|  |  |
| --- | --- |
| Plan of Action  Honeyjar and Malware studies | Abstract  This document describes the background of this project, as well as being a clear summary of expected actions and responsibilities for all project members involved. |

# Version management

|  |  |  |  |
| --- | --- | --- | --- |
| **Version** | **Author** | **Date** | **Comments** |
| 0.1 | Morcel | 20-02-2018 | Added chapters |
| 0.2 | Alexander | 21-02-2018 | Added front page, as well as further defining of the problem definition |
| 0.3 | Morcel & Alexander | 26-02-2018 | Improved Feasibility and Risks and applied Moscow-method to Project boundaries |
|  |  |  |  |
|  |  |  |  |

Table of Contents

[1. Background 5](#_Toc506971345)

[2. Subject of research 6](#_Toc506971346)

[3. Goal and Problem Definition 7](#_Toc506971347)

[3.1. Problem definition 7](#_Toc506971348)

[3.2. Goal 7](#_Toc506971349)

[3.3. End-results 7](#_Toc506971350)

[4. Theoretical framework 8](#_Toc506971351)

[5. Tools of Research 9](#_Toc506971352)

[6. Organization 10](#_Toc506971353)

[7. Planning 11](#_Toc506971354)

[8. Costs 12](#_Toc506971355)

[9. Feasibility and risks 13](#_Toc506971356)

# Background

## Origins

The project originated from the University of Aalborg (AAU). There have been prior project based around a Honeypot system.

The project dates back to 2013 and was a master thesis project. In this first project a first architecture was developed and a first implementation was made.

The motivation the initiate the project was:

*“The need for good ground truth data to be used for research and training machines through machine learning algorithms.”*

While analyzing the network traffic the group realized that there were surprisingly few usable traces available. Because of this researchers still use old datasets such as the KDD’99 dataset from 1999. From this project onward a basic architecture has been created.

## C:\Users\Morcel\Dropbox\HoneyJar Business\Business\Honeypot System Figure.pngArchitecture

Since then the term HoneyJar has been used as a host for many different projects. The Honeyjar consist of three parts;

### The test environment

This environment is meant to look like a real network, not a virtual one. Furthermore it is possible to create virtual machines which can be created and taken down again, automatic installation of software and operating systems as well as an emulated internet environment.

### Containment

This part is created so that it is possible to connect to the outside but preventing harmful traffic to go outside.

### Analysis

Here data is collected for analysis.

These parts are all very complex and require refinement and improvement. So since 2013 this has been done on the initial setup. This setup consists of thirty old computers.

There have been several projects that were fairly successful. One project even got the Danish Tele Award in 2016 BRON HIER. The project group had analysed 300.000 pieces of malware. They managed to find patterns which differentiated malware from cleanware.

But to continue the research there had to be a better setup, thirty old pc’s wasn’t going to cut it anymore. It also often happened that a computer broke down and caused problems with experiments.

In 2017 a Danish foundation granted two new and very powerful servers to continue this research on. This is also the reason that new Honeyjar projects have been initiated.

These servers will be used for the Honeyjar project as well as white hat hacker training. Both the project as well as the training will require virtual machines and networks. Another grant has been granted for the training as well to create a Danish training platform for cyber security. Another perk of the new hardware is that there is a faster connection to the Danish Research Network.

Everyone except for the people from AAU are new to this project. These project members from AAU have worked on this project half a year prior to the rest and have made a basic simple android based honeypot system.

Their assignment was to improve cybersecurity. They took their research to Android phones since according to their research, people store a lot of sensitive data on their smartphones and cybersecurity on smartphones isn’t deemed urgent by the masses at this point in time.

Saxion has brought two business IT students to the project. Their job is to create a system that is appealing for the market. In what form can this be a product that will be appealing to a potential customer? Other tasks that these students are specialized in is:

* Different kinds of modeling, think of BPMN, Database models, Data Flow Diagrams
* Translating customer needs into IT solutions
* Experience in working within a multidisciplinary group
* Facilitating the team where needed

The students from AGU and UTP will both improve the current system as well as do a network analysis to determine what is and isn’t malware. With this analysis they will be able to create a basic machine learning algorithm.

# Problem Definition

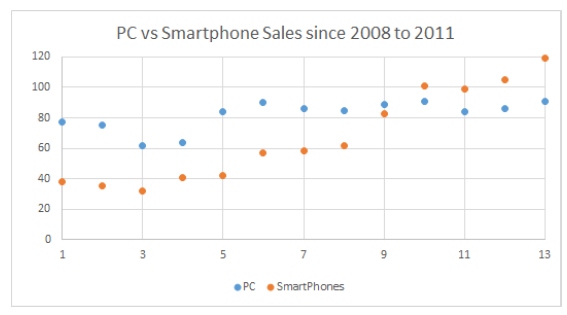
In this paragraph the group will describe what the reason for the project is as well as the desired end result when the project is finished. This will be accomplished using research questions. These questions will divide the project into smaller chunks. When all the sub questions have been answered the main issue will almost automatically be answered.

## Statistics

Companies as well as regular consumers have a lot of sensitive data stored on their smartphones, for example (but not limited to):

* Credit card data
* Compromising photos
* Passwords
* E-mails
* Customer data

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 to the market, however it seems that Android phones, while vastly gaining popularity over PC’s, are still relatively under protected to Malware attacks. This is represented in the graph below, which is derived from a survey done by the project members from AAU. (BRON HIERR)



To exploit the fact that the market hasn’t really focused on security on Android phones yet, the project members from AAU created a basic Honeypot system.

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. This Honeypot system in its current state is still very basic and needs improving.

AAU has created a figure of how the honeypot system should work, this is illustrated in the following figure.

The three computers in the example will be running an emulated version of Android, making the emulation seem as realistic as possible, i.e. emulating human behavior. Once these systems are infected through the internet, the malware will be contained, and analyzed to determine what the category of the Malware is.

## Goal

The goal of this Honeyjar is to provide intelligence on the nature of the Malware that is captured and contained in the quarantine. Furthermore, based on the intelligence that is gained in containment, the Honeyjar algorithm has to show improvement in recognizing Malware.

## End results

At the end of this project the following will be finished and accomplished:

1. A functional Honeyjar system that shows improvement in recognizing Malware through Machine Learning
2. A Business Plan to describe the Business aspects of the Honeyjar and how to make the system profitable
3. At least 1 Android Virtual Machine that functions as bait for Malware

# Project Boundaries

The following paragraph defines what the project will and will not deliver. The purpose of this paragraph is to define the scope of the project, and making clear what should not be worked on during the project, maximizing efficiency. For this, the Moscow-method is used.

## Method

The Moscow-method is a way of prioritizing requirements. Moscow stands for:

1. M: Must haves
2. S: Should haves
3. C: Could haves
4. W: Won’t haves

*Must haves* describe the essential requirements of the project. These are the core requirements, and must be attained at all costs. The next step is *Should haves*, which describes requirements that are deemed very useful but aren’t integral to the success of the project. *Could haves* is another step below that, and can be seen as requirements that can be added if there’s time left. *Won’t haves* describe the requirements that will not be implemented.

Using this method sets a clear scope.

## Applying the method

|  |  |
| --- | --- |
| Must haves | Should haves |
| Functional Machine Learning algorithm  Virtual simulation of Android phones  An improved version of the already known ML: algorithm  Self-categorizing functionality of Malware  Clear Business plan | Dashboard  Virtual Android phones that are as ‘’human’’ as possible |
| Could haves | Won’t haves |
| Web-application  Demo for potential customer | Antivirus functionality  Apple iPhone-functionality |

# Tools of Research

In this chapter will be described which tools the project group will use to gain the information necessary to conduct the project.

## 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

## Field research

For information that needs to be actual, field research will be performed. Information will be collected through interviews with companies focused on security, as well as other relevant people which are to be selected during the course of this project.

Through the conducting of interviews information will be collected that will be summarized through axial coding, making the information usable and measurable. This information can then be put into a program like Excel to showcase which results of the interviews are the most recurring, thus giving the project group insight in what is considered relevant.

## Sharing of knowledge

To make sure everyone in the group is up-to-date with the progress that’s being made, during every virtual meeting all project members share what they have been researching the week prior. This keeps every project member informed, and allows for room for discussion.

# Organization

In this paragraph the project group is defined and elaborated upon.

The current project group consists of nine people coming from four different universities.

Everyone in the group are equal and have the same obligations. The group works together and makes sure everyone is able to do their part in the project. Since everything is intertwined it is important the group keep communicating and gives each other feedback where possible.

As said in background the project group consists of nine different people spread across four different universities. There will also be a supervisor which is closely involved with the project. On top of that every university has a teacher that will guide the student throughout the process.

|  |  |  |
| --- | --- | --- |
| **Function/role** | **Name** | **Tasks** |
| Supervisor/AAU Teacher | Jens Myrup Pedersen | Supervising/guiding the whole group during the project |
| Supervisor | Etto Salomons | Supervising and guiding the business students during the project |
| Supervisor | Mehmet Şükrü Kuran | Guiding Project member |
| (UTP Teacher) |  | Guiding Project member |
| Programmer | Jacob Vejlin Jensen | Improving the Honeypot system |
| Programmer | Peter Bolstad Møller | Improving the Honeypot system |
| Programmer | Daniel Britze | Improving the Honeypot system |
| Programmer | Robert Nielsen | Improving the Honeypot system |
| Programmer | Magnus Stensli | Improving the Honeypot system |
| Programmer/Networking | Ahmet Türkmen | Improving the Honeypot system and doing network analyses |
| Networking | Anna Switala | Improving the containment zone of the Honeypot |
| Business IT | Alexander Pluimers | Creating a vision as well as a business plan for the product |
| Business IT | Morcel el Ouahbi | Creating a vision as well as a business plan for the product |

The following table contains contact information from everyone who is involved during the project.

|  |  |  |
| --- | --- | --- |
| **Name** | **Phone number** | **E-mail address** |
| Jens Myrup Pedersen | +45 99 40 87 71 | jens@es.aau.dk |
| Etto Salomons | +31 6 22 49 06 77 | [e.l.salomons@saxion.nl](mailto:e.l.salomons@saxion.nl) |
| AGU Teacher | 05325210283 | sukru.kuran@agu.edu.tr |
| UTP Teacher |  |  |
| Jacob Vejlin Jensen | +45 25 14 09 90 | [jvje17@student.aau.dk](mailto:jvje17@student.aau.dk) |
| Peter Bolstad Møller | +45 26 95 30 33 | [pmolle17@student.aau.dk](mailto:pmolle17@student.aau.dk) |
| Daniel Britze | +45 28 64 31 17 | [dbritz17@student.aau.dk](mailto:dbritz17@student.aau.dk) |
| Robert Nielsen | +45 28 76 82 75 | [rnni17@student.aau.dk](mailto:rnni17@student.aau.dk) |
| Magnus Stensli | +45 51 96 45 60 | [mstns17@student.aau.dk](mailto:mstns17@student.aau.dk) |
| Ahmet Türkmen | +90 541 204 37 48 | [f.ahmet.turkmen@icloud.com](mailto:f.ahmet.turkmen@icloud.com) |
| Anna Switala | +48 782 33 55 97 | [annswi004@utp.edu.pl](mailto:annswi004@utp.edu.pl) |
| Alexander Pluimers | +31 6 44 16 33 38 | [314831@student.saxion.nl](mailto:314831@student.saxion.nl) |
| Morcel el Ouahbi | +31 6 21 35 29 90 | [423819@student.saxion.nl](mailto:423819@student.saxion.nl) |

# Costs

This paragraph will discuss the costs associated with the project and any potential gain that this project can bring. It is intended to see if the project is justified to continue.

The current HoneyJar project has a funding of €40.000 which has been granted from the A Danish Research Network. So the €40.000 will be taken used as the whole funding for the entire project.

When this project is concluded in half a year ideally everything has been done. Realistically this won’t be possible since the architecture that is currently standing is very basic and needs a lot of improving. When this is done machine learning has to be implemented. Since the project members that are going to work on this aren’t familiar with it this will take a lot of time as well.

While it is still unclear on what exactly will be delivered, for now it seems like it’s going into the direction of an antivirus system with machine learning incorporated in it. So this is what will be used in the following example.

Whenever an antivirus is created there has to be a good understanding of the operating system it will run on, in this case that will be android. The group has to be up-to-date with malware trends so that it can update the system accordingly. Even with machine learning in place it is important to give updates where needed. If machine learning will be the only way of detecting malware, new malware might slip through the cracks. Also a proper programming language must be mastered.

So now the following has been established:

* Architecture needs ***a lot***of work
* Machine learning is new to the current project group
* There probably won’t be enough time to finish the created vision

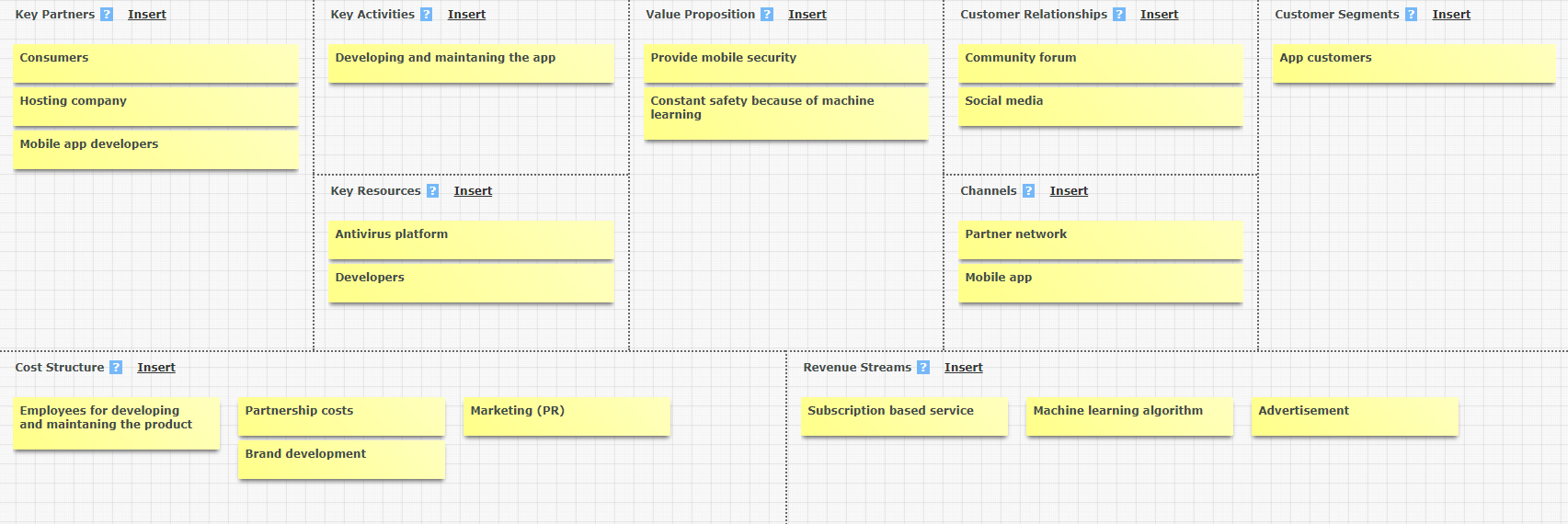
Even though there probably won’t be enough time to finish an end product it is still possible to create an estimate on the revenue that the potential finished product will make.

So let’s say that this product will cost €40.000 to make. With this €40.000 a basic product will be made to show the basic functionalities. Further development will require more money of course. Also for this example assume that the product will be sold as a service to consumers who will pay an average of €60 per calendar year.

So right now the product is a SaaS with a subscription for consumers. See the following business model canvas. €40.000 is nothing to create an app this size but, it will be the baseline to create the fundamental features.

Labor costs will be included in the €40.000 for now. As said previously this will include all the work to create a basic system. However with this basic system money will not be earned, yet. For this future development is needed. Since it is hard to predict the future an assumption has to be made on how long this development will take.

Now that a product has been sketched an assumption can be made about the money it will cost and bring in once it’s finished.

**Business model canvas:**

Toelichting kopjes hier door Alex

Labor costs will be included in the €40.000 for now. As said previously this will include all the work to create a basic system. However with this basic system money will not be earned, yet. For this future development is needed.

Now that a product has been sketched an assumption can be made about the money it will cost and bring in once it’s finished.

|  |  |  |
| --- | --- | --- |
|  | Estimated € needed |  |
| Basic system development | € 40.000 |  |
| Development for a marktable product | € 300.000 |  |
| **Total development costs** | **€ 340.000** |  |
|  |  |  |
|  |  |  |
| Future improvements and support costs (annually) | € 120.000 |  |
| Partnership costs | € 31.250.000 | 25% of profit |
| Marketing | € 12.500.000 | 10% of profit |
| Brand development | € 12.500.000 | 10% of profit |
| **Total** | **€ 56.370.000** |  |
|  |  |  |
| **Gain money (annual)** |  |  |
| Subscription | € 120.000.000 | avg €5p/month with 2million users |
| Machine learning algotrithm | € 5.000.000 | avg €5000p/month with 1000 businesses |
| **Total** | **€ 125.000.000** |  |
|  |  |  |
|  |  |  |
| **ACTUAL PROFIT PER YEAR** | **€ 68.630.000** |  |

Development for a marketable product has a price of €300.000 because the assumption has been made that it’ll be developed by people that’ll work on this app full time (40hrs/week) for a period of half a year to a year.

# Feasibility and risks

## Feasibility

Feasibility describes what the project needs to function, i.e. money, infrastructure, materials, expertise, and time.

### Money

For this project money is not an issue as all intercultural meetings are accounted for by EPIC.

### Infrastructure

One of the pillars of this project is that it’s intercultural, so project members will work together from their home countries. There are weekly online meetings where progress is discussed. Other than that there are plans to meet in a city somewhere in Europe at least one more time during the project.

### Materials

For setting up the server on which the Honeyjar is hosted, the University of Aalborg provides free server space. Other than that all project members will be using software that is either freeware or accounted for through a student license.

### Expertise

As the project group consists of 9 members, there are a different expertises; business, networking, and programming. These different expertises are useful for providing different approaches to the project.

### Time

All project members are expected to work on the project every day, ranging from a few hours every day to full-time. Even though the project group is the largest among the EPIC groups, the project is big and ambitious, so maximal effort is required.

## Risks

### Different deadlines

As all project members work on this project in a different year and/or in a different format (i.e. the Business project members are doing this project for their Minor studies, while the Danish group is doing this for their first year of school), the deadlines for each project member differ. This opens up the possibility that project members are going to follow their own agenda as opposed to working towards a common deadline as a team. To prevent this, the group created a planning with each deadline carefully noted.

### Miscommunication

Due to the intercultural nature of this project, the project members expect there to be miscommunication to a certain degree and cultural differences. The project members try to avoid this by setting up the following ground rules:

During meetings:

1. Give each other the space to talk, no interrupting
2. If something isn’t clear, ask
3. One person is assigned to take notes during the meeting

In general:

1. A global planning has been set up in Google Calendar
2. Meetings at least once a week
3. Subjects to discuss during the meeting must be submitted 2 days prior
4. Project members are expected to be able to reply in a window of 24 hours

|  |  |  |
| --- | --- | --- |
| Risks | Causes | Measures |
| *Financial* | | |
| End product not feasible within set investment range | Underestimation of funds needed for end product | Adjust investment range accordingly |
| *Organisatorial* | | |
| Project members aren’t living up to the contract | Various, could be anything ranging from illness to lack of interest, to miscommunication | Communicate with said project member, ultimately supervisor |
| End product is not deemed useful by businesses | Misunderstanding of what companies are looking for | Adjust requirements of end product to make it interesting for businesses again |
| Interested businesses already have a Honeyjar solution in place | The business has a functional Honeyjar system | Add requirements that make our Honeyjar system stand out |
| Failing to define why businesses should use our Honeyjar | Scope too large | Define a set of key selling points for businesses to get interested |
| *Technical* | | |
| Usage of personal data | Risk of law infringement, depending on country where the data is collected from | Only use personal data from countries where it’s usage in a Honeyjar is legal |
| Malware not attracted to the Honeyjar | Lack of relevant data to attract Malware | More research needs to be done on what attracts malware |
| Machine learning giving false negatives | Regular data being identified as malware | More research needs to be done on how to correctly authenticate data |
| Server outage | Any possible reason | Secure data on private back-up curated by Daniel |

# Appendices

## C:\Users\Morcel\Dropbox\HoneyJar Business\Business\Planning.pngAppendix I: Planning