoon tussen kunnen hangen?

Wesley:

Nee, eentje in Deventer, eentje in Enschede een management overzicht voor beide dat zou mooi zijn en dan vanaf daar alles zouden moeten kunnen monitoren. Zoiezo niet alleen om de Honeypot (algoritme) te voeden maar traffic over het algemeen dat over je netwerk gaat zou heel interressant kunnen zijn om te zien. Je pakt het toch direct mee als je vanaf de ISP naar je router gaat dus dan zou je net zo goed dat ook kunnen laten zien. Je moet toch alles kunnen filteren en zien dus het zou mooi zijn als je dan een heel monitor dashboard zou hebben waarin je dat allemaal kan zien.

Als je dan ook zou kunnen filteren op bijvoorbeeld een top 10 lijst van pakketjes die van china zouden komen of rusland en welke poortjes het meest gebruikt worden en welke websites het meest gehit worden.

Morcel:

Gewoon een soort… dat je per land kan zien wat er vandaan komt?

Wesey:

Bijvoorbeeld, als je dingen ziet dat er heel veel traffic van een bepaald ip-adres zou kunnen komen of een reeks. Als je echt proactief naar aanvallen zou willen kijken.

Morcel:

En wat zou je eraan hebben als je ziet dat er bijvoorbeeld vanuit china wordt aangevallen?

Wesley:

Ja, daar zou je diep op in moeten gaan kijken, land zegt op zich niet zo heel veel aangezien toch iedereen VPN gebruikt die kwaad wil. Bij Saxion was het ook zo dat wij alleen bepaalde poorten open zetten vanuit China omdat er heel veel VPN’s werden opgezet om poortscans te gaan doen, DDOS aanvallen noem het maar op. En we lieten dan alleen dingen die eventueel studenten vnauit china konden zijn die dingen van saxion moesten checken. Maar voor de rest hebben we heel china keihard dicht gezet.

## Interview Bram Wenting

1.     Basic vraag: Wie ben je, wat is je rol bij SST?

Bram Wenting, mede eigenaar. Ben  binnen SST vooral verantwoordelijk voor aansturen developers en schakel tussen de programmeurs en developers. Daarnaast regel ik natuurlijk ook de bedrijfsvoering. SST is eigenlijk een uit de hand gelopen hobby, en groeit als kool.

2.     Is security een hot topic bij SST software? Zo ja, wat zijn je ervaringen?

Ja, en niet alleen intern en ook voor onze klanten. Wij beheren veel system waar veel persoonsgevoelige informatie wordt opgeslagen. Bijv Radboud Universiteit, daar monitoren we de statistieken van kinderen die daar zijn opgenomen. Dat is data die versleuteld is, 2F, dus we moeten ook monitoren van data ekken, zeker nu met de AVG op komst (Europese wet tbv internetsecurity/privacy).  Je moet laten zien dat je met AVG bezig bent. In die zin dus wel. Klanten gaan er ook naar vragen. Intern gedreven maar ook vanuit de klant, die licht ons ook in. SSL leveren wij ook niet, de klant moet daaraan voldoen – wij gaan geen onveilige websites opleveren.

3.     Hoe relevant vind je smartphone security?

Algemeen: Vind ik hel belangrijk, je ziet bij bank-apps, bij mobiel bankieren dat je met 1 pincode kan betalen, dingen worden makkelijker, alles ligt open, alles gaat via dat kreng. Dat is een risico, veel mensen zijn zich er niet van bewust en letten niet op wat ze installeren. Mensen hebben geen idee, het is ook een stuk bewustwording wat ontbreekt.

Google zou de Play Store net zoals Apple moeten behandelen. Binnen SST: ook belangrijk. Maar, wat ik moet zeggen is dat het personeel heel bewust is, die letten wel op wat ze op hun devices registreren. En we zorgen dat er op onze interne netwerk een bepaald aantal Mac-addresssen wordt toegelaten. Zo proberen we de pool van devices te berperken op basis van MAC-adres.  De IT’ ers worden vertrouwd.

Telefoons zijn van de mensen zelf, wij kopen alle licenties, de hardware is van de developers. Dan is daar ook geen gezeur over.

4.     Hoe belangrijk zijn mobieltjes voor SST Software en in wat voor vorm worden ze gebruikt?

We hebben vaste lijn op kantoor, hier zitten devs mee te bellen. Directie is degene die mobiel wordt gebeld. Voor de klant/SLA-bereikbaarheid hebben we onze losse simkaarten in de telefoon . Dat worden nieuwe Nokia’s waar je maar een paar apps op kan zetten, waar je verder niks mee kan.  Standby tijd als in batterij.

5.     Hoeveel mensen hebben een werktelefoon binnen SST software, en welk OS gebruiken ze?

Iedereen heeft dus een telefoon, niet van werk maar van zichzelf. Van 14 gebruiken er 4 ios. En 2 daarvan is directie, ook Android.

6.     Heb je wel eens ervaring gehad met Malware op smartphones?

Nee. We hebben heel bewust personeel.

7.     Hoe zorgt SST Software ervoor dat ze up-to-date blijft op security gebied?

Ontwikkeling dwingt ons. Wanneer er nieuwe frameworks uitkomen dan zitten daar zitten verbeteringen in qua security. Daarna lezen we veel hierover op internet, we willen snel op de hoogte is als er iets aan de hand is. Bijv. Security.nl, Tweakers.

8.     Welke informatie zou je graag zien op het Honeypot Dashboard?

Verbonden compuiters. Meer laten zien waar maliocous pakketjes vandaan komen. Pinpointen naar het individu. Knop met een blokkade.  Ze moeten dan naar de beheerder om het fixen. Het totaal is interessant, maar het malafide verkeer moet geblokkeerd worden.

9.     Hoe sta je ertegenover dat additionele hardware zou moeten worden geinstalleerd met ons systeem?

Boeit me niet zoveel, ik heb een IT bedrijf, we zijn het gewend, maar we doen een heleboel intern, dus het onderhoud wat daarbij hoort zijn we zelf verantwoordelijk voor. De maintenance op onze machines hangt dus af van wat we neerzetten.

10.  Heb je verder nog advies dat je zou willen meegeven?

Als ik alleen data kan ZIEN, is het een gimmick. Het werkt dus zo, een toevoeging zou dus zijn als de gebruiker automatisch geblokkeerd worden, dat scheelt servercapaticeit als het volumeus wordt.

Het zou interessant zijn om als je mensen op wilt voeden,  verplicten om iets te downloaden om op ons netwerk te kunnen. Dan kun je mensen echt monitoren, die terugkoppeling. Bijvoorbeeld op de telefoon. Hoe krijg je anders een berichtje dat je bijv. geblokkeerd bent?

## Coding

|  |  |  |
| --- | --- | --- |
| Sentence | Open coding | Axial coding |
| check what have been the most important Malware attacks in the last 3 years for android, and ID them. | Dashboard | Client knowledge |
| “Remember this recent big attack?” This triggers the company. | Dashboard | Client knowledge |
| Because people don’t realize how much information is stored in their phone. | Security, Awareness | Client knowledge |
| Door de forums van microtik te volgen, dus nog voordat het in het nieuws kwam. Je hebt vaak wel sites zoals securityNL, die wachten een paar dagen voordat ze het erop zetten om beheerder nog even de kans te geven te gaan patchen voordat het naar het publiek wordt gegooid. Want zodra het naar het publiek wordt gegooid gaat iedereen natuurlijk proberen om er misbruik van te maken. | Following forums and keep ahead of the news | Client Knowledge |
| Verbonden computers. | Infrastructure, Ease of Use | Dashboard |
| Pinpointen naar het individu. | Infrastructure, Ease of Use | Dashboard |
| Knop met een blokkade. | Security, Ease of Use, Infrastructure | Dashboard |
| Als ik alleen data kan ZIEN, is het een gimmick. Het werkt dus zo, een toevoeging zou dus zijn als de gebruiker automatisch geblokkeerd worden, dat scheelt servercapaticeit als het volumeus wordt | Security, Ease of Use, Infrastructure | Dashboard |
| Het zou interessant zijn om als je mensen op wilt voeden, verplicten om iets te downloaden om op ons netwerk te kunnen. Dan kun je mensen echt monitoren, die terugkoppeling. | Security, Infrastructure | Dashboard |
| always implementing new proposals that are state of the art. | Security | Dashboard |
| It would also be useful if you could introduce some Malware in the Honeypot. Something like a demo. | Infrastructure, Dashboard | Dashboard |
| keep it simple. | Infrastructure, Dashboard | Dashboard |
| Should monitor HDD, network. What are the processes that are running, what’s being used. | Dashboard, Ease of Use | Dashboard |
| Dingen worden makkelijker, alles ligt open. | Ease of Use, Security | Ease of Use |
| Google zou de Play Store net zoals Apple moeten behandelen. | Expectations, Ease of Use, Availability | Ease of Use |
| We zorgen dat er op onze interne netwerk een bepaald aantal Mac-addresssen wordt toegelaten. | Security, Infrastructure | Infrastructure |
| Hardware is unneccesary in this case. It’s a complexity you want to avoid. | Infrastructure | Infrastructure |
| Make it easily deployable, isolated. | Ease of Use, Security | Infrastructure |
| Companies are not employing people for this specific topic | Infrastructure, Expectations | Infrastructure |
| Op het moment nog niet echt beveiliging op het niveau om andere netwerken te verbeteren dan ze al zijn is nog te hoog gegrepen, met de ervaring die ik nu heb. Het gaat er nu nog voornamelijk om netwerkjes die basic zijn opgebouwd en steeds verder zijn gaan groeien en daar niet echt goeie richtlijnen op zijn toegepast waardoor alles er heel rommelig uit kwam. Denk maar aan hele netwerken in vlan01 gooien. NAT op NAT. Je kan het zo gek niet bedenken. Dat is allemaal 1 grote chaos daarbij moet ik verkeersstromen in beeld gaan brengen kijken welke firewalls er overal op staan. Dat uit gaan kammen, structuur in brengen en weer opnieuw in elkaar gaan zetten. | Networksecurity | Infrastructure |
| Je ziet eigenlijk ook nergens dat een netwerk van scratch gelijk goed wordt neergezet, er komen overal wel fouten in. En daar probeer ik dan structuur in te brengen. | scalable network | Infrastructure |
| Voornamelijk om te kijjken naar verkeer dat door bepaalde devices gaat dat eigenlijk niet hoort dus om dat een beetje dicht te timmeren. Of bijvoorbeeld een heel subnet dat toegang heeft tot een bepaalde server zonder porrtconfiguratie of wat dan ook. Kijken wat er nodig is en dan daar specifiek op gaan filteren. | Mapping the network, | Infrastructure |
|  | Control the firewall, | Infrastructure |
|  | Scan the network | Infrastructure |
|  | Unauthorized acces | Infrastructure |
| Ik heb dan in het verleden die ethical hacking cusus gedaan. En bij mij is er dan als conclusie uitgekomen dat als je iets gewild wilt aanvallen, ergens binnen wilt komen, bijna geen enkel netwerk is veilig. Je kan overal uiteindelijk wel binnen komen als je zou willen. | No network is waterproof | Infrastructure |
| Nee, zeker niet. | Waterproof | Infrastructure |
| Haha, vaak is het zo dat als het een keer fout gaat en alles komt boven water gaan ze het dicht proberen te timmeren als het dus eigenlijk al te laat is. Dat soort dingen zijn heel moeilijk om van te voren dicht te gaan zetten. | Acting reactively, difficult to predict. | Infrastructure |
| De meeste fouten gebeuren naar mijn mening door systemen niet te gaan patchen, verouderede systemen. Dan hoor je in het nieuws opeens, als je securitynl bijvoorbeeld volgt, hoor je in een keer dat er een nieuwe vulnirability is gevonden in een bepaalde software. De hele wereld gaat dan vervolgens dat proberen te scannen en kijken waar dat lek inderdaad zit en er dan massaal op gaan aanvallen. Dan kan je daar heel snel een netwerk mee platgooien als ze hun zaken niet op orde hebben. | Systems don’t get patched frequently, | Infrastructure |
|  | Weaknesses exposed in the news, | Infrastructure |
|  | Weaknesses get exploited | Infrastructure |
| Jazeker, haha. We hadden laatst op mijn werk heel veel microtik routers staan, zo’n 1500 en zoals ik eerder ook al zei is het slordig opgebouwd maar wel enorm gegroeid. Ze hebben bijvoorbeeld geen centrale update server. Nou wat gebeurde er, paar weken geleden was er een kwetsbaarheid gevonden, in een microtik router os versie waarbij bepaalde managementport; WinBox benaderbaar was en je er gebruikersnamen en wachtwoorden uit kon slopen. Dat was in een aantal software versies. Nou wij hadden ik denk wel meer dan 1000 routers, waarvan sommige aan het internet hingen, die hadden die verkeerde software versies. Wij hadden dan geen centrale software server dus wij konden alles 1 voor 1 updaten. Dan denk je toch van; Goh, hier had misschien wel eerder over nagedacht moeten worden. Maar goed dan gebeurt zo iets en wij zijn nu bezig met het implementeren van een centrale patch server voor de volgende keer als er weer een brakke software is. | Acting reactively  Software bug  Project after the incident | Infrastructure |
| We moeten ook datalekken monitoren, zeker nu met de AVG op komst. | Compliance, Security | Law Compliance |
| Je moet laten zien dat je met AVG bezig bent. | Compliance, Security, Expectations | Law Compliance |
| Ontwikkeling dwingt ons. | Security, Infrastructure, Compliance | Law Compliance |
| Daarna lezen we veel hierover op internet, we willen snel op de hoogte is als er iets aan de hand is. | Compliance, Security | Law Compliance |
| Usually for companies it’s being one step behind, because we’re preventing | Security, Awareness | Malware |
| try to extract all the features that you and from the data that you have | Machine Learning, Quality | Malware |
| to try to insert features that are not dependent of the rest | Machine Learning, Quality | Malware |
| The trend is that everything is moving to encrypted, but it’s more difficult to detect. | Trends, Security | Malware |
| is a second step. First you need to get the data and use it for learning process. If you use ML to ID what is malicious and what is not. | Machine Learning, Quality | Malware |
| One of the challenges is good accuracy with Machine Learning. | Machine Learning, Quality | Malware |
| Heb je dat wel eens gehoord? Dat je dan alsnog wel naar de helpdesk van die Ransomware kan bellen? | Ransomware, decrypt | Malware |
| Dan krijg je gewoon iemand uit Rusland aan de telefoon en die gaan je dan legit proberen te helpen om die betaling over te maken en dan gaan ze het voor je regelen. | Payment, Abroad | Malware |
| Ja, ja. Normaal gesproken wel. Er is wel een tijd geweest waarin dat niet gebeurde maar dat kwam omdat er een fout in de software van de ransomware zat. | Decrypt, faulty software | Malware |
| Ooh ja daar heb ik wel van gehoord ja. | Faulty software, recent attacks | Malware |
| Maar dat zal inderdaad wel gaan groeien, het zal me niks verbazen inderdaad. | Increase of ransomware on android | Malware |
| Or we can directly sell it. Subscription based for example | Marketing | Profitability |
| Zeker weten. Ja absoluut. Dat verkoopt zoiezo wel lekker. | Product, profit | Profitability |
| Het personeel is heel bewust, die letten wel op wat ze op hun devices registreren. | Ease of Use, Security | Staff knowledge |
| De IT’ ers worden vertrouwd. | Security, Compliance | Staff knowledge |
| Nowadays even the smaller companies have noticed it’s crucial for their services. | Security, Awareness | Staff knowledge |
| In talaia’s case we try to be agile. | Awareness, Security | Staff knowledge |
| I don’t recommend to use your cellphone for work. | Security | Staff knowledge |
| Heb ik nog niet gezien in ieder geval. | Laxity of companies, reactive attitude | Staff knowledge |
| Alles is veilig. (sarcasme) | Naïve | Staff knowledge |
| Iedereen gebruikt het dus het zal wel veilig zijn. (sarcasme) | Naïve | Staff knowledge |
| Ik werk bij Tedas in brummen, daar doe ik netwerkbeheer. Voornamelijk op het gebied van security, netwerk infrstructuur ontwerpen en zoeken naar verbeteringen. | Networkadministrator | Staff knowledge |
| Social enginieering doet al heel veel, daar is het grootste issue bij de meeste netwerken. Maar grotendeels is alles op puur hacking niveau ook wel te doen. | Social engineering | Staff knowledge |
| Veel meer traffic op bepaalde poorten zie je overal naartoe gaan, ja. | Newscompanies give administrators a couple days time | Client Knowledge |
| Het probleem heel vaak bij dat soort dingen is; Je kan het wel tegenhouden maar je moet ook verkeer ook wel toe kunnen laten dat wel ernaartoe hoort te gaan. | Increase in traffic | Malware |
| En dat is wel heel lastig, voornamelijk als je netwerken hebt dat je door het hele land heen hebt zitten. | Filtering good from bad is difficult | Malware |
| Uuhm, Ja dan moet je het goed hebben opgezet. Dan moet je VPN overal hebben geïmplementeerd. Maar dit is niet overal het geval dus sommige system hingen nog aan het netwerk dat wel kwetsbaar was en dan maar hopen dat ze niet gehit worden. In dat geval kon je de management poorten wel dicht gaan zetten en wanneer je erbij moest komen moest je iemand opbellen die het met de hand even open ging zetten. | The bigger your network the harder it is to filter | Infrastructure |
| Het is moeilijk | The better your infrastructure the better your network | Infrastructure |
| Ja. Het kan wel, maar bij dit netwerk is dat niet echt mogelijk geweest. Bij Saxion in het verleden kon dat wel. We hadden daar ook een next generation firewall ziten en daar kan je ook veel dieper in pakketjes pakken wat er nou aan de hand is en waar het vandaan komt, maar bij dit netwerk ij tedas is dat nog niet zo netjes geregeld maar dat komt er allemaal nog. | Seperating good from bad is difficult | Malware |
| Grotendeels werdt het al wel tegengehouden. Op het moment dat ze het netwerk op wilden komen werdt het al afgekaatst. Maar het gebeurde wel eens dat mensen op het netwerk zelf zaten en op een verkeerd mailtje klikten waardoor inderdaad bestanden geencrypt werden. | depends on your firewall | Infrastructure |
| Bijvoorbeeld ik vertelde je toen ook nog van die collega, die was niet zo heel erg snugger. Die werkte bij de eerste-lijn support en we hadden toen heel erg last van phishing mailtjes, ransomware. Maar zijn zoontjes deed een of andere ICT Opleiding MBO 2 of 3. En zei; Hey Pap, ik heb hier software gevonden die bestanden kan decrypten zonder probleem. | Bring Your Own Device, ease of acces | Ease of use |
| Hij had natuurlijk het licht uitgevonden want daar hadden wij nog nooit aan gedacht om bestanden gewoon te decrypten, probleem opgelost toch?! | Phishing mails | Malware |
| Dus hij zegt tegen z’n zoontje: stuur maar naar mij toe, ooh dat kan niet via de mail? Doe maar via wetransfer. | Inside out attack | Malware |
| Nou hij heeft die troep op z’n laptop gezet, op z’n werk waar hij dus ook meer toegang had omdat hij bij support werkte en toen werd een groot deel van het netwerk plotseling geencrypt. Rara hoe kan dat. | | Staff knowledge |
| Nee, dat hebben wij toen niet gedaan maar dat gebeurt soms wel. Daar houden de developers van die ransomware rekening mee. Hoeveel zou dat ongeveer gaan kosten (om het terug te zetten) en daar handen ze dan een prijskaartje aan zodat bedrijven gaan overwegen om te betalen. | | Staff knowledge |
| Ja, maar dat gebeurd over het algemeen wel. | Back-up | Infrastructure |
| Het is bijna niet te blocken. Ik heb gewoon alle medewerkers gevraagd, dat was toen ook een projectje van mij, als zij mailtjes van… Ze hebben eerst ook cursussen awareness training gehad; wanneer lijkt het inderdaad op een nepmailtje wanneer is het een phishing mailtjes en dat soort dingetjes. | Decrypt, backup | Malware |
| Als ze zo’n soort mailtje kregen, vroeg ik of ze het wouden doorsturen naar support en dat zou dan naar mij doorgestuurd worden. | Backup, decrypten | Malware |
| Ik ga dan in het mailtje kijken wat voor website erachter hangt als je op het linkje zou klikken en die websites heb ik allemaal op een blacklist gezet, die IP adressen. | Decrypt | Malware |
| Dat werd op een gegeven moment een mooie lijst toen dacht ik van; Goh er zullen vast wel andere mensen zijn die dat ook doen. Dus ik heb op internet gezocht naar hele lijsten van IP adressen waarvan men dacht inderdaad dat het daarvan daar kwam, dus die heb ik er allemaal bij in gegooid en op een gegeven moment na een aantal maanden kreeg ik dan weer een mailtje: Deze is ook malicious. Als ik er dan op klikte gebeurde er niks omdat het toch allemaal al geblocked was. Dus het hielp uiteindelijk wel heel veel maar dat heeft gewoon tijd nodig en dat moet je blijven bijhouden en dat is gewoon niet handig. | Phishing mail, blacklist | Malware |
| Daar waren wel oplossingen voor en hebben we naar gekeken. Toen dachten we dat gaan we kopen maar, dat is uiteindelijk niet gedaan vanwege beleid en bla bla bla. Mensen zagen het nut er niet van in. | Blacklist | Infrastructure |
| Want het was maar een paar keer fout gegaan. |  | Staff Knowledge |
| Dat was… Hoe heet dat ook al weer? Misschien weet jij dat nog wel, als je bij saxion in wou loggen op het netwerk werden sommige devices direct geblocked door een of andere service, maar daar ben ik de naam even van kwijt. Ik weet niet precies hoe dat werkte maar die keek als een soort virusscan op je device en op basis daarvan kreeg je dan wel of geen toegang tot het netwerk. | | Staff knowledge |
| Maar dat verkeer dat dan.. Je ziet al het verkeer van een IP-adres vandaan komen die die actie op die Honeypot doet. Wat doe je daar vervolgens mee? Zet je die op een blacklist of,,? | Costs, senior management | Infrastructure |
| Nou ik werk nu met microtik en dat is ook allemaal GUI gebaseerd en dat werkt gewoon logisch. In het verleden heb ik met GUI’s gewerkt en dat is gewoon compleet onlogisch en dan was het gewoon kut maar als het werkt is het zeker fijn. | Acting reactively | Infrastructure |
| Dat is wel mooi, dat zou cool zijn. Het zou super handig zijn als dit bestaat. | Quickscan | Dashboard |
| Voornamelijk als er malicious pakketen binnen komen dat we kunnen zien waar het vandaan komt wat het probeert uit te voeren, wat het doet inderdaad, dat hij dat in het vervolg gaat blokkeren en proactief tegen gaat houden vanaf de andere kant. | Blacklist | Dashboard |
| En dat je daarna ook echt verschil gaat zien in load op je firewalls. | GUI’s | Dashboard |
| Het zal een service zijn die je in principe naast je beveiliging neer zet, ik denk niet dat het heel veel nodig heeft, het zal vast wel keiharde hardware nodig hebben. | Good solution | Infrastructure |
| Dat verschilt per netwerk denk ik. | Source, destination, block, firewall load. | Infrastructure |
| Hmmm, het verschil bij ons netwerk is dat wij heel publiek hangen op het internet. Wij routeren zelf via BGP wij zijn zelf ook een BGP endpoint. Dus bij ons zou dat lastig gaan. | Powerful servers, service | Infrastructure |
| Maar als ik zou gaan vergelijken bij Saxion bijvoorbeeld, daar zou je dat heel makkelijk kunnen implementeren. | Powerful servers | Infrastructure |
| Nee, eentje in Deventer, eentje in Enschede een management overzicht voor beide dat zou mooi zijn en dan vanaf daar alles zouden moeten kunnen monitoren. Zoiezo niet alleen om de Honeypot (algoritme) te voeden maar traffic over het algemeen dat over je netwerk gaat zou heel interressant kunnen zijn om te zien. Je pakt het toch direct mee als je vanaf de ISP naar je router gaat dus dan zou je net zo goed dat ook kunnen laten zien. Je moet toch alles kunnen filteren en zien dus het zou mooi zijn als je dan een heel monitor dashboard zou hebben waarin je dat allemaal kan zien. | Network dependant | Dashboard |
| Als je dan ook zou kunnen filteren op bijvoorbeeld een top 10 lijst van pakketjes die van china zouden komen of rusland en welke poortjes het meest gebruikt worden en welke websites het meest gehit worden. | Network dependant | Dashboard |
| Bijvoorbeeld, als je dingen ziet dat er heel veel traffic van een bepaald ip-adres zou kunnen komen of een reeks. Als je echt proactief naar aanvallen zou willen kijken. | | Dashboard |
| Ja, daar zou je diep op in moeten gaan kijken, land zegt op zich niet zo heel veel aangezien toch iedereen VPN gebruikt die kwaad wil. Bij Saxion was het ook zo dat wij alleen bepaalde poorten open zetten vanuit China omdat er heel veel VPN’s werden opgezet om poortscans te gaan doen, DDOS aanvallen noem het maar op. En we lieten dan alleen dingen die eventueel studenten vanuit china konden zijn die dingen van saxion moesten checken. Maar voor de rest hebben we heel china keihard dicht gezet. | Logical GUI |  |
|  | Proactive |  |
|  |  |  |
|  | Deep traffic analysis |  |

1. The practice of hacking with malicious intent. [↑](#footnote-ref-1)
2. The interviews from the Field Research can be found in the Appendix [↑](#footnote-ref-2)