RAMS

Reliability, Availability,
Maintainability, and Safety

This is the Title of my Thesis

Your Name

August 2014

PROJECT / MASTER THESIS
Department of Production and Quality Engineering
Norwegian University of Science and Technology

Supervisor 1: Professor Ask Burlefot
Supervisor 2: Professor Fingal Olsson

Preface

Here, you give a brief introduction to your work. What it is (e.g., a Master's thesis in RAMS at NTNU as part of the study program xxx and...), when it was carried out (e.g., during the autumn semester of 2021). If the project has been carried out for a company, you should mention this and also describe the cooperation with the company. You may also describe how the idea to the project was brought up.

You should also specify the assumed background of the readers of this report (who are you writing for).

Trondheim, 2012-12-16

(Your signature)

Ola Nordmann

Acknowledgment

I would like to thank the following persons for their great help during . . .

If the project has been carried out in cooperation with an external partner (e.g., a company), you should acknowledge the contribution and give thanks to the involved persons.

You should also acknowledge the contributions made by your supervisor(s).

O.N.

(Your initials)

Summary and Conclusions

Here you give a summary of your your work and your results. This is like a management summary and should be written in a clear and easy language, without many difficult terms and without abbreviations. Everything you present here must be treated in more detail in the main report. You should not give any references to the report in the summary – just explain what you have done and what you have found out. The Summary and Conclusions should be no more than two pages.

You may assume that you have got three minutes to present to the Rector of NTNU what you have done and what you have found out as part of your thesis. (He is an intelligent person, but does not know much about your field of expertise.)

Contents

Chapter 1

Do bots, make click, text and behavior metrics irrelevant?

1.1 What are bots and how are they used?

In general, bots are types of software which run automated tasks(which are repetitive and allows for the program to be faster at doing some tasks than what normal humans would be able to. There are many types of bots, for example gaming bots which can be programmed to efficiently react faster than what a human would be able to on certain events in the game, thus making the humans better players. Others examples include auction bots, which hunt for bargains. Ebay went to court in 2000(Rosencrance (2000)) in order to stop this type of behavior, the federal courts in turn then decided to block Bidders Edge and their bots from accessing Ebays API.

In the US 2010 mid-term election, social bots were reportedly used to influence the support of some candidates while effectively slandering others by tweeting thounsands of tweets which pointed to websites which contained fake news about the other candidate (Ratkiewicz et al. (2011)). This example highlights the potential seriousness of social bots.

It is not only in elections these bots have a great influence, during the aftermath of the Boston bombings it was noted that social media(and bots) had a great effect on getting the message out there(Cassa et al. (2013a)), but there was also a negative effect which bots attributed on Twitter by re-tweeting peoples' unverified accusations or even checking out the credibility of the source, causing more hurt than good.

Furthermore bots are used to mimic human activity, in for example service applications which appear to be human interaction but is instead a bot which for example tries to ”help” you based on keywords. Bots are used in many areas, but are probably most known for their maliciousness in areas such as spam, bandwidth-thiefs, scrapers, worms and viruses as well as as nodes in a larger scale bot-nets is used for example for DDOS. These bots can be part of bigger bot-nets in order to for example generate revenue through for example click fraud.

The observers article on fake traffic Holiday (2014) describes the problem and its extent:

> A recent study by comScore found that 54 percent of display ads shown in thousands of campaigns between May 2012 and February 2013 never appeared in front of a human being. Rather, the traffic came from bots. As an advertiser, this would be like buying a billboard you were told was seen by thousands of cars a day only to find out that was because the billboard sat next to the assembly line at a Ford plant. Sure, that’s a lot of cars, but there’s no one in them.
>
> In fact, the system is so broken that, for some publishers, knowingly buying traffic that comes from bots is part of their business model. An anonymous publishing executive, who claimed to be buying up to $35,000 worth of traffic per day, recently told Digiday that for publishers running an arbitrage model, all that matters is profit; quality of traffic does not factor into the equation.
>
> Holiday (2014)

This quote from comScore highlights a important point in today’s society. How does click metrics influence us, and how are they used? Are they useless, can they be used to measure something even though results most likely have been contaminated? and how does this type of behavior influence the buyers decision? These and some others are the question I will answer during this chapter.

I will also look into bots in social media, how they interact and how they spread in order to seem like normal users as well as interact as normal users.

1.2 What is social media statistics, and how is it utilized?

With the rise of social media use, social media marketing got a bigger foothold, allowing marketeers to interact with their buyers to a bigger degree, and it allowed the buyers to interact with their brands. Following this trend, many companies have big social media presences. Social media statistics allows the companies to closely monitor and narrow down their demographic and easily make specific content for a specific demographic. Furthermore, it allows for the company to get free publicity through what is called ”organic reach” (getting exposure through users' activity in order to generate more clicks from users' contact) as well as for example paid reach(ads sent to a predefined demographic) Social media statistics allows for all of this to happen, as Singh (2013) mentions:

> Social Media statistics about audience likes and dislikes makes it plausible to employ ”push marketing”-techniques to target audiences with advertisements that are relevant to their interests. Also, parameters such as click through rates or CTR can further quantify the success (or failure) of an advertising campaign.

This type of push marketing, utilizes the information the users' themselves have provided to the mother service(e.g. Facebook), like for example metadata which gives a certain characteristic which allows them to be targeted by the broad filters which services like e.g. Facebook uses.
Businesses also utilizes what is called pull marketing, which is a little more subtle, e.g. referrals, social competitions(”like and share” and get the *chance* to win X”) these methods can generate traction in social media, and be spread by word of mouth.

But most notably, these types of metrics can be abused by bots.

An example of why metrics may be come obfuscated

What happens to social media analytics in a business environment when bots enter the fray? Depending on the function of the bot, its behavior can vary greatly. If the bot is what is called a crawler, the bot will crawl through real peoples feed and try to mimic their behavior, and then crawl and gather more data from other people in the first peoples feed and so on. This type of activity has atleast three types of repercussions, it firstly generates fake interest for other companies' social media analytics, it hides their original intention(liking ”this page”) and it likes ”this page” and generates fake revenue and publicity.

A good real-life example is Narang (2015), which had a set of inter chained accounts posing as real accounts in order to spread their own links about some dubious diet pills. The accounts stole meta data from real accounts. Instead of using compromised accounts to tweet spam links, they were using accounts that impersonated brands and celebrities. Symantec goes on to describe how they defined three types of accounts involved in this scam:

- **Mockingbird**: Used real data from real celebrities for impersonating these individuals
- **Parrot**: Fake accounts using stolen tweets and photographs of real women
- **Egg**: New users with no set avatar

Mockingbirds have the goal of promoting the weight loss tricks. These mockingbird-tweets would get thousands of likes from Parrot-accounts which spiders through the real accounts of the mockingbirds. Parrots then follow any and everyone in the hope that users will follow them back because they are using avatars of attractive women, a tactic that has proven very efficient. The Parrots have real content that they post each day which is fake, an not only the content of the Mockingbird, in order to seem more real. These tweets are usually stolen from real accounts in order to seem real. The parrots will also engage in discussions and post ”reviews” of the diet pills in order to make the diet pills seem more real while the egg accounts just inflate the like counts in order to make the mockingbirds as well as the parrots seem trustworthy. The egg accounts do not post any content, they just follow parrots.

The link provided by the mockingbirds seem real(with pictures of famous people). When

a customer orders a free trial, they register the credit card and subsequently lose their money.
The point of this example is to show how easily for example likes can be misguiding for a company that tries to establish themselves as a brand within social media.

1.3 Do these metrics influence buyers?

As we have seen, metrics can be obfuscated by many different factors; ”click farms”, bots, and other variables which can affect social media analytics. These factors may present in many different ways. In an article entitled ”Beyond likes and tweets: Consumer engagement behavior and movie box office in social media”, Oh et al. (2016), they researched how consumer engagement behavior(CEB) was associated with economic performance based on popular movies released in the US. The study found that CEB in Facebook and YouTube correlated with gross-revenue in opening-week movies. The study further concluded that CEB played a pertinent role in relation to future economic performance, so in some cases pure metrics can have great effects on the consumer itself.

Another good example of this is metrics in movies. In particular ratings on movies and Tv-series through IMDB(Internet movie data base), which is used to share users' opinions on films and tv-series in the form of ratings(top 250 movies, bottom 100) as well as recommendations and critics, both professional and user generated. In a paper entitled ”Judgement devices and the evaluation of singularities: The use of performance ratings and narrative information to guide film viewer choice”(Bialecki et al. (2016)), the researchers noted that the use of metrics such as ratings(moviegoers set a threshold in which served as a ”hurdle” in which movies they wanted to see had to pass). Research also indicated that if a movie had a high rating or were in the top 250 rating-list, it was likely moviegoers would see it because of the ratings themselves. Some subjects noted that the movie did not meet their expectations, but that they had to watch it because of the the ratings:

> *”*I rented this movie on the strength of the ratings and glowing reviews at this site [IMDb]. “Brilliant”,they said. “Dark and beautiful”, they wrote. 8.4 stars. Well, all I can say is, these people must have been on some serious drugs when [they] saw this totally inane movie. . .I give this movie 1 black hole.*”*
>
> Bialecki et al. (2016)

Another example of research done in this field is an article entitled ”The Influence of Social Media: Twitter Usage Pattern during the 2014 Super Bowl Game” which analyzed how people used twitter during the superbowl event specifically looking at consumer interaction with advertisement. The researchers used data mining in order to find correlations in data which may not be otherwise easily delectable. In the article(Shin et al. (2015)), the researchers asked two research questions: ”How does the overall number of tweets differ between a game day and non-game day?”Shin et al. (2015)) and ”What major topics in commercial related tweets were exchanged during the 2014 superbowl game?”Shin et al. (2015)). The data analysed for research question 1 (How does the overall number of tweets differ between a game day and non-game day?) indicated that Super bowl commercials created a buzz. The research also indicated that tweets about a commercial product accounted for 33% of total commercial tweets generated on superbowl day(february 2) as opposed to 0.64% and 9.48% on january 26 and february 9th. Further, more than half of tweets(37 of 49 tweets) registered mentioned Budweiser products or their commercial.

Even though these examples do not directly prove that consumers act on these type of data, one can draw the conclusion that user interacting might create a buzz and get your brand out there, thus leading to potentially more customers, subconsciously or otherwise.

In the case of IMDB-ratings, they seem to influence what movies people want to see, and therefore revenue of movies which are featured in a popular way. This is backed up in a article entitled ”An empirical investigation of user and system recommendations in e-commerce” about general statistics with e-commerce metrics Lin (2014) which could report that:

- 1% increase in user recommendation volume increases product sales by 0.013%.
- 1% increase in user recommendation valence increases product sales by 0.022%.
- 1% increase in system recommendation strength increases product sales by 0.006%.

1.4 How can the statistics become irrelevant?

So, what happens when these statistics become diluted by social media bots, does SMA become irrelevant? is there grounds to believe that bots dilute commercial interest rather than stimulate it?

An article released in 2013, named ”Is that a bot running the social media feed? Testing the differences in perceptions of communication quality for a human agent and a bot agent on Twitter”Edwards et al. (2014) investigated the claim that humans would not see the difference between a bot, and a human in a single newsfeed on Twitter. They used a sample of 240 undergrad students enrolled in a communication course at a large mid-western research university with subjects ranging from the age of 18 to 39 years old. The researchers then used two treatment groups(the twitterbot and the human twitter agent). Those who chose to participate were given a link to a secure webpage with the study. After consenting to the study, the participants were randomly given one of the two twitter pages. The twitter profiles were designed to appear as information provided by the central for disease control(CDC) about sexually transmitted infections. The pages were identical all except for one important detail - the author; in the twitterbot case it was clearly stated that it was a CDC twitterbot, in the other case it was stated that the author was a CDC scientist. The study concluded that

> The findings demonstrate that Twitterbots can be viewed as credible, attractive, competent in communication, and interactional, and might be an appropriate program to transmit information in the social media environment.
>
> Edwards et al. (2014)

This can mean that humans may not be able to detect the difference between humanoids and bots on the web, and one can therefore draw the conclusion that metrics may not provide the whole picture. In the next section we look at an example as to why.

Chapter 2

Approaches for Detecting Robots in Social Media

Management Summary

This is the management summary blablabla mhmhm...

2.1 Introduction

Social media is an important part of todays Internet. Of the approximately 3.4 billion internet users, about 2.3 billion use social media actively what means that almost 70 percent of all internet users use social media on a regular basis ins (2016). Especially in order to obtain information about events and the like, social media is often irreplaceable. For some people social media platforms even substitute real life personal contact to a large degree.

With this importance of social media platforms in the lives of so many people, these platforms get more and more interesting for people that want to benefit economically from them. These are for example companies that want to promote their products and services via those social channels. However, the problem here is that, due to their size, it is hard to disseminate a certain message on social media platforms. Like in many other areas of today's information technology, big data is a huge topic in social media.

This is one of the reasons, why social robots, in the form as we know them today, have been developed. As software robots are often simply called bot, we will use the term social bot or simply bot in the following synonymously for social robot. The propagation of such bots has reached a remarkable high level. In 2014, the social media platform Twitter, about we will hear more later on, stated in a report to the United States Securities and Exchange Commission, that up to 8.5% of their active user base may be social botstwi (2016). We will see in the upcoming chapters, that they are used to spread certain messages, hide others or simply to give the impression of a much bigger support for a person, thing or belief than there actually is. As we will see as well, though, social bots are also used to directly attack other, actual users directly, for example to steal their personal data.

First of all, we will take a look at the definition and history of social bots, in order to clearly define what we are talking about and where it originates from. We will then take a look at the online social media platforms that are important for this chapter, namely Twitter, Facebook and Renren, and give a short overview about their characteristics relevant here. Afterwards, we will motivate why social bot detection is actually important. In the main part of this chapter, we will introduce several social bot detection approaches and give examples for them. In the last section we will finally summarize our findings and give an outlook about the further challenges in this area.

2.2 Definition and History of Social Bots

This section will introduce the term social bot formally and give a short overview about the beginning and the development of this topic.

In order to be able to discuss social media bot detection, we need a clear understanding of what social bots actually are. For that, we use the definition given by Ferrara et al. in their article The Rise of Social Bots:

> ”A social bot is a computer algorithm that automatically produces content and interacts with humans on social media, trying to emulate and possibly alter their behavior.” Ferrara et al. (2015)

The root of of social bots, or just bots how we will sometimes call them here as well, can probably be found in the Turing test, developed by Alan Turing in 1950 Turing (1950). It involves three parties, two of which are human and one is a computer program. While one human is having a conversation with the software, it is the task of the other human to identify the program. If he is not able to do so, the software is passing the Turing test. This led to the development of a lot of so called chatbots, which just aimed to appear as human as possible in a conversation.

A rather famous and often cited example for such a chatbot is ELIZA, introduced by Joseph Weizenbaum 1966 in Weizenbaum (1966). It mimicked a psychotherapist and showed that – at least some kind
of – communication between a human and a computer is possible.

Since then, a lot of things have changed. Today, bots are a lot more than bare entertainment or proof of concept. With the triumph of the Internet and especially social networks like Facebook and Twitter, the possible use cases for social bots have increased dramatically. While they were initially mostly used to simply post content, today they are able to credibly interact with each other and even humans Boshmaf et al. (2013); Hwang et al. (2012). As we will see in the next section, nowadays bots are used to spread messages, for marketing and a lot more.

2.3 Important Social Media Platforms

In this section we want to give an short overview about the social media platforms, or online social networks (OSNs), that are relevant for this chapter. This is necessary in order to better understand the problems and attacks that will be presented later on. However, we will describe those networks only as detailed as necessary in this context, since a comprehensive explanation would go beyond the scope of this chapter.

The first and probably most important OSN that needs to be mentioned here is Twitter[1]. Twitter is a microblogging network which allows its users to broadcast short messages, so called tweets. Tweets can be seen by anyone that follows the tweeting account. Following is hereby a one-way process, that does not need any confirmation by the followed account. Thus, tweets can be seen as public. Furthermore, tweets can be retweeted. Retweets can be seen as a forwarding of a tweet in order to disseminate it: all followers of the retweeting account get the retweeted message. Tweets can be tagged with hashtags, that associate them with a certain topic or an ongoing discussion.

Another OSN that is important for this chapter is Facebook[2]. As on Twitter, it is possible to publish messages on Facebook. However, the visibility of those messages, or posts, can be adjusted. Usually, posts are only visible to users that are friends with the posting account. A friendship is here, in contrast to follow-relationship on twitter, bilateral: a user can send a friendship request to another account which has to accept it in order for the relationship to be built. A difference to Twitter that is important is well is, that Facebook accounts respectively profiles usually contain a lot more personal information than Twitter accounts do. Like it was the case with posts, the visibility of Facebook profile pages can be adjusted and usually they are only visible to accounts that are friends.

The last platform we want to mention shortly is Renren[3], which is a chinese OSN directed mainly at college students. Like Facebook it uses the concept of friendships and highly personalized profiles. It also allows to post messages that can be seen by friends.

2.4 Why is Bot Detection in Social Media Important?

Another point that is important to be looked at before we go into detail about the actual detection methods is, why social media bot detection is important at all. In this section we will argue, why this is a – especially today – important topic.

As already said above, social media platforms are nowadays an important component in the life of many people. For some, they even substitute real personal contact sometimes. Thus, it is important to ensure, that humans are still able to estimate what is happening in their social media surrounding and what can be seen as trustworthy – and what can not.

[1] https://twitter.com/

[2] https://www.facebook.com/

[3] http://renren.com

While the challenge of verifying facts or news from the Internet is not new, social bots add a new factor to this challenge. They can make a non-trustworthy source look trustworthy by simulating that it is highly popular.

An example for this, is the so called astroturfing. In an astroturfing campaign, an attacker tries to give the impression of a broad grassroot support for a certain person, belief or political position. A known political astroturfing case are for example the 2010 Massachusetts senate elections in the US. During these, a small number of Twitter social bot accounts produced a large amount of tweets that contained a link to a website that smeared one of the candidates. The tweets also mentioned users that had shown the same political position in beforehand and thus were likely to retweet. In a few hours the smearing website link spread rapidly what was even noticed by the Google search engine. That way, the website got promoted to the top search results for the name of the smeared candidate Mustafaraj and Metaxas (2010).

A similar approach can also be used to distract from potentially inconvenient or just unwanted facts or opinions and is then called smoke screening. By just flooding the platform with information it aims to draw attention away from the unwanted topic to a topic that suits the attacker more. The flooding information can hereby even be related to the topic that is to be screened. It just focuses on a component that is more pleasant to the attacker and withholds the inconvenient part. An example for this is given by Abokhodair et al. in Abokhodair et al. (2015) in the context of the syrian civil war. They analyze a social botnet on Twitter from April until December 2012. It made heavy use of this technique in order to cover up news about the civil war in Syria.

Another, more direct, example for an attack scenario is described by Boshmaf et al. They operated a network of social bots on the social media platform Facebook and successfully tried to build as many relationships as possible with real users. After this was done, they were able to extract data from the profiles of those real users that were not publicly available, like for example mail addresses or phone numbers. While they show that operating such a botnet for the sole purpose of data extraction might not be efficient for an attacker, they argue that this data can be used for more advanced attacks afterwards Boshmaf et al. (2013).

Besides all these specific attacks, which are carried out by bots built for this purpose, there are also simple negative effects from social bots that have not necessarily been built for an attacking purpose. The information in social media is often used by various entities. An example for this is the area of emergency response. By analyzing the information streams of social media platforms it is possible to estimate emergency situations and to take proper actions for decision makers. Cassa et al. for example, show in Cassa et al. (2013b) that information about the Boston Marathon attacks in 2013 were available on the social media platform Twitter more than five minutes earlier than the public health alerts, even though on this event there were already many first responders present. Another area where information from social media platforms is leveraged is the stock market. By monitoring the general mood on Twitter for example, it is possible to predict the market trend to a certain degree Bollen et al. (2011). A case can be made that this is also already done by traders. When in 2013 for example the Twitter account of the Associate Press was hacked and it posted rumors about a terrorist attack, the stock market crashed significantly Ferrara et al. (2015). It is therefor not hard to see that bot-caused information distortion on social media platforms – that does not even necessarily needs to be intended malicious – can have severe impact. Some social bots are built just to retweet and if they retweet false information or rumors they help to make this information popular. This happened for example also after the former mentioned Boston Marathon attacks, where false accusations were spread by such social bots Gupta et al. (2013).

Furthermore, social bots are often used by public persons in order to appear more popular and thus to gain influence or by companies in order to promote their products on social media platforms Stringhini et al. (2013). Here, social bot detection is necessary as well for ordinary users to be able to distinguish between real and bought support.

2.5 Social Bot Detection Approaches

In this chapter we want to introduce several techniques for detecting social bots. Based on Ferrara et al. Ferrara et al. (2015) we distinguish three detection approach classes.

The first category of detection approaches is based on social network information. They are also called graph-based, since they map users and their relations into a graph and then try to identify bots in the hereby obtained social network by means of graph theory.

Afterwards we will discuss crowd-sourcing based social bot detection approaches. They use actual humans to detect bots, assuming that the human ability to notice details in communication will make this an easy task.

The last category we want to elaborate on are detection approaches based on behavioral features. Mechanisms that make use of this approach try to observe behavioral patterns that are typical for social bots. By doing so they aim to distinguish bots from human users. Ferrara et al. (2015).

In the following sub sections, we will go into detail about each of these three approaches and illustrate them using real detection systems. It has to be noted though, that the borders between the individual approaches is not always very clear, so that some examples could also be mentioned in a different category. Many schemes for example, use some kind of graph theory. However, we try to categorize them by their core traits.

2.5.1 Based on Social Network Information

A term that is often used in combination with detection of bots by using social networks is sybil or the sybil attack. It was presented as a thread to distributed systems by John R. Douceur in Douceur (2002). In the specific context of social media platforms when conducting a sybil attack, an attacker creates a large amount of fake identities in a system to the point where these identities make up a considerable fraction of the system's whole user base. When this is achieved, the attacker can influence the whole system and control its contents to a certain degree. A sybil, sybil node or sybil account is therefor simply one of the fake entities, or, depending on the attack architecture, just a social bot. It is not hard to see that social bot detection can, more specifically, be viewed as a defense against the sybil attack.

The general proceeding of social network based bot detection approaches is rather simple. They map the users base of the social platform they aim to defend into a social graph, where a node is corresponding to a user and an edge between two nodes exists if there is a specific kind of relationships between the two respective users on the platform. The nodes can be hereby be distinguished in sybil nodes, respectively bots, and non-sybil nodes, respectively legitimate users. The goal of the detection approach is now, to identify whether a given node is a sybil or not Viswanath et al. (2010).

There are a number of proposed social network based sybil detection schemes like for example SybilGuard Yu et al. (2006), SybilInfer Danezis and Mittal (2009) or SumUp Tran et al. (2009). While they all have different assumptions and use varying algorithms to achiever their goal, Viswanath et al. show in Viswanath et al. (2010), that, at a high level, all of them work by them same principles.

Basically they can be viewed as graph partitioning algorithms, which partition a given graph into multiple disjoint subgraphs. As already mentioned above, ideally two subgraphs are assembled, one that contains only sybil nodes and one that contains only non-sybil nodes. Since a clear distinction is often hard to make, the approaches basically assign a rank to each node and decide afterwards, depending on several parameters, which ranks are classified as sybil and which as non-sybil. It is thereby obvious, that the ranking algorithm is crucial for the whole scheme. Though, of course, the ranking algorithms for the different detection schemes are differing, they generally have in common, that they base their rating on how tightly connected the respective node is to a known trusted node. Thus, they work by detecting local communities of nodes. In other words closely connected groups of nodes Viswanath et al. (2010).

It is not hard to see, that these algorithms are therefor easy to deceive. If an attacker is able to establish so called attack edges, connections between his sybil nodes and non-sybil nodes which are connected to a trusted community, it gets significantly harder to identify the bots. A common assumption is that these attack edges are hard to create Yu et al. (2006), which means, that legitimate users tend not to establish social network connections to social bots. However, Boshmaf et al. show in Boshmaf et al. (2011) that this assumption is to be questioned. They tried to infiltrate the social media platform Facebook with a large number of sybil accounts and tried to establish connections to real users. The average acceptance rate of their relationship requests came to about 20% and could be increased to 80% depending on how many indirect relationships between the sybil and the user already existed Boshmaf et al. (2011).

A well known example for a bot detection approach that is based on social network information is the Facebook Immune System (IMS) Stein et al. (2011). This system aims to defend the social media platform Facebook and its users not only against sybil attacks but also to prevent spam, malware distribution, phishing and so on. To achieve this goal, it takes a lot more actions than the above described general approach for social network based detection schemes. The IMS runs checks on every action performed on Facebook in realtime in order to give attackers as less time as possible to accomplish there goals and to react. It classifies these actions according to predefined policies and makes it thereby possible to judge them and the corresponding users.

An example for this could be a newly created Facebook account, that sends a lot of friendship requests. These are used in Facebook to establish relationships between users. A legitimate user starts sending friendship requests usually mainly to people he knows and vice versa, that is, people that are likely to accept his friendship request. If a lot of these requests are now declined, this could be a indication for the system that the sending user might not be legitimate. The IMS also makes heavy use of machine learning, to which we will come back in later on, and generates training data automatically in order to adapt to the fast changing attacks and user behavior Stein et al. (2011).

2.5.2 Based on Crowd-Sourcing

A rather straight forward approach for social bot detection is based on crowd-sourcing. In contrast to the above described social network information based approaches, a connection between bots and legitimate users is not a problem for schemes based on this approach – at least not a direct one. The basic idea of crowd-sourcing based social bot detection is, to engage actual human workers to study user profiles and subsequently to decide whether it belongs to an actual user or a sybil.

In Wang et al. (2012), Wang et al. presented a study on the effectiveness of this approach and introduce a sample for a crowd-sourcing based social bot detection system. For their tests they use sample data from Renren, China's most popular social media platform, Facebook US and Facebook India. They subdivide their human investigators, or simply testers, in expert workers, and crowd-sourced workers. Each of the testers is presented a number of social media profiles and has to decide, whether it is a real one, or a sybil. While the experts achieve a detection rate at about 90%, the crowd-sourced workers perform not as good individually. If their single votes are aggregated and used for a majority decision though, the results can significantly increased. If this is done for the expert workers, their results can be increased even more, too. A very desirable result of their studies is also that the false positive rate, that is the amount of profiles that are falsely classified as sybils, is in all groups very close to zero. This ensures, that the probability that legitimate users are accused to be social bots, what will probably offend them, is very low.

Wang et al. conclude, that it is very hard for sybil creators, to assemble social bots respectively profiles that are able to pass a ”social turing test” and that crowd-sourcing based approaches can perform very well. They proceed with introducing a general practical system that is illustrated in figure 2.1 [4].

The system is working like this: The social network respectively social media platform that is to be defended generates suspicious profiles. This can either happen through explicit reports from users or through automatically applied filters that detect abnormal behavior. The so obtained profiles are then checked by crowd-sourced workers, which are subdivided in accurate and very accurate. To establish this differentiation, some of

[4] Turkers are in this case the former mentioned crowd-sourced workers.

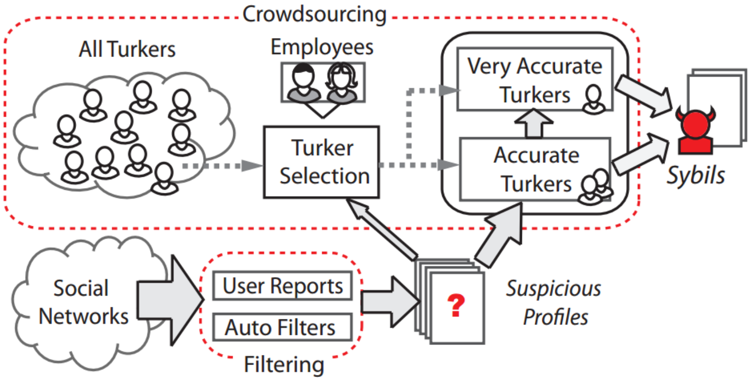


Figure 2.1: A crowd-sourced sybil detection system introduced by Wang et al. Wang et al. (2012).

the suspicious profiles are mixed with some profiles that are known to be sybils. They are presented to all workers and their results allow a classification and to filter out unreliable workers. In the actual checking process, the profiles are first presented to the accurate workers which make a majority vote. If their decision is not clear, the profiles are presented to the very accurate workers which once again make a majority vote. The authors claim, that a system like this is very reliable and cost effective.

Though such a social bot detection approach based on crowd-sourcing seems very promising at first sight, several problems become obvious on closer examination. First of all, this approach works optimal if every newly created profile can be reviewed in the above described way. However, young social media platforms will probably not be able or willing to pay for this bot detection scheme. A factor that may influence this could be also, that those platforms probably do not have huge issues with the social bot problem in their early stages. Once the platform has grown an the issue arises though, their is usually already a huge user base that has to be reviewed retrospectively. This is not optimal and may take a considerable amount of time.

Another point that mus be taken into consideration is that this approach might not be suitable for all types of social media platforms. While it might be well suited for platforms like Facebook or Renren, where profile pages can be customized a lot and contain usually plenty of information, other networks, like for example Twitter could be less appropriate. Generally it can be said that this detection approach is highly based on the information given in the users' profiles. If the information contained in the profiles is low, human investigators will probably not be able to distinguish as precisely as in the tests of Wang et al.

The last and probably most difficult issue that has to be mentioned is connected to this. Human investigators that are charged with distinguishing between bots and legitimate users have to have access to the profiles of the users. While those profiles are sometimes publicly available anyways, detailed profiles, which are more interesting for this approach, are, as discussed above, usually only visible for certain users. It is not hard to see the privacy issue that arises here, especially when keeping in mind, that the investigators are – at least to a great extent – only crowd-sourced workers, that can not be supervised as easily as ordinary employees Ferrara et al. (2015).

2.5.3 Based on Features

The last detection approach for social bots is based on features. Features hereby means observable characteristics of posted information, profiles and possibly everything else that can be associated to the account in question. Schemes that make use of this approach often also use machine learning algorithms. That are algorithms that are able to make decisions based on learned patterns. They basically consist of two phases. In the first phase, the so called training phase, the algorithm processes training data which is labeled with the decision that should be made on this piece of data. By processing a lot of training data, the algorithm learns the features that are indicators for a certain decision. In the second phase, real data is processed and the algorithm is able to make decisions based on the learned data. Thus, schemes that make use of this detection approach can be compared to anomaly based intrusion detection systems. In the social bot detection topic, the decision being made is usually whether a specific account is controlled by a bot or a human. Since it is not easy to make a clear decision in many cases, there are often multiple decision options that each express a degree of certainty.

A well known example for this kind of social bot detection is the Bot or Not? system, presented by Davis et al. in Davis et al. (2016). It is a publicly available[5] system that allows a feature based social bot detection for accounts on the social media platform Twitter. Bot or Not? was trained with about 31.000 account samples of both, human and social bot accounts. For classification makes use of more than 1.000 features that can be assigned to six feature-classes. In the following we want to mention these classes shortly:

- **Network** features reflect on the spreading of information, for example citations of

[5] http://truthy.indiana.edu/botornot

different users or the like.

- **User** features take the information given in the actual account into consideration. Meta-data, that is basically data about data, is especially relevant in this feature.
- **Friends** features are about the social relationships the account in question has.
- **Temporal** features analyze issues like the rate in which content is produced and the like.
- **Content** features are about wording characteristics in the texts produced by the relevant account.
- **Sentiment** features reflect on emotional aspects and are obtained by using specific algorithms Davis et al. (2016).

If a Twitter account name is entered in the Bot or Not? system, it checks all of those features and afterwards decides, based on the former mentioned training data, whether this account is a human or bot account.

Another social bot detection approach that can be classified as, kind of, feature based relies on detecting the coordinated behavior between social bots. Schemes that make use of this approach focus generally on the above described network, content and temporal features and often incorporate some graph theory as well. They usually aim at detecting attacks which try to distract from unpleasant facts or to distribute fake facts, like in the above described astroturf attack. An example for such a scheme is SynchroTrap, presented by Cao et al. in Cao et al. (2014). Again, comparable to anomaly based intrusion detection systems, SynchroTrap makes use of clustering algorithms. These algorithms try to group values, that are similar in some aspects into distinct sets.

The system monitors all user activity on a social media platform over an extended time period and monitors aggregated user actions. Then, the pairwise similarity between actions is determined and a hierarchical clustering algorithm is used to group users that show similar behavior at approximately the same time. If a cluster is growing too big, it can be assumed as malicious, since legitimate users tend to behave diverse Cao et al. (2014).

An issue with this detection method is though, that it is for example not able to detect bot accounts that show a mixture of bot and human behavior. Thus, the detection systems have to be developed further in order to keep up with the fast evolving social bots and their evermore sophisticated behavior Ferrara et al. (2015).

2.6 Summary and Outlook

In the last chapter, we firstly introduced the term social bot, defined it and looked at the history of robots in social media. Afterwards we gave a really short overview about Twitter, Facebook and Renren and motivated why social bot detection is important after all. In the main part of this chapter we discussed social bot detection approaches. We distinguished hereby social network information based, crowd-sourcing based and feature based approaches.

Social network based approaches model the analyzed OSN into a graph where users correspond to the vertices and relationships between the users correspond to the edges of the graph. Afterwards, an algorithm tries to make a decision whether a given node is a social bot or an actual user. The decision is hereby usually based on the connections the node and his surroundings have. As examples for this kind of detection approach, we mentioned SybilGuardYu et al. (2006), SbilyInferDanezis and Mittal (2009), SumUpTran et al. (2009) and the Facebook Immune SystemStein et al. (2011). A problem these systems have to cope with is, that they tend to assume, that bots can't establish relationships to actual, what emerged to be not true.

Next we elaborated on crow-sourcing based approaches. The basic underlying idea of these approaches is to engage actual humans to analyze given user profiles and to decide whether those belong to an actual user or a social bot. We discussed the paper ”Social turing tests: Crowdsourcing sybil detection” by Wang et al. Wang et al. (2012) which analyzes this approach thoroughly and introduces a general practical system. One of the issues this approach has to handle is that some social networks users don't have very detailed profiles what makes crowd-sourced bot detection very hard.

At last, we discussed feature based social bot detection approaches. They observe characteristics of OSN users and information published by them in order to decide – or give a tendency – whether a given account is a bot, or not. Often, they make use of machine learning algorithms. This means that, in a first step, they are trained with datasets in which it is known whether a profile is bot- or human-owned. Afterwards they are used to make decisions about real data. As an example we introduced the Bot or Not? system by Davis et al. Davis et al. (2016).

After analyzing the different detection approaches, it becomes apparent that a combination of different approaches yield good results. Observing the social graph and behavioral features of accounts together is for example a promising strategy. Ferrara et al. bring in the Renren sybil detector Wang et al. (2013); Yang et al. (2014) as a good example for these combined approaches Ferrara et al. (2015).

However, it has to be noted that not all robots in social media have to be necessarily evil. Some social bots provide useful services that many users don't want to miss. It is just important, that users are able to distinguish between other humans and bots for example in order to differentiate whether content they see is actually as popular as it seems.

It is safe to assume that the number of social bots will increase even more and that their methods and behaviors will get even more sophisticated. We expect an arms race between bot herders and bot detection mechanisms, like it was – and still is – the case with malware and malware detection software.

Chapter 3

Social Media attacks on mobile devices vs. attacks on computers

Management summary

We have seen an enormous increase in online social media in popularity during the past decade. It has taken over a lot of the networking people tended to do in real life earlier and it makes keeping in touch with a larger network of people a less laboursome task. We tend to trust a friend more than we would any random person on the street and this trust mechanism has been transferred to the cyber space. Cybercriminals, who seem to be mostly concerned with gaining wealth, political high ground, or are ideologically motivated, have shown a keen interest in utilizing these new networking platforms. They come from several different directions, such as attacking via vulnerabilities in hardware, OS, protocols, and apps, but they are also successfully exploiting the trust that exists between members of a social network. In this paper I am looking at the differences between the security topography for social media users, on a mobile device, versus that one would navigate when utilizing a personal computer. I have found quite a few differences and in other areas the situation is quite similar. The conclusion I reached was that it is, at least at present, much safer to use social media on a regular personal computer than on a mobile device, but that it also depends to a great degree how the user behaves security vice. The share of time people spend on social media networks on their mobile device versus on their computer is already 79

3.1 Hypothesis and methodology

My hypothesis is that it is easier to attack social media users via their mobile platform gadgets, than via their use of the same media on computers. I have weighed different factors and gathered information around the subject by reading published material saying something about usage pattern, platform weaknesses, distribution of vulnerabilities, and how the users time is divided between the two platform categories. I have looked at how software and hardware protection is being utilized the two cases and I will in the end try to conclude based upon the information presented. The subject of social has been moving so quickly, behavioural statistics changing drastically from one year to the next. This constant change makes it difficult to obtain standard peer reviewed research with the latest numbers. For this reason, I have partially leaned on numbers obtained from white papers published by some of the big commercial players in the information security market. These papers are recognized by the industry as reliable sources for such information.

3.2 Introduction

Social media attacks are increasingly common. People's usage pattern of social media has also changed dramatically during the past few years. An increasing amount of people's social media interaction is being conducted on mobile hardware platforms. Instead of using Facebook, LinkedIn and other media on the Personal Computer they may or may not still have in their home, they are now using the same media on their mobile devices, such as pads and smartphones. There are also new social media appearing, especially made for these mobile platforms, such as Instagram, which is made to share photos taken with your smartphone on the go. People now spend more time interacting on social media using their smartphone, than they do using their computer. According to comScore 79

3.3 Attacks on social media users in general

Some of the most popular social media providers today are Facebook, LinkedIn, Twitter, Google+, YouTube, Pinterest, Instagram, Vine, Snapchat and Reddit. There are myriads more and new services pop up all the time. While using portable devices to access social media, the user is both a possible target for the same types of attacks that we find being used against computer users and at the same time the limited resources of the mobile

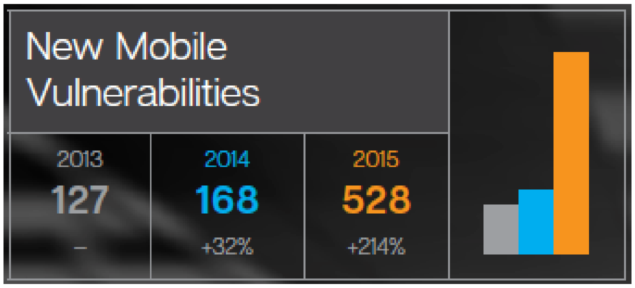


Figure 3.1: New mobile device vulnerabilities per year [10].

devices limits the possibility of using the same advanced mechanisms and software to stay safe [4]. Most attacks on the users of these media are economically motivated. There can be different paths to earning money on a social media attack. The reason for an attack can also be politically motivated or even nation state security concerns. I'll go through a few and compare possibilities on the different platforms. As the mobile devices gets ever more advanced and the number of apps, protocols and uses seem to rise towards the sky, it is no big surprise that the number of vulnerabilities follows suit as seen in Figure 3.1.

3.4 Specific attack vectors for computers

3.4.1 Microsoft Windows

Microsoft Windows has been the most widely used operating system and thereby represent the largest attackable population. It has also been regarded much easier to attack than the other common computer operating systems, because of the way it is built.

3.4.2 Apple OSX

Apples operating system was long regarded as the safest choice of operating system for a computer, but

3.4.3 Linux distros

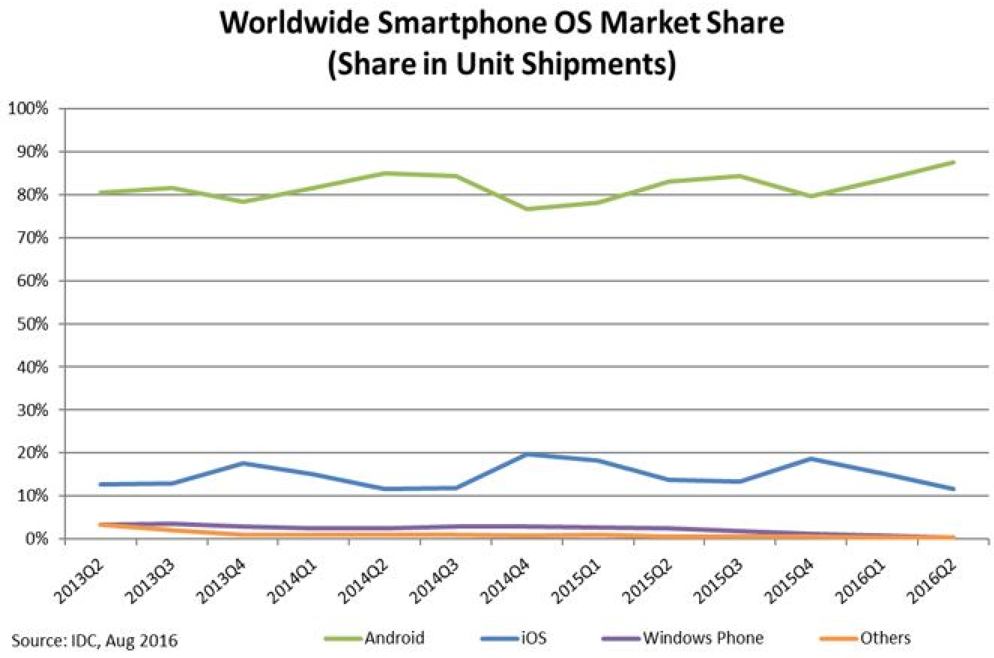
3.5 Specific attack vectors for smartphones

Android

New types of attacks are coming all the time. Cross platform attack on Android is being performed via Google Play in a web browser on an ordinary computer. When the victim logs on to the Google Play account on a computer, to install apps on a mobile device running Android, malware on the computer can steal the browser cookies for that session and use it to impersonate the user. It is then possible for the attacker to remotely install apps on the victim's Android unit. We can also see that the malware is becoming smarter and the sophistication level is rapidly reaching the level of malware for computers. Example are that they are now both obfuscating the code so as not to be detected by malware protection software that uses signatures, and malware that can check if it is running on a real device or a security company's emulator, thereby avoiding detection. Another weakness with the Android platform is that even if Google pushes out security patches to the makers of all the different devices as fast as they can, many makers take a long time to push these out to their end users [10].

3.5.1 iOS

Apples screening of apps, before letting them into their App store, is very strict, so one can generally be more trusting towards apps found there. Attackers of Apple's portable devices, running the iOS operating system, for the most had to rely on finding vulnerabilities or attack a jailbroken device to succeed (unlike Android). This is no longer the case. Several new ways of attacking these devices has recently come to light. Symantec mentions two examples of such attack techniques namely XcodeGhost and YiSpectre which can compromise an iOS device without using vulnerabilities or jailbreaking. The OS distribution in the market means that attackers probably concentrate on Android to get the highest return for their effort. In fact, Kaspersky Lab and INTERPOL has made a report that states that an estimated 98.05



Source: IDC, Aug 2016

Figure 3.2: According to IDC (IDC, Aug 2016) the market share for the Android smartphone OS in August 2016 was 87.6% And for iOS 11.7%.

3.6 Social media attacks directly on the given platform

As we have seen there are many ways of attacking social media users/accounts via the platform they are using the media on. There are however a lot of ways to attack via the social media itself, without depending on vulnerabilities in the hardware or software. The attacks simply rely on the many vulnerabilities found in the users, so to speak. These kind of attacks are usually categorized as

3.6.1 User behavioural aspects

3.6.2 Hardware and software

The way we use these mobile platforms will largely effect how safe we are from attacks. A lot depends on whether the user installs some kind of antivirus/antimalware program on their device and also how the user thinks about security, compare to when using computer. Many users leave the Wi-Fi on all the time. It is convenient to have your device connect automatically to all the networks you use, but this convenience comes at a high cost security wise. It is increasingly easy to attack these users by setting up a portable “Wi-Fi-box” that answers the unknowing smartphone owner’s device’s call for a named network,

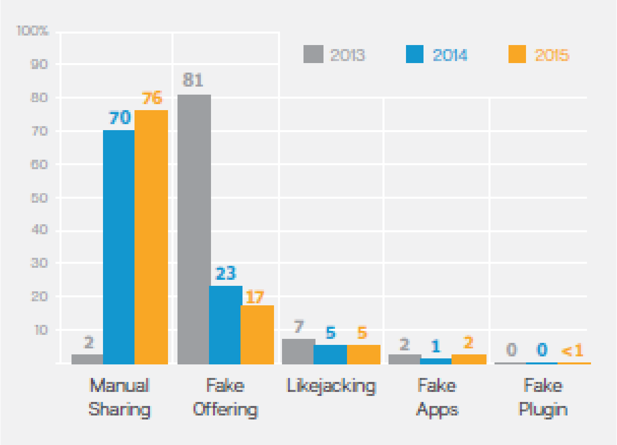


Figure 3.3: Social media network scams by popularity [10].

saying “Yes, here I am, please connect to me”. Such devices can be bought readymade on the Internet; an example is the Wi-Fi Pineapple (https://www.wifipineapple.com/). The other radios in the devices, like Bluetooth and NFC, are also possible entry points for an attacker. We have seen attacks lately using NFC-code stickers containing malicious code or pointing the device user to a malicious website. These stickers can be bought for less than a dollar, programmed via a smartphone and printed with for example “Scan to get 100 free Instagram followers” or something to get people to scan it. An example of a possible attack could be to make the sticker point the user to a mock-up web page that looks like Instagram and since they already think they are there to get more followers they wouldn’t react on having to enter their username and password.

3.6.3 Social media network trust

People build their social networks online, for most, as they do in real life. They either do know or at least feel they know the people in their social media network to a certain extent. This again leads to trust. People tend to trust the people in their social media network, just the way they trust their friends and co-workers’ in real life. This trust makes the users easy prey to attackers making it look like links and other things has been sent to them by someone in their network. This can be both sent directly as a personal message, shared to all friends or simply liked. The most popular form of social media scam is manual sharing. Manual sharing consists of something that looks great in some way so that users themselves will want to share it with their friends, like for example a cool video or the chance of winning a new car if you share the post. The mechanisms of these scams works the same regardless of platform, since they are relying on the intended functionality of the app or web page. There should not be any difference in the way people

3.7 Conclusion

Based upon the material I have reviewed I would have to conclude that it is much easier to attack social media users via their portable devices, than it would be to attack their computer. The main considerations behind my conclusion are the following: The majority of time spent on social media is spent accessing these on a mobile device and not a computer. Very few people install any form of malware protection on their mobile devices, while a good majority installs malware protection on their computers. The malware protection solutions that exists for mobile devices, are inferior to the systems available to do the same task on a computer. The number of vulnerabilities found for mobile devices per year is rising quickly. People tend to walk around town with Wi-Fi, Bluetooth, and NFC turned on for anyone to exploit. So the advice to the general public should be to at least install some sort of malware protection on all their devices, don't leave Wi-Fi, Bluetooth or NFC on when it is not in use.

Chapter 4

Common vulnerabilities and attack surfaces for Web Applications in 2016

Management summary

The web stack has evolved to serve feature rich and dynamic experiences within the browser by expanding on the legacy applications. The majority of cyber attacks are done at web application level, and most of the vulnerabilities are well known and documented - but alot of web applications fail to mitigate or counteract them.

This chapter several reports from security companies and non-profit organizations was analyzed to find some of the most common web application vulnerabilities with the possibility of severe consequences. The chapter focuses on Cross-Site Scripting, SQL Injection and vulnerable JavaScript Libraries to explain how they work and how they can harm the companies and their visitors/clients. It briefly looks at the rising usage of Exploit Kits, and how these vulnerabilities can compromize the web application and make it a part of an Exploit Kit infrastructure. It continues to look at possible mitigation techniqes and counter-measurements and discuss why so many sites are vulnerable to these well-known attack surfaces.

4.1 Introduction

Over the last years web application hacking has increased, and as many as 75% of cyber attacks are done at web applications level or via the web Acunetix (2016b). The introduction in the Acunetix report states that 55% of web applications has atleast one high-severity vulnerability - and thats up 9% since 2015.

The web stack has evolved to serve feature rich and dynamic experiences within the browser by expanding on the legacy applications. This has widened the attackers oppertunities, and the amount of vulnerable applications on the web is ever increasing Acunetix (2016a). In recent years an alarming number of high profile web service providers has lost huge amounts of personal data, including emails and weakly hashed passwords.
This chapter aims to find the most common vulnerabilities and attack surfaces for web applications in 2016, what the consequences can be, what measures can be taken, and hopefully a hint to why so many web applications are still vulnerable.

This chapter will focus on the application code vulnerabilities that might arise from poor programming and vulnerable third-party libraries in the application itself. A combination of the results of four induvidual information security reports will be examined to find a few vulnerabilities that occur often and are of high severity. Three of the reports are from 2015/2016, and the fourth is OWASP Top 10 from 2013. **More on methods**

OWASP Top 10

Open Web Application Security Project is a international charitable non-profit organization, and is a colaberation between a varity of security experts around the world OWASP (2016a). The primary aim of the OWASP Top 10 is to raise awareness for web application security, and so far the report has been released twice, in 2010 and 2013 OWASP (2016b).

Acunetix Web Application Vulnerability Report 2016

Acunetix is a company spacializing in automated tools to scan servers for vulnerabilities, and have many high profile private and public companies as clients Acunetix (2016b). They released an annual report on statistics from their scans throughout the period of 1st April 2015 to 31st March 2016. Acunetix has gathered, aggretated and analyzed data from over 61.000 scans over a two year period, and are in a great position to observe trends in the field Acunetix (2016a).

Hewlett Packard Enterprise Security Research Cyber Risk Report 2016

Hewlett Packard provide a broad view of the 2015 threat landscape, based on industry-wide data and a focused look at open source, mobile and IoT Enterprise (2016).

Symantec Internet Security Threat Report 2016

Through the Global Intelligence Network, Symantec has one of the most comprehensive sources of internet threat data Symantec (2016). It's made up of more than 63.8 million attack sensors in over 157 countries and territories through a combination of Symantec products and services Symantec (2016).

4.2 Trending vulnerabilities and consequences

This section will introduce the three common high-severity vulnerabilities that this chapter will focus on. By examinin the reports listed in the Introduction, the choice has fallen on Cross-Site Scripting (XSS), SQL Injections and Vulnerable JavaScript libraries. All of the threats described here have consistently been reported as the most common vulnerabilities in web applications for several years, and 2016 is no different Project (2010)Project (2013)Acunetix (2016a). They are all vulnerabilities that arise in the development phase of a web application, and they can all potentially cause significant damage.

4.2.1 Cross-Site Scripting (XSS)

If an attacker is able to submit malicious HTML such as JavaScript to a dynamic web application, they are able to execute a Cross-Site Scripting, or XSS, attack Kirda (2011). Unlike traditional distributed systems security where access control equals authentication and authorisation, a web application uses the same-origin policy Gollmann (2011). This policy is exploited by a Cross-Site Scripting attack, when the vulnerable site is viewed by a victim the malicious content seems to come from the trused site, and the attacker can steal cookies, session identefiers and other sensetive information that the web site has access to Kirda (2011). Cross-Site Scripting can be viewed as two categories, Persistent and Non-persistent attacks. In a persistent attack the attacker is able to store some malicous code in the database or a file that are rendered as part of the website for future visitors Edmunds (2016). In a non-persistent attack the attacker uses parameters in the url to inject malicious code to the server, potentally deleting or disclosing sensitive information Edmunds (2016).

Consider the following scenario: A user connects to *trusted.com* where they are a registered and authenticated user. In *trusted.com/forum* a malicious user has posted a seemingly innocent post, but managed to append some unsanatized JavaScript code at the end of the post. The JavaScript snippet is not visually rendered by the browser, but executed in the background. The JavaScript snippet then sends the cookie of the victim to *malicious.com/save?cookie=* and append the cookie for storage.

A twitter tool called TweetDeck had a serious Cross Site Scripting vulnerability in 2014 that made it possible to tweet unsanitized input SucuriBlog (2014). This was a self retweeting tweet, with a small red heart as the only visible content. Lets look at the code as it was injected:

```
<script class="xss">$('.xss').parents().eq(1).find('a').eq(1).click()
;$('[data-action=retweet]').click();alert('XXS in TweetDeck')</script>
```

If we examine the code in this attack, it starts out by opening a *script* tag and giving it a css class called *xxs* to be able to use it as an reference. The $ refers to jQuery, a popular JavaScript Library, and uses the class previously created to locate the parent containers. Then it picks the second parent out of the list of parents, which in this case is the tweet box. It then continues to find all the link tags and again select the second one, which in this case is the retweet button - and clicks it. This in turn opens a popup to confirm the retweet and the next jQuery tag clicks on the comfirm tweet button, or *[data-action=retweet]*. It finally promts the user with a JavaScript alert saying *"XXS in TweetDeck"* and closes off the script tag.

Now this is a fairly innocent attack, and you could argue that this attacker did this to either show of their skills or alert the twitter developers of their error - or even both. There is no reason to alert the user if you want this to go unnoticed as long as possible, so it's safe to assume that this vulnerability could have caused alot more damage if the attacker wanted to.

Both of these previous examples are so-called persistent attacks. The malicious code is stored on the server to render for all visitors. An example of a non-persistent attack could be a page taking page number as a parameter in the url, where the attacker can send malicious code as a parameter - and then somehow get the victim to visit that url while authenticated Edmunds (2016).

4.2.2 SQL Injection

Structured Query Language (SQL) is the language enabeling applications to talk to most relational databases. In a web application scenario SQL is used for creating, reading, updating, and deleting, in addition to searching and filtration. It is often connected to forms and other dynamic user input functionality, and this is where most SQL-Injection attacks come into play. If an attacker is able to inject code that the server treats as SQL code, the attacker can gain access to the underlying database Bisson (2005). Databases often contain sensitive consumer or user information, and result in violations such as identity theft, loss of confidensial information, fraud and even system control.

This could possibly work on any website or web application that uses a SQL-based database, and is therefore one of the oldest, most prevalent and dangerous vulnerabilities Acunetix (2016a).

4.2.3 Vulnerable JavaScript Libraries

JavaScript has become the number one client-side language for user-friendly content display, and does often communicate with backend languages through Asynchronous Javascript And Xml (AJAX). A library in programming terms is a pre-written set of code to ease the development of applications, and is used by most dynamic websites today. One of the challenges that arises from this approach is that when a vulnerability is exposed by an attacker, a large number of sites suffer from the same vulnerability. 27% of the sampled targets in Acunetix Web Application Vulnerability Report 2016 used vulnerable JavaScript libraries within their application Acunetix (2016a).

4.2.4 Exploit Kits

Vulnerabilities compromising a site could lead to the site being a part of an Exploit Kit infrastructure. Exploit Kits are server applications made to deliver malware istead of web content Preuss and Diaz (2011). They are not a web application vulnerability in themselves, but a tool used for drive-by-download attacks through a multitude of vulnerabilities in browsers and browser addons packaged together in a kit Kotov and Massacci (2013). The reason exploit kits get a mention here in this report is that they can be a result of the three vulnerabilities mentioned above, for example an <iframe> in a XSS or even a SQLInjection or a JavaScript based redirect Preuss and Diaz (2011). The exploit kits inject shellcode to the vicim process to download a independent malware executable to the victim's hard drive and executes it Preuss and Diaz (2011). These kits has become a competative market and range from several hundreds to over a thousands dollar on the black market Preuss and Diaz (2011), and therefore also have sophisticated code and sometimes even zero-day exploits.

4.3 Countermeasures and mitigation

Losing sensetive information, about your users or your own systems, is a serious skratch in the reputation of your online enterprise. Maybe even worse is beeing part of an Exploit Kit infrastructure, putting all your visitors in harms way for drive-by downloading of malware.

Even though both SQL Injection and XSS attacks almost as old as the utilazation of databases and javascript in web applications, it still is a major source of vulnerabilities today.

The basic forms of attacks on both SQL Injection and XSS are easy to pervent, simply by constraining and sanitizing the data. Constrain the length, type, range and possibly format of any input data, and then sanitize the remaining input by escaping characters that could be interpered as code Bisson (2005). For a query parameter that set the page number in a search funcion, this data should be a integer in the range between 1 and the total number of pages. In a input box in a form, for instance a email, this data should only be accepted if it is a combination of letters and numbers on the format xxx@xxx.xx. Characters like $>$ and $<$ should never be stored as is, but rather changed to their html equvalent or some other form of rewrite **Find some sources**. To mitigate the damage from a SQL Injection attack, it's important to do ragularly backups of the database and store passwords using strong and computationally heavy hashing functions with salting. **More on how to store passwords**

This raises the question: why are so many web applications still vulnerable to these types of attacks today? My thoughs on the subject, that in no way is scientificly validated, is that there is two major reasons for this: 1. The web application scene is filled with developers that not nessesarly have a formal education in programming or web development, thus lack the basics like security. 2. An increase in frameworks and content management systems usage silently teaches the developers that the security is taken care of in the core of the framework or CMS, so that when they do create a plugin or special feature they forget to handle these risks themselves.

The JavaScript Libraries on the other hand is harder to pervent, the easiest solution here is to stay on top of all libraries the site is using. Update or remove the library as soon as a bug or vulnerability is found. Another thing that is worth mentioning is that JavaScript libraries could be used in XSS attacks aswell, as seen with jQuery in the Twitter example above. **More on JS Libs**

Bibliography

(2016). Global social media research summary 2016.

(2016). United states securities and exchange commission - form 10-q - twitter, inc.

Abokhodair, N., Yoo, D., and McDonald, D. W. (2015). Dissecting a social botnet: Growth, content and influence in twitter. In *Proceedings of the 18th ACM Conference on Computer Supported Cooperative Work & Social Computing*, pages 839–851. ACM.

Acunetix (2016a). Acunetix Web Application Vulnerability Report 2016. https://d3eaqdewfg2crq.cloudfront.net/resources/acunetix-web-application-vulnerability-report-2016.pdf.

Acunetix (2016b). Combating the web security threat.

Bialecki, M., O'Leary, S., and Smith, D. (2016). Judgement devices and the evaluation of singularities: The use of performance ratings and narrative information to guide film viewer choice. *Management Accounting Research*, pages –.

Bisson, R. (2005). Sql injection. *ITNOW, Oxford Journal*, 47:25.

Bollen, J., Mao, H., and Zeng, X. (2011). Twitter mood predicts the stock market. *Journal of Computational Science*, 2(1):1–8.

Boshmaf, Y., Muslukhov, I., Beznosov, K., and Ripeanu, M. (2011). The socialbot network: when bots socialize for fame and money. In *Proceedings of the 27th Annual Computer Security Applications Conference*, pages 93–102. ACM.

Boshmaf, Y., Muslukhov, I., Beznosov, K., and Ripeanu, M. (2013). Design and analysis of a social botnet. *Computer Networks*, 57(2):556–578.

Cao, Q., Yang, X., Yu, J., and Palow, C. (2014). Uncovering large groups of active malicious accounts in online social networks. In *Proceedings of the 2014 ACM SIGSAC Conference on Computer and Communications Security*, pages 477–488. ACM.

Cassa, C., Chunara, R., Mandl, K., and Brownstein, J. (2013a). Twitter as a sentinel in emergency situations: Lessons from the boston marathon explosions. *PLoS Currents*, (JUNE).

Cassa, C. A., Chunara, R., Mandl, K., and Brownstein, J. S. (2013b). Twitter as a sentinel in emergency situations: lessons from the boston marathon explosions. *PLOS Currents Disasters*.

Danezis, G. and Mittal, P. (2009). Sybilinfer: Detecting sybil nodes using social networks. In *NDSS*. San Diego, CA.

Davis, C. A., Varol, O., Ferrara, E., Flammini, A., and Menczer, F. (2016). Botornot: A system to evaluate social bots. In *Proceedings of the 25th International Conference Companion on World Wide Web*, pages 273–274. International World Wide Web Conferences Steering Committee.

Douceur, J. R. (2002). The sybil attack. In *International Workshop on Peer-to-Peer Systems*, pages 251–260. Springer.

Edmunds, B. (2016). *Safe Defaults, Cross-Site Scripting, and Other Popular Hacks*, pages 41–47. Apress, Berkeley, CA.

Edwards, C., Edwards, A., Spence, P. R., and Shelton, A. K. (2014). Is that a bot running the social media feed? testing the differences in perceptions of communication quality for a human agent and a bot agent on twitter. *Computers in Human Behavior*, 33:372 – 376.

Enterprise, H. P. (2016). Security research cyber risk report 2016.

Ferrara, E., Varol, O., Davis, C., Menczer, F., and Flammini, A. (2015). The rise of social bots. *arXiv preprint arXiv:1407.5225v3*.

Gollmann, D. (2011). *Problems with Same Origin Policy*, pages 84–85. Springer Berlin Heidelberg, Berlin, Heidelberg.

Gupta, A., Lamba, H., and Kumaraguru, P. (2013). $1.00 per rt # bostonmarathon # prayforboston: Analyzing fake content on twitter. In *eCrime Researchers Summit (eCRS), 2013*, pages 1–12. IEEE.

Holiday, R. (2014). Fake traffic means real paydays.

Hwang, T., Pearce, I., and Nanis, M. (2012). Socialbots: Voices from the fronts. *interactions*, 19(2):38–45.

Kirda, E. (2011). *Cross Site Scripting Attacks*, pages 275–277. Springer US, Boston, MA.

Kotov, V. and Massacci, F. (2013). *Anatomy of Exploit Kits*, pages 181–196. Springer Berlin Heidelberg, Berlin, Heidelberg.

Lin, Z. (2014). An empirical investigation of user and system recommendations in e-commerce. *Decision Support Systems*, 68:111 – 124.

Mustafaraj, E. and Metaxas, P. T. (2010). From obscurity to prominence in minutes: Political speech and real-time search.

Narang, S. (2015). Uncovering a persistent diet spam operation on twitter.

Oh, C., Roumani, Y., Nwankpa, J. K., and Hu, H.-F. (2016). Beyond likes and tweets: Consumer engagement behavior and movie box office in social media. *Information & Management*, pages –.

OWASP (2016a). About.

OWASP (2016b). Top ten project.

Preuss, M. and Diaz, V. (2011). Exploit kits - a different view.

Project, O. W. A. S. (2010). OWASP Top 10 - 2010. https://www.owasp.org/index.php/Top10#OWASP_Top_10_for_2010.

Project, O. W. A. S. (2013). OWASP Top 10 - 2013. https://www.owasp.org/index.php/Top10#OWASP_Top_10_for_2013.

Ratkiewicz, J., Conover, M., Meiss, M., Goncalves, B., Flammini, A., and Menczer, F. (2011). Detecting and tracking political abuse in social media.

Rosencrance, L. (2000). Federal judge blocks web bot from tapping into ebay.

Shin, H., Byun, C., and Lee, H. (2015). The influence of social media: Twitter usage pattern during the 2014 super bowl game. *International Journal of Multimedia and Ubiquitous Engineering, 2015*, vol 10(3):109–118.

Singh, A. (2013). Social media and corporate agility. *Global Journal of Flexible Systems Management*, 14(4):255–260.

Stein, T., Chen, E., and Mangla, K. (2011). Facebook immune system. In *Proceedings of the 4th Workshop on Social Network Systems*, page 8. ACM.

Stringhini, G., Wang, G., Egele, M., Kruegel, C., Vigna, G., Zheng, H., and Zhao, B. Y. (2013). Follow the green: Growth and dynamics in twitter follower markets. In *Proceedings of the 2013 Conference on Internet Measurement Conference*, IMC '13, pages 163–176, New York, NY, USA. ACM.

SucuriBlog (2014). Serious Cross Site Scripting Vulnerability in TweetDeck – Twitter. https://blog.sucuri.net/2014/06/serious-cross-site-scripting-vulnerability-in-tweetdeck-twitter.html.

Symantec (2016). Internet security threat report 2016.

Tran, D. N., Min, B., Li, J., and Subramanian, L. (2009). Sybil-resilient online content voting. In *NSDI*, volume 9, pages 15–28.

Turing, A. M. (1950). Computing machinery and intelligence. *Mind*, 59(236):433–460.

Viswanath, B., Post, A., Gummadi, K. P., and Mislove, A. (2010). An analysis of social network-based sybil defenses. *ACM SIGCOMM Computer Communication Review*, 40(4):363–374.

Wang, G., Konolige, T., Wilson, C., Wang, X., Zheng, H., and Zhao, B. Y. (2013). You are how you click: Clickstream analysis for sybil detection. In *Presented as part of the 22nd USENIX Security Symposium (USENIX Security 13)*, pages 241–256.

Wang, G., Mohanlal, M., Wilson, C., Wang, X., Metzger, M., Zheng, H., and Zhao, B. Y. (2012). Social turing tests: Crowdsourcing sybil detection. *arXiv preprint arXiv:1205.3856*.

Weizenbaum, J. (1966). Eliza—a computer program for the study of natural language communication between man and machine. *Communications of the ACM*, 9(1):36–45.

Yang, Z., Wilson, C., Wang, X., Gao, T., Zhao, B. Y., and Dai, Y. (2014). Uncovering social network sybils in the wild. *ACM Transactions on Knowledge Discovery from Data (TKDD)*, 8(1):2.

Yu, H., Kaminsky, M., Gibbons, P. B., and Flaxman, A. (2006). Sybilguard: defending against sybil attacks via social networks. In *ACM SIGCOMM Computer Communication Review*, volume 36, pages 267–278. ACM.