Attacks exploiting users' common-interests on Social Network Systems

Η Διπλωματική Εργασία
παρουσιάστηκε ενώπιον
του Διδακτικού Προσωπικού του
Πανεπιστημίου Αιγαίου

Σε Μερική Εκπλήρωση
των Απαιτήσεων για το Δίπλωμα του
Μηχανικού Πληροφοριακών και Επικοινωνιακών Συστημάτων

του
ΓΕΩΡΓΙΟΥ ΒΛΑΣΣΟΠΟΥΛΟΥ
ΕΑΡΙΝΟ ΕΞΑΜΗΝΟ 2017

Η ΤΡΙΜΕΛΗΣ ΕΠΙΤΡΟΠΗ ΔΙΔΑΣΚΟΝΤΩΝ ΕΓΚΡΙΝΕΙ
ΤΗ ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ
ΤΟΥ ΓΕΩΡΓΙΟΥ ΒΛΑΣΣΟΠΟΥΛΟΥ:

ΚΑΜΠΟΥΡΑΚΗΣ ΓΕΩΡΓΙΟΣ , Επιβλέπων
Αναπληρωτής Καθηγητής
Τμήμα Μηχανικών Πληροφοριακών και
Επικοινωνιακών Συστημάτων

ΑΝΑΓΝΩΣΤΟΠΟΥΛΟΣ ΜΑΡΙΟΣ, Συν-επιβλέπων
Διδάκτορας - Ερευνητής
Τμήμα Μηχανικών Πληροφοριακών και
Επικοινωνιακών Συστημάτων

ΡΙΖΟΜΥΛΙΩΤΗΣ ΠΑΝΑΓΙΩΤΗΣ, Μέλος
Μόνιμος Επίκουρος Καθηγητής
Τμήμα Μηχανικών Πληροφοριακών και
Επικοινωνιακών Συστημάτων

ΚΩΝΣΤΑΝΤΙΝΟΥ ΕΛΙΣΑΒΕΤ, Μέλος
Μόνιμη Επίκουρος Καθηγήτρια
Τμήμα Μηχανικών Πληροφοριακών και
Επικοινωνιακών Συστημάτων

ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΙΓΑΙΟΥ
ΕΑΡΙΝΟ ΕΞΑΜΗΝΟ 2017

ΓΕΩΡΓΙΟΣ ΒΛΑΣΣΟΠΟΥΛΟΣ
Τμήμα Μηχανικών Πληροφοριακών και Επικοινωνιακών Συστημάτων
ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΙΓΑΙΟΥ

ΠΕΡΙΛΗΨΗ

Ο σκοπός αυτής της διπλωματικής είναι να εκτελεστεί μια επίθεση ηλεκτρονικού «ψαρέματος» σε χρήστες του Twitter χρησιμοποιώντας την επιρροή συγκεκριμένων χρηστών. Για να επιτευχθεί αυτό το διαφορετικό είδος της επίθεσης, συλλέχθηκαν προσωπικά δεδομένα ανυποψίαστων χρηστών.

Στη συνέχεια, υπολογίζεται η επίδραση όλων των χρηστών και αποστέλλεται ένα μήνυμα «ψαρέματος» στους χρήστες με τη μεγαλύτερη επίδραση. Είναι απαραίτητο να δημιουργηθεί ένα αληθοφανές μήνυμα αλλά και μια αληθοφανής σελίδα «ψαρέματος».

Μέσα από αυτή τη μελέτη, αντιλαμβανόμαστε πόσο ισχυρά είναι τα μέτρα ασφαλείας του Twitter. Δεν είναι εύκολο να συγκεντρωθούν πληροφορίες για ένα χρήστη, αλλά ούτε και ανέφικτο. Υπάρχουν ισχυρά μέτρα που λαμβάνονται από το Twitter για να εμποδίσει κακόβουλους χρήστες να συγκεντρώνουν αδιαλείπτως προσωπικά δεδομένα.

Αυτή η διπλωματική, ωστόσο, αποδεικνύει ότι η συλλογή δεδομένων στο Twitter είναι εφικτή και οι χρήστες του, δεν προστατεύονται στο βαθμό που θα έπρεπε. Αποδεικνύει επίσης ότι ένας εισβολέας είναι σε θέση να στείλει κακόβουλα μηνύματα χωρίς αυτά να ανιχνεύονται.

Η δομή της διπλωματικής έχει ως εξής: Το **Κεφάλαιο 1** περιλαμβάνει μια εισαγωγή σε ιστότοπους κοινωνικής δικτύωσης και τις επιθέσεις από τις οποίες υποφέρουν. Στο **Κεφάλαιο 2** υπάρχει μια ανάλυση της επίθεσης ηλεκτρονικού «ψαρέματος» γενικά, αλλά και στα κοινωνικά δίκτυα συγκεκριμένα. Το **Κεφάλαιο 3** αναφέρεται στο κομμάτι της υλοποίησης της διπλωματικής. Συγκεκριμένα, αναφέρεται στη συλλογή δεδομένων από το Twitter, την επεξεργασία των δεδομένων αυτών προκειμένου να βρεθούν οι χρήστες με τη μεγαλύτερη επιρροή και την εφαρμογή μιας επίθεσης «ψαρέματος» σε χρήστες του Twitter. Περιλαμβάνει επίσης τα αποτελέσματα του πειράματος, καθώς και τις δυσκολίες που συναντήσαμε. Το **Κεφάλαιο 4** αφορά το Data mining και πως θα μπορούσαμε να το ενσωματώσουμε στο πείραμα μας, ενώ το **Κεφάλαιο 5** αναφέρεται στη μελλοντική δουλειά που θα μπορούσε να γίνει για αυτή τη διπλωματική. Τέλος, στο **Κεφάλαιο 6** υπάρχουν κάποια συμπεράσματα που προέκυψαν κατά τη διάρκεια της έρευνας και της υλοποίησης.

ABSTRACT

The purpose of this thesis is to detail on a social phishing attack on Twitter users by using the influence of specific users. In order to achieve this different kind of attack, personal information of unsuspicious users were crawled. After that, the influence of all users is calculated and a phishing message is sent to the most influenced users. Also, it is necessary to create not only a believable phishing message but also a believable phishing page.

Through this study, we realize how strong the security measures of Twitter are. As explained, it is not easy to gather user information but it is not infeasible either. There are strong measures taken by Twitter to prevent malicious users to gather data nonstop. This thesis, however, proves that crawling in Twitter is feasible and its users are not so much protected as they should be. It also demonstrates that an attacker is able to send malicious messages without Twitter detecting it. The rest of this thesis is organized as follows: **Section 1** includes an introduction on Social Networking Sites and the attacks they suffer from in general. In **Section 2** there is an analysis of phishing attack in general, as well as on Social Networks specifically. **Section 3** is referred to the implementation part of this thesis. That is, Twitter crawling, the processing of information in order to find the most influenced users and the implementation of a Social Phishing Attack on Twitter users. It also include the results of the experiment, as well as some challenges we came across during the implementation. **Section 4** refers to Data Mining in Social Network Systems and how this could implemented in the experiment. **Section 5** is about the challenges we came across during the implementation of this thesis and **Section 6** refers to the future work that could be made for this thesis. Finally, in **Section 7** there are some conclusions that have been made during the research and implementation part.

ACKNOWLEDGEMENTS

At that point, I would like to thank the Assistant Professor of the Department of Information and Communication Systems Engineering, Dr Georgios Kampourakis for assigning this thesis to me. I would also like to thank Ms Anastasia Douma for the valuable assistance that she provided me from the beginning and during the programming part of my thesis.

In addition, I would like to thank the professors of the committee who agreed to attend the examination of my diploma thesis.

Finally, I would like to thank my parents and friends for the strength and courage they provided me during my studies in University of the Aegean and all the external professionals for their advice on my studies and thesis.

CONTENTS

FIGURES TABLE

1. INTRODUCTION

1.1 Social Networks

As "Social Networks" are defined services that are based on the Internet and allow authorized users to create a public or semi-public profile within a delimited system, to articulate a list of other users with whom they share a connection, and view and traverse their list of contacts, and those lists that are made by others in the system[1]. The nature of these connections may vary from site to site.

With the progress of technology, communication has grown bigger. It is now easier and cheaper to communicate and connect with people around the globe, due to the rise of social networks.
Although social networks have incorporated a variety of techniques and characteristics for ease of the user, their basic structure consists of visible profiles that display a structure list of friends, who are also users. Profiles are unique pages - since the respective social network assigns each new user with a unique ID to the end of the URL - where one can publicly expose all the elements that characterize it, either what the name and age are or location and interests. Most sites also encourage users to upload a picture for their profile or even add multimedia content. Furthermore, many Social networks such as Facebook, allow users to modify the view and feel of their profile and add applications that enrich their profile.

Online networks provide advantages not only to individuals, but also to businesses. Some of these benefits are:

- People stay connected in a very convenient and effective way, even when they are in different regions or countries.
- Help individuals to interact with each other.
- Provide a forum where people with same interests can collaborate, share knowledge/experience and establish bonds of trust.
- As for companies, social networks can enhance the company's collective knowledge, learn about their customers, create bonds and advertise their services and products.

More and more social networks have been created within the last years, with Facebook being the top in demand, followed by WhatsApp, QQ, WeChat and many others. More specifically, Facebook has 1.712 billion users, WhatsApp 1 billion and QQ reaches 899 million[2] .

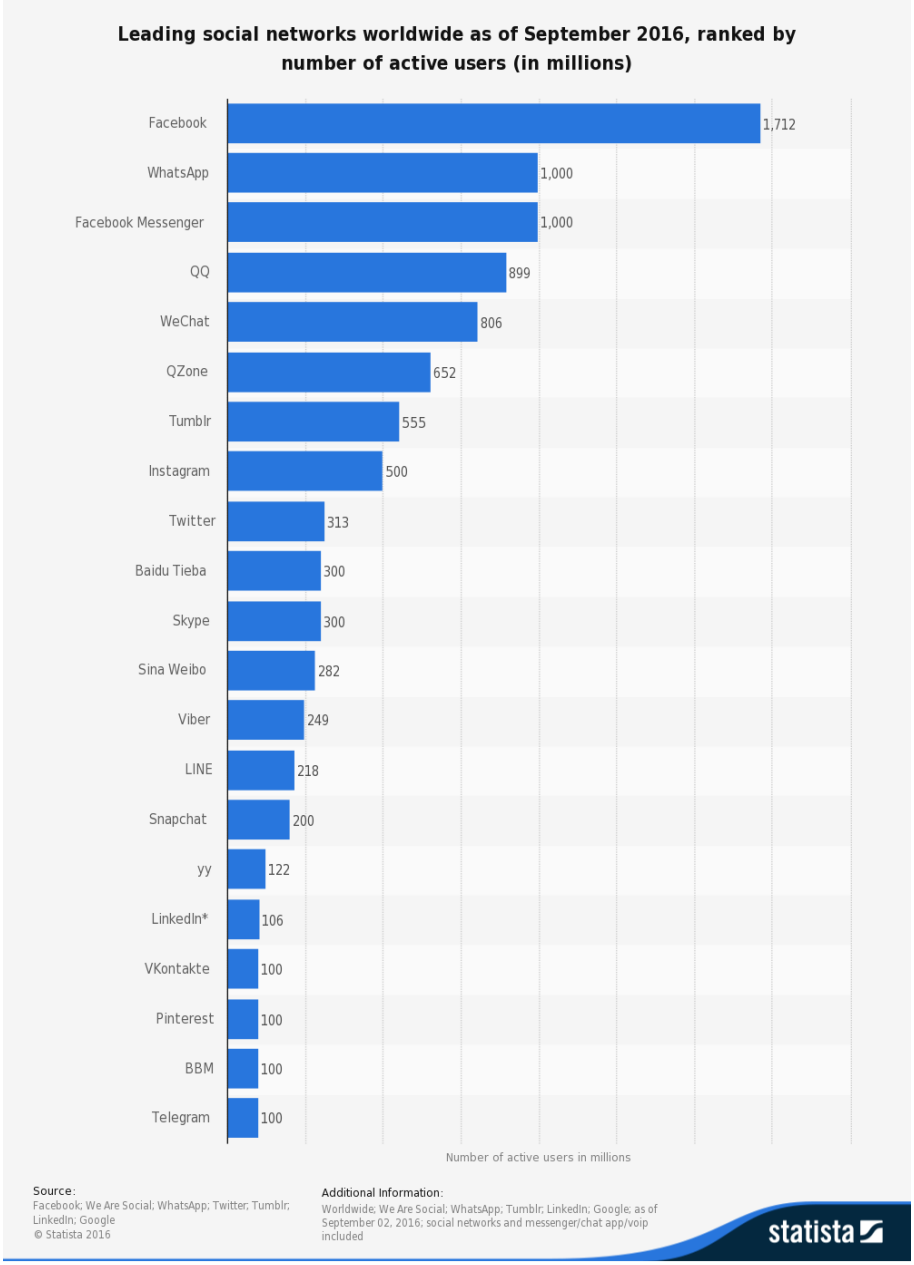

Figure 1-1 Leading social networks worldwide - September 2016

Most social networks are offered free of charge and include one quick and easy registration process. However there are some major differences not only in the services provided by them, but also in the kind of connections made with other users. These differences will be discussed later.

Myspace was the largest social networking site in the world until 2009. It is a social networking website that lets users interact with others, post multimedia content (photos, music and videos) and create a personal profile. Myspace had played a significant role in the music industry, as artists can upload their songs onto Myspace and have access to millions of people on a daily basis. Over eight million artists have been discovered by Myspace and many more continue to be discovered daily [3].

Facebook, the most widely used social network as mentioned before, allows users to create and modify their profile whenever they want. Users can also hide or show specific

information, make their profile information private, public or restricted to specific users of their contact list. Moreover, users have the ability to upload photos or any other multimedia content (songs, videos, etc), share their thoughts and how they feel and even their recent location such as restaurants, cafes, bars.

As for the communication, there is the chance of group chatting or becoming a member of a specific group, like students of Aegean University, movies fans, music group fans etc.

Instagram is an online mobile photo-sharing, video-sharing, and social networking service that enables its users to take pictures and videos, and share them either publicly or privately on the app [4].

Instagram didn't include instant messaging for users until recently as it added this feature on 2015 (Instagram exists since 2010). Users in Instagram except of messaging, can also share photos, edited or not, and videos in order to show their everyday life to their friends. Users in this network, choose who to follow and by whom they been followed, so that they can see their photos. That means that a user can follow another user, but not be followed by, and vice versa. Finally, they can share their "Instagram story" – specific photos and/or videos that are available for 24 hours- among their followers.

Instagram users can choose whether their profile will be public or private. In a public profile all the context is available to everyone (stories, photos and videos). Stories let the user know by whom they have been watched, but photos and videos don't let the user know.

Google+ is a widely known social network that launched on 2011 and is used for communication and information exchange. It provides teleconferencing services, photo sharing, adding friends, etc.

LinkedIn is the most successful social network for professionals around the world. Its users have the ability to post their resumes, write about their skills, certificates, invite friends and acquaintances in their network, etc. Through this network, users can find jobs depending on their skills, as companies keep posting ads about jobs every day. Companies can also find desirable candidates, as they can see users' resumes and skills.

Twitter.com is an online platform used by 320 millions of people, where they can stay connected through computer and mobile phones with their friends, family and coworkers. Users can post short messages (up to 140 characters), which are called "tweets", that can be read by their "followers". Users can also "follow" other users, if they want, so they get notified when that user posts a new tweet.

1.2 Attacks on Social Networks

However, there are not only benefits when it comes to social network sites. Many risks come along with the use of these services. The challenges of privacy and surveillance are complicated when it comes to social media, because anyone can access user's personal information, without even knowing who has access. This information may be used to harm the user.

The natural consequence of the progressively usage of online social networks (OSNs) in users' daily lives, is that personal information is accessible from everyone. The majority of everyday users are unaware of the numerous security risks that exist in social networks, such as privacy violations, sexual harassment and identity theft, malware , fake profiles (also in

some cases referred to as sybils or socialbots). This lack of knowledge and awareness of the cyber attacks is an important factor of success for many malicious attacks.

OSN users expose personal and private information about themselves, including email address, phone number, relationship status, date of birth or even home address. This information can be harvested by the OSN operator itself and by third-party commercial companies.

Information harvesting, has been identified as a significant security concern for OSN users, as if this information is put into the wrong hands, can harm user in both real and virtual life. These risks become even more severe when the users are children.

As for Twitter, Twitterers do not include identifiable information such as phone numbers, email and home addresses. However, about a quarter of tweets do include information about where and when they do specific activities. This kind of information may have privacy implications when found in the same tweet or if coupled with other kinds of publicly available information.

For many years now, providers are trying to find ways to deal with the attacks. In many cases, however, this is impossible since they cannot control their users' actions and decisions. Even if a user's computer is infected, no one can see fraud actions going on and the user may not realize that he has been victim all this time. As a result, the user cannot be protected or prevent malicious activities because he doesn't even know that such an activity is going on to his computer.

The main reason why social network providers cannot protect their users is that all the information and activities related to a user's profile are public not only to the user's friends but across the network. If the user himself does not change that setting from public to private, all his information will be available to everyone unexceptionally. It is entirely up to the user to make his profile and information private or not.

Twitter allows people to share information among friends and "followers". However, the default privacy setting is that all these messages are public and can be seen by a registered Twitter user. In addition, all public tweets may be posted to a public timeline website which showcases the twenty most recent tweets.

In this subsection I will present some of the most common attacks on social networks in general, and in twitter more specifically, along with their architecture and all the steps needed to be taken in order for an attack to take place.

Phishing: Phishing is an identity theft attack that aims to steal sensitive information and credentials, such as passwords, credit card information, online banking credentials etc, from users. It will be discussed further in the next chapter.

Social spam: Social spam is a type of spam attack. Traditional spam methods include sending spam emails and creating spam web content. In order to be more effective, spammers tend to send individual customized messages towards the targeted user, and obfuscate the destination URL in order to avoid detection.

The past few years have witnessed the rapid rise of online social networks. One key feature of such systems is the reliance on content contributed by users. OSNs help build trust relationship between internet friends, even though they may not know each other in real world. This made OSNs an ideal target of social spam. Social spammers can achieve a much higher success rate by exploiting the social trust among users.

Spam is difficult to measure as it change day after day. Statistics on the amount of spam in e–mail vary. In 2009, Microsoft reported that 97 percent of all e–mail messages sent over the Web were unwanted, Google reported that spam hovered between 90–95 percent, and Symantec reported that spam accounted for 90.4 percent of all e–mail (Swidler, 2009; Symantec, 2009; Waters, 2009). In August2009, 11% of Twitter messages were spam messages. This changed in the early 2010's, as Twitter minimized the percentage of spam messages to 1% [5]. Nevertheless, a 2013 article [6] states, "Social spam, as it already exists on Twitter, will continue to grow and unless the company addresses the problem quickly, it may be the one thing that sinks it".

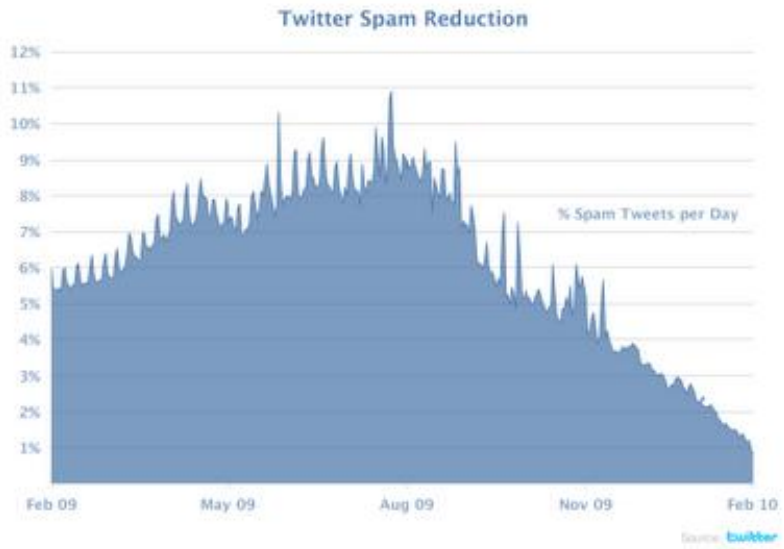

Figure 1-2 Twitter Spam Reduction

These days, most e-mail spam is filter by hosts and e-mail clients and users don't notice it. However, spammers persist and try to advance spamming methods in more sophisticated and creative ways.

Malware: Malware is short for "malicious software". It refers to software programs designed to damage or do other unwanted actions on a computer system. Common examples of malware include viruses, worms, Trojan horses, and spyware. Viruses can damage user's computer by deleting files or directory information. This would create a total havoc. Spyware on the other hand is a type of passive attack. By using spyware, the attacker can gather data from a user's system without the user knowing it. Information an attacker could gather vary from cookies of the visited web pages, to credit card numbers and credentials. Firewalls, antivirus and antimalware, can protect us from these kinds of attacks or help us recover after these attacks.

One of the first online social network worms was Myspace.A [7], which was detected in 2005. Worms are type of Cross-site Scripting attacks (XSS– mostly found in Web applications). This worm launched against Myspace and allowed an attacker to add millions of new contacts. Another famous worm was Koobface [8], which first surfaced in 2008 and was the first malware to successfully propagate through OSNs such as Facebook, MySpace, and Twitter. Koobface inserted comments with fake links pointing to malicious websites. A victim's machine, once infected by this worm, turned into a zombie machine on a botnet in

order to send spam messages and attack other computers and servers over the Internet. The following diagram shows the total number of infections after 20 hours of some well-known computer worms.

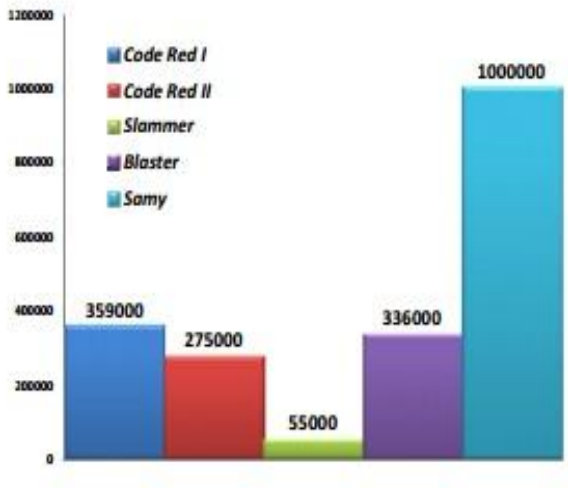

Figure 1-3 Well known Malwares and their total number of infections

- The first two worms, Code Red I & II, were designed to exploit a security hole in the 0 indexing software included as part of Microsoft's Internet Information Server (IIS) web server software.
- Slammer worm infected the process space of Microsoft SQL Server 2000 by exploiting buffer overflow.
- Blaster Worm was a virus program that mainly targeted Microsoft platforms in 2003. The worm attacked computers by exploiting a security flaw with Microsoft remote procedure call (RPC) process using Transmission Control Protocol (TCP) port number 135. The virus propagated itself automatically to other machines by transmitting itself through email and other methods.
- Samy worm, also known as Spacehero was a Javascript worm that infected Myspace pages. It gave its creator over a million friends one day in the fall of 2004.
- Last but not least, a worm called “Mikeyy” spread on approximately 10.000 tweets on Twitter, back in 2009. The Mikeyy worm started to spread via Twitter posts by encouraging users to click on a link to a rival microblogging service StalkDaily.com. As soon as they clicked on the link their account would be infected and begin to send out similar messages. SYBIL attacks refer to individual malicious users creating multiple fake identities (called sybil identities or Sybil nodes) in open-access distributed systems (such as peer-to-peer systems). These open-access systems aim to provide service to any user who wants to use the service (instead of, for example, only to a predetermined group of 10 users). This attack is also used by spammers to gain access to multiple accounts on free email systems.

Internet Fraud: Internet fraud, also known as cyber fraud, refers to using Internet access to scam or take advantage of people. It is a type of Social Engineering attack, where attackers try to manipulate the victims and establish strong bond with them, in order to steel their credentials, or even money. In recent years, for example, fraudsters have been hacking into the accounts of Facebook users who travel abroad. Once they manage to log into a user's account, the scammers cunningly ask the user's friends for assistance in transferring money to the scammer's bank account.

2. PHISHING ATTACKS

2.1 Phishing Attacks

A typical phishing attack is described below:

- Attacker collects data for the victim through different means, such as forums and social network sites.
- The attacker creates a malicious webpage where the victim will be redirected through email or a spoofed message. Email or message will seem to be sent by someone by a familiar to the victim user, in order to be more persuasive. In SNS phishing particular, this message will be sent to all the friends in the victim's friend list.
- After the redirection, victim will have to enter his credentials, something that actually happens, as users seem to trust entities that have been sent by friends.
- The credentials and information, such as credit card numbers, bank accounts etc are now in the attacker's hands, ready to be used.

Phishing method has been used for many years, so defensive mechanisms have been developed in order to protect the users.

However, not only defensive mechanisms, but also attackers have advanced. Phishers nowadays frequently co-operate with spammers, in order to send out faked e-mails in very large numbers.

According to a research conducted by Kaspersky Lab, 22% percent of the phishing scams target Facebook. The report stated that Kaspersky Lab identifies more than 20.000 incidents per day in which the users of Kaspersky Lab attempt to visit Facebook pages.

Phishing attacks can be categorized in different categories:

Spoofing e-mails and web sites

The earliest form of phishing has been used since mid 90's. The attacker sent spoofed e-mails to the victim and tried to persuade them to send back their credentials and passwords. However, users nowadays are aware of such actions and avoid sending sensitive information via e-mail. This happened because many security-sensitive organizations such as banks use their websites to interact with the users and they don't use email providers to send and receive sensitive information.
As mentioned before, attackers have also advanced their techniques. They now combine spoofed e-mails and websites to perform the attack. In a typical attack, the attacker sends a spoofed email to the victim that appears to be sent from a legitimate organization and urge the victim to update their personal information. This email contains a redirection url, so the victim redirects to the spoofed website. In this website, that looks and feels like the real one, the victim is more possible to enter his personal information than the regular phishing attack, as success rates of such attacks are much higher.

Exploit-based phishing attacks

Attackers in this category make use of vulnerabilities to install malicious software, such as Trojans and malware, to victims' computers. By using this software they can collect all the information they need. A type of software that attackers can use is a key logger, in order to

keep logs of all pressed keys. When the victim visits a desired website (social networks, online banking web sites), all the credentials will be kept in logs.

Although it is very difficult to detect phishing attacks, researchers are working on different phishing detection mechanisms.

Abraham and Raj [9] proposed a string matching approach for detecting phishing campaigns.

This mechanism checks the similarity of a URL with the blacklisted URLs, depending on text-based characteristics. If these two URLs are similar, then it is classified as phishing. Otherwise it is classified as legitimate. The main idea behind this method is described below.

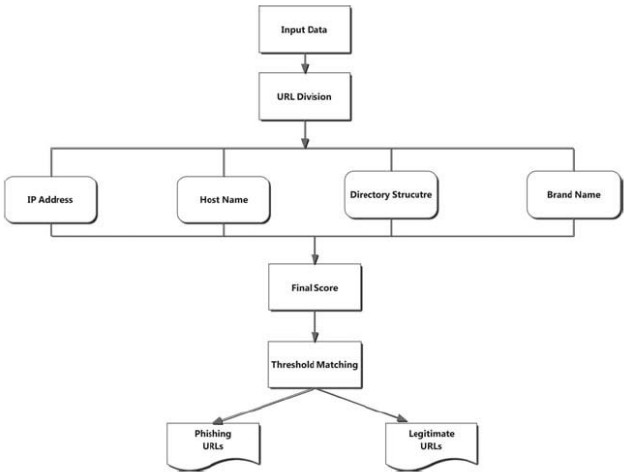

Figure 2-1 Abraham & Raj string matching approach for detecting phishing campaigns

As we can see, the URL divides into tokens and each token is compared with the blacklisted URLs tokens. The scores are calculated based on the number of the occurrence of each token in the blacklist.

This approach has great results in phishing detection, with very false negatives and false positives, as it combines two string-matching algorithms to compare the hostnames, longest common subsequence (LCS) and edit distance. The accuracy rate of LCS is 99.1% and for edit distance is 99.5%.

Beiddermann, Ruppenthal, and Katzenbeisser (2014) [10], proposed another phishing detection architecture based on transparent virtualization technologies and isolation of the own components. This architecture can be used in cloud computing, as it is a security add-on for a virtual machine (VM). It uses VM introspection (VMI) to obtain and filter the fingerprint of a webpage which are managed by a browser from the VM's memory. This architecture is proper for detection for two kinds of phishing attacks: 1) "man in the browser" attacks (MitB) and 2) spoofed web page redirection.

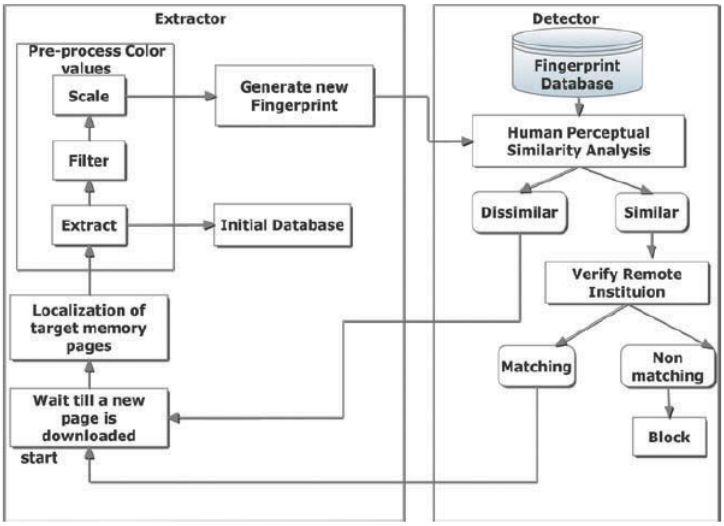

Figure 2-2 Beiddermann, Ruppenthal, and Katzenbeisser phishing detection architecture

As it is described in the figure, this architecture consists of two major components: 1) the extractor and 2) the detector.

- The extractor detects, retrieves and pre-processes all the important information that is stored in the memory, and
- The detector performs human perceptual similarity analysis. Once a phishing attack has been detected, then it intervenes.

This architecture runs in isolation thus cannot be fooled by malware. Finally, it can be easily enabled by an ordinary cloud user, as it doesn't require any other software and it is supported by every browser in all the operating systems.

As for Twitter, some researchers proposed the use of PhishAri [11], a tool that helps in the detection of phishing tweets in real time. PhisAri uses different features such as attributes of the user that have posted the tweet, content of tweet, the properties of the URL posted etc. This tool uses machine learning techniques in order to categorize a tweet as "phishing" or "safe. There is also an extension for Chrome browser for real time phishing detection by appending a red indicator to phishing tweets. The accuracy of this tool is higher than 92.5%.

2.2 Phishing Attacks on Social Networks

Because of the growing usage of social networks, phishers nowadays are trying to take advantage of the information that users share. Social phishing is a type of phishing attack that relies on social network users and their actions within it, especially in Facebook, Twitter etc. Attackers follow specific steps, as in every phishing attack, in order to accomplish their goal. At first, attackers pretend to be a trusted entity, in order to gather sensitive information from a possible victim. They usually use the cover of specific websites, such as auctions, banking sites and any page with sensitive information.

Detecting phishing activities on Social Network Sites is a challenge for the researchers. First of all, the data amount that shared every day in social networks is huge, as users allowed

to upload unlimited content about their selves. Shortened URLs also have a huge impact on SNS phishing detection, as a lot posts can contain malicious hidden URLs.

A typical Social Phishing attack is described below:

- First, the attacker, unleash the Denial of Service (DoS) attack, in order to block victim's password.
- Attacker sends victim an email asking him to reveal his credentials, so that his account is activated again.

Attackers not only use emails, but they also can send a direct message to the victim, containing a malicious hyperlink to a fake website. Typically in these sites, they ask the victim to write their credentials, but the information asked inside can be varied.

Researchers Tom N. Jagatic, Nathaniel A. Johnson, Markus Jakobsson and Filippo Menczer in their research, launched an actual phishing attack targeting university students aged 18 to 24 years old.

Students received a spoofed e-mail stated that their password had been deactivated. Figure 2-3 illustrates the architecture of the attack and its steps.

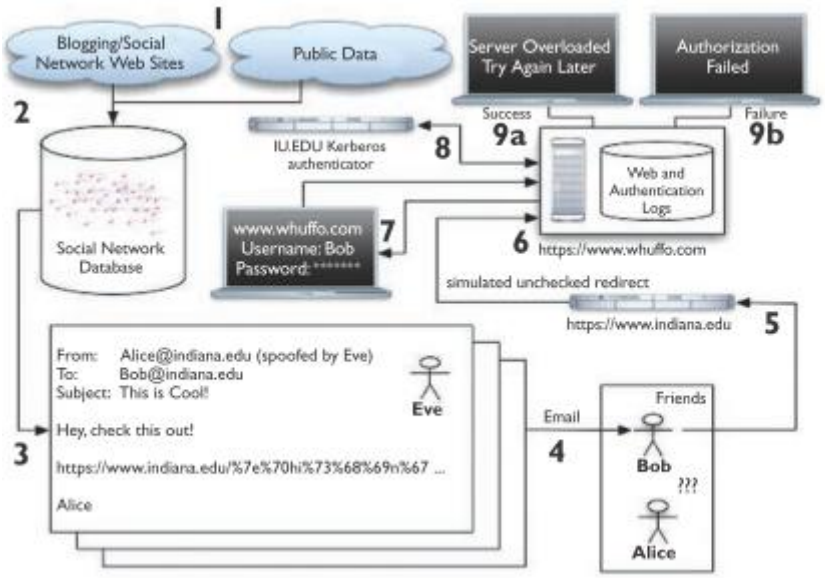

Figure 2-3 Jagatic, Johnson, Jakobsson & Menczer phishing attack procedure

1. Attacker retrieves public data or data from social networks, blogs and other websites.
2. A social network database is created, where all this data is stored.
3. The attacker creates a malicious e-mail, where his true identity is hidden. This e-mail contains a fake link that redirects the victim to the spoofed website.
4. The e-mail is sent to the victim.
5. When the victim clicks to the link, he is redirected to the desired website.
6. Victim is sent to whuffo.com site but it seems to be the website of his University.
7. Victim is asked to enter his University credentials (username & password).

8. These credentials are verified by the University authenticator (Kerberos).

9. (a) “Server Overloaded. Try Again Later” message is shown. This message is used because there doesn’t exist an actual server and there is no website to redirect the victim after a successful login. The attacker has, now, successfully stolen the credentials.

   (b) “Authorization Failed” message is shown when something goes wrong during the phishing procedure.

3. IMPLEMENTATION

As already pointed out in the Abstract, the purpose of this thesis is to perform a more targeted kind of social phishing attack based on Twitter profiles. A key step for the proper operation of the application is to get the data (Dataset) needed for subsequent processes. In fact, experiment consists of 3 different tasks. First we needed to collect data and valuable information of Twitter users. All this kind of information will later be used in order to calculate the influence/impact of users and determine whether a user is an influencer or not. After finding influencers and possible victims, we needed to design and launch the attack.

3.1 Twitter Data crawling

The first task was to crawl all the information of the potential victims. For this reason, a Java program was developed by to connect to Twitter page using our access token key and credentials. The access token key was available at any time through the developer's page of Twitter (https://dev.twitter.com/).

At first, using twitter4j library (http://twitter4j.org/en/index.html) an unofficial Java library for the Twitter API, we created a twitter crawler in order to crawl and save in a database all the crucial information, such as followers of a user, followings, lists of a user and the members of these lists, tweets of a user and some metrics we will discuss bellow. This information is saved in a SQLite table, in order to help us with the data processing in next stages. By the use of this data, we will understand which profile has more influence among the users we collected, so the phishing attack will be more effective. However, the Twitter API itself, created some difficulties, the majority of which have been overcome. Twitter API comes with plenty of Rate limiting. This is primarily on a per-user basis - or more accurately described, per user access token. In order to overcome rate limiting, I created 10 different applications in the developer's page to achieve the nonstop crawling. For this project we used a DELL Inspiron 15 3000 Series with 4GB RAM and 1.7GHz Intel Core i5-4210U CPU.

For this experiment, as we wanted our target to be the University of Aegean we decided to collect information from the members of list "AegeanUni::SAMOS" (https://twitter.com/MyAegean/lists/aegeanuni-samos), approximately 104 users.

Our database consists of 7 different tables:

TWITTER_GROUP: Generally includes information for the selected lists, such as group id, name of this list, description and slug. In our case only one list has been added.

GROUP_MEMBER: Consists of all the users of a specific list ("AegeanUni :: SAMOS" in our case). For these members we save their username and their id.

FOLLOWERS: Through our application, we can crawl every follower of a specific user, in a real simple way. All this information that has been crawled, is stored in a table with the name "FOLLOWERS". In this table we store every detail of the follower of a specific user, such as username and id.

USER_TWEETS: In this table we store the name of the user that posted a specific tweet and also the tweet itself.

USER_FAVORITES: Consists of the username of the user that marked a tweet as a favorite, the tweet itself and the id of the user that posted it.

USER_METRICS: In this table we store different kind of metrics for specific users. These metrics are the number of followers, number of users that a user follows, number of tweets in general, number of original tweets, number of retweets and number of tweets that marked as favorites. This information will later be used in order to measure user influence.

USER_INFLUENCE: In this table we store the influence rate of the users, as well as the *klout score*. It will be discussed further later.

In order to use Twitter API, OAuth protocol is used to authenticate the users. OAuth is an authentication protocol that allows users to approve application to act on their behalf without sharing their password. To bypass the Rate limiting of Twitter API, I created 10 different Twitter applications, each of them comes with Consumer Key (API Key) and Consumer Secret (API Secret). These keys are saved in a text file and they are used when necessary.

After every call we check the remaining calls. In the last call we move on to the next pair of keys and start again. The exact same method is used for followers too. After filling the FOLLOWERS table, we create a list with the distinct names of users we have their followers. For these users we run the method for metrics. First, the USER_METRICS table is created and then, by calling some methods from the twitter4j API, we fulfill specific columns of the table. These columns are number of followers, friends and tweets. Original tweets, reposts and tweets containing url will be fulfilled later, as well as favorites. Here, as Rate limiting wasn't a problem, we didn't use other pairs.

For favorites, as we mentioned before, there is a table USER_FAVORITES where we save the id of the user that posted this tweet, tweet itself and the username of the user that marked it as favorite. In this method we also update the METRICS table with the number of favorite tweets.
As for tweets of a user, we use another method. In this method not only the USER_TWEETS table is created but also the metrics table is updated with the number of original tweets, retweets and url containing.

3.2 Influence algorithm

Influencers are individuals who have the power to affect purchase decisions of others because of their (real or perceived) authority, knowledge, position, or relationship. It is a term most used in marketing. Influencer marketing has exploded the past years, as it helps in targeting exposure to the right kind of consumer, one who is already interested and will likely pay attention. This trend is not limited to mainstream and popular markets such as fashion, athletics, or entertainment. There are influencers in markets centered on everything. Keeping that in mind, it would be interesting to examine whether an influencer can make a phishing attack more effective or not.

There are plenty of algorithms that can be used to find the influence of a user. In this experiment, we used the algorithms described below, as they were easy to use, accurate and they can be used with the collected data. In order to find the rate of influence of the users there is the need of finding the popularity and the general activity of users. For general activity we used the "Tweet count score" measure by Noro, Ru, Xiao, and Tokuda (2012) [12] as described in the paper of Fabián Riquelme, Pablo González-Cantergiani [13]. Users can be described as active when participate in the social networks constantly and frequently in a period of time, regardless of the attention they receive for their participations.

Tweet count score counts the number of original tweets plus the number of retweets. For every user *I* we define the General Activity as follows

GeneralActivity (i) = OT1 + RT1 + FT1

where OT1 are the original tweets of the users, RT1 the number of user's tweets retweeted by others and FT1 the number of user's tweets marked as favorites by others. As for the Population, the simpler popularity measures just count the follow-up relationships between users. The FollowerRank Nagmoti [14] , also known as Structural Advantage Cappelletti and Sastry [15], is the normalized version of the traditional indegree measure Hajian and White[16]; Jin and Wang[17] for social networks in general:

$$FollowerRank(i) = \frac{F1}{F1+F3}$$

where F1 are the followers of a user and F3 the followings.

In addition to this algorithm, there are some tools that measure the influence. The majority of these tools (Peer_Index, Crowdbooster, SumAll, Sprout, Commun.it) aren't free or can be used for free in demo versions. The only free tool is the most well-known tool as well, Klout. Klout is a website and mobile app that uses social media analytics to rate its users according to online social influence via the "Klout Score". Klout measures the size of a user's social media network and correlates the content created to measure how other users interact with that content. Klout uses Bing, Facebook, Foursquare, Google+, Instagram, LinkedIn, Twitter, YouTube, and Wikipedia data to create Klout user profiles that are assigned a unique "Klout Score". It comes with a java API too, making it ideal for our project. Through Klout API (https://klout.com/s/developers/home) we linked our app with the Klout website so we can get Klout score for the users we gathered before.

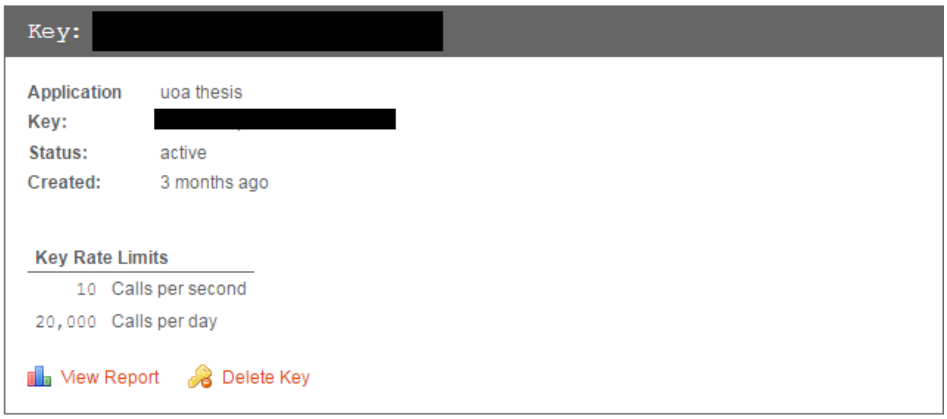

Figure 3-1 Klout registered application

All these influence metrics are stored in the database, in the table USER_INFLUENCE. If a user is above average in GENERAL_ACTIVITY and POPULARITY rates of the table and has a klout score greater than 30 (KLOUT developers state that the average Klout Score is 40), is automatically an influencer and a future victim.

3.3 Phishing

The idea here is to tempt the victim with a contest only for members of University of Aegean.

After finding the most influencing users, we send them a tweet mention, letting them know about the contest and its rules. The rules in order to participate are:

- Share the phishing page
- Login in the phishing page with the university credentials
- Follow Aegean Members Club Twitter account

Figure 3-2 Phishing tweet sent to victim

Tweet also contains a redirect link that redirects the victim to the contest page. For every influencer, a personalized url is created. This will let us know who was responsible for the redirection of the victim and whether an influencer will make the attack more effective or not. For this reason, we sent a tweet mention to some users that didn't meet our criteria.

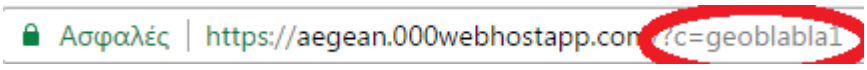

Figure 3-3 Phishing URL sent to victim

Victim will be redirected to a web page that contains information about the contest. In the left column there is information about the prize and in the left column terms of the contest are written. First term of the contest is that the user should share a tweet about the contest and its url, follow twitter page and click the button, in order to confirm that is member of the University.

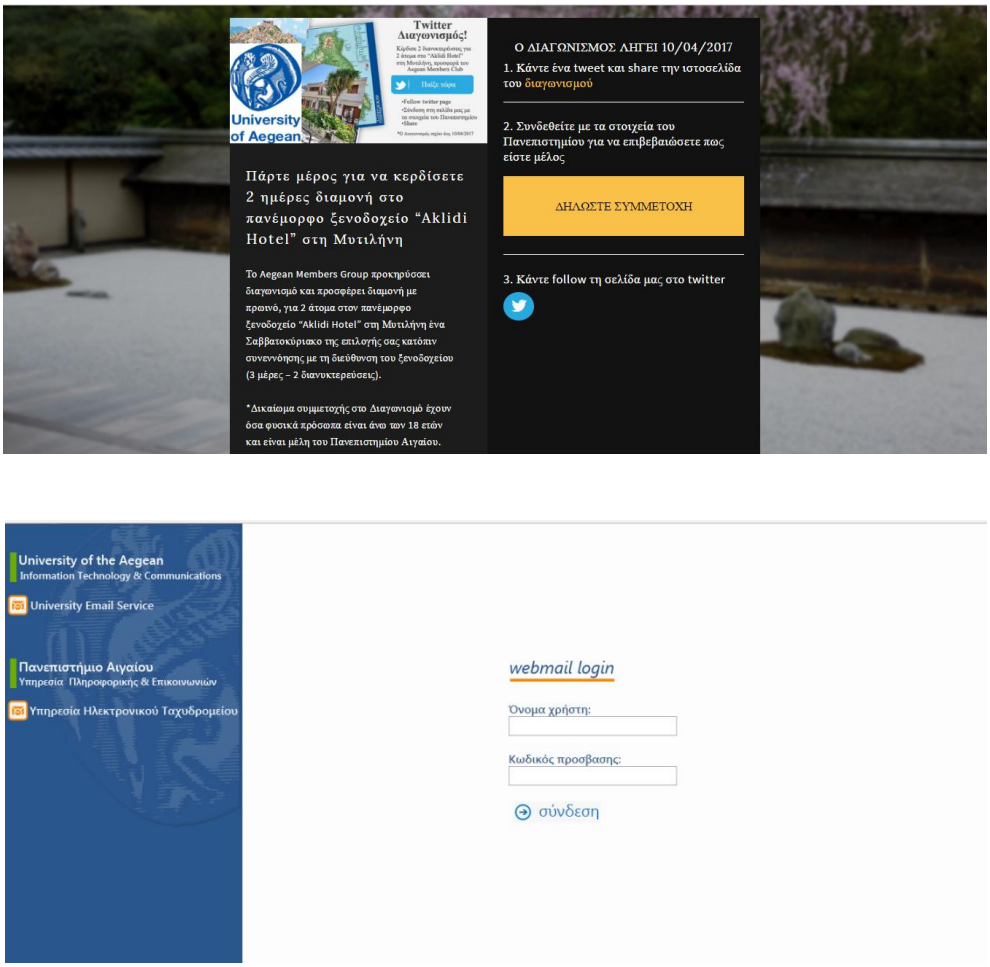

Figure 3-4 Phishing website

Figure 3-4 shows the redirect page. It is an exact copy of the University of Aegean webmail page (http://webmail.aegean.gr/). Phishing page asks about the username and the password of the user. After submit, a pop up message appears informing users about a server problem.

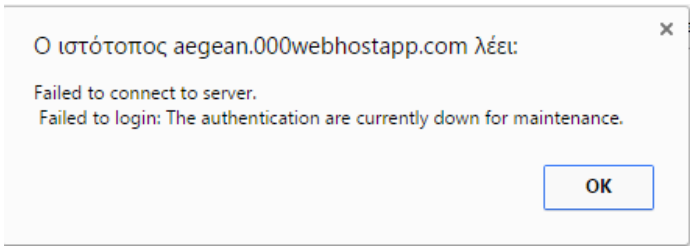

Figure 3-5 Error message after submit

Now everything is saved in our database. In the database there is a table where username and the author are been saved. By author we mean the victim that is been written in the url, as seen before.

Password isn't stored, as we didn't want to cause any damage to the possible victims.

3.4 Results

In the first phase of the experiment we managed to collect data for the 108 members of the "**AegeanUni :: SAMOS**" list. However, we managed to obtain credentials of only one user. Our first thought was to keep tweeting about the contest to influencers. However, this would be useless, as repeating tweets don't guarantee success.

Another thought was to try to improve relations with the possible victims. In order to improve relations, the first thing was to follow users and wait to be followed by. However, as this take a lot of time, we didn't implement it.

The best solution was to create a bigger database.

So, in the next phase we enriched our crawled data with four more lists: RHODES, LEMNOS, LESVOS and SYROS. After crawling these lists we managed to enrich our database with 249 new users. For these new users we crawled tweets and metrics, but not their followers' information, as it was the most time consuming process. After this phase, our database consists of 353 users, 199864 tweets and influence for 281 unique users.

The metrics of these new users calculated again using the same algorithms, arising 20 influencers.

In addition to these influencers, we found 20 non-influencers that don't meet our criteria. The phishing message has successfully been sent to the 20 influencers and ~30 non-influencers, as Twitter detected the spam tweets and blocked our account temporarily.

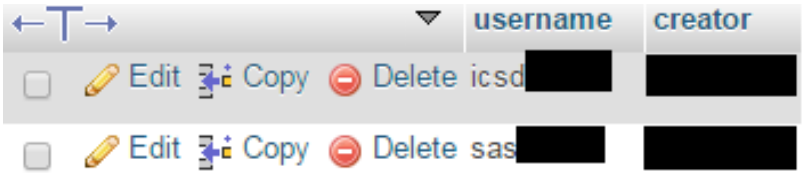

Figure 3-6 Stolen Credentials by phishing attack

3.5 Challenges

During this thesis we came across some challenges and difficulties in terms of Twitter security and permissions.

First of all, Twitter lets specific calls/app in order to collect data. Rate limits depend on the kind of data a user wants to collect.

Endpoint	Resource family	Requests / window (user auth)	Requests / window (app auth)
GET account/verify_credentials	application	75	0
GET application/rate_limit_status	application	180	180
GET favorites/list	favorites	75	75
GET followers/ids	followers	15	15
GET followers/list	followers	15	15
GET friends/ids	friends	15	15
GET friends/list	friends	15	15
GET friendships/show	friendships	180	15
GET geo/id/:place_id	geo	75	0
GET help/configuration	help	15	15
GET help/languages	help	15	15
GET help/privacy	help	15	15
GET help/tos	help	15	15

Figure 3-7 Twitter API rate limits

As described before, we overcame this challenge by creating different apps.

A challenge that we couldn't overcome was the direct messages. Our first thought was to spread the contest information by sending direct message to the victims. However, Twitter allows sending direct message only to users that they follow your account. As almost none of the users follow our account, it was impossible to spread the phishing attack by a direct message.

Twitter spam detection mechanisms have developed over the years. Although we didn't have a problem crawling information from users, sending spam tweets have triggered these mechanisms and our account had temporarily been banned. However, the unblock process was really simple, as only email/sms verification was needed.

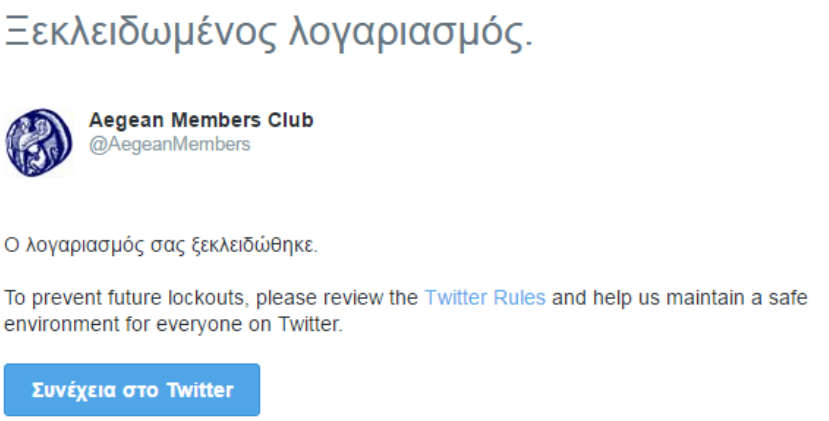

Figure 3-8 Unlocked Twitter account after email/sms verification

4. DATA MINING

4.1 Data mining background

Data mining, as a young field, has been spearheading research and development of methods and algorithms handling huge amounts of data in solving real-world problems. The main goal of a data miner is to extract valuable information from data set that is not readily apparent and not always easily obtainable. With the growth of social networks, an unprecedented amount of data is available among the internet. This data contains information about different fields of study including sociology, business, psychology, entertainment, politics, news, and other cultural aspects of societies. Applying data mining to social media can yield interesting perspectives on human behavior and human interaction. Data mining can also help in understanding the opinions people have about a subject, identify groups of people amongst the masses of a population, study group changes over time, and influential people. Social media data have three characteristics that pose challenges for researchers: the data are large, noisy, and dynamic. In order to overcome these challenges, data mining techniques are used by researchers to reveal insights into social media data that would not be possible otherwise.

Data mining techniques can help effectively deal with the three main challenges with social media data. First, social media data sets are large and can be noisy. Without automated information processing for analyzing this data, social network data would remain unexploited. Data from online social media is also dynamic, as it changes frequently and updates over short periods of time.

In order to produce a data mining benefit, we should keep in mind that every type of social media and each data mining purposes applied, may require a unique approach and specific algorithms to produce this benefit. Different data sets and data questions require different types of tools. If it is known how the data should be organized, a classification tool might be appropriate. If we understand what the data is about but cannot ascertain trends and patterns in the data, a clustering tool may be best. The problem itself may determine the best approach. There is no substitute for understanding the data as much as possible before applying data mining techniques, and second, understanding the different data mining tools that are available.

After understanding we need to represent data as a graph that contains a select number of nodes, known as seed. Graphs are traversed beginning with the set of seeds and as the link structure from the seed nodes is exploited, data is collected and the structure itself is also analyzed.

Using the link structure to extend from the seed set and gather new information is known as crawling the network. As social media present challenges such as format changes and invalid links, the application and the algorithm must deal with them. As the crawler discovers new information, it stores the new information in a repository for further analysis. As link information is located, the crawler updates information about the network structure.

The most common data mining applications related to social networking sites [18] include:

Group detection: Finding and identifying a group is one of the most popular applications of data mining in social networking sites. In general, group detection applied to social networking sites is based on analyzing the structure of the network and finding individuals that associate more with each other than with other users. Understanding what groups an individual belongs to can help in data analytics, such as what activities goods and services an individual might be interested in.

Group profiling: Once a group is found, we need to profile this group and understand what the purpose of its existence is. This is really useful for a variety of purposes ranging from purely scientific interests to specific marketing of goods, services, and ideas.

Advanced data mining techniques are proposed to account for changes in the group profile over time by defining a topic taxonomy. Identifying and tracking changes using the topic taxonomy can let us know how group values changes as well as provide a mechanism for identifying similar groups. To accomplish this, the list of topics representing a group is organized into a tree. The parent child relationships in the topic tree define a taxonomy for the group. This tree can be compared with alternative tree structures to identify the most accurate classifier for a particular group.

Recommendation systems: A recommendation system analyzes social networking data and recommends new friends or new groups to a user. The ability to recommend group membership to an individual is advantageous for a group that would like to have additional members and can be helpful to an individual who is looking to find other individuals or a group of people with similar interests or goals.

4.2 Implement Data mining in experiment

The analytics platform at Twitter has experienced tremendous growth over the past few years in terms of size, complexity, number of users, and variety of use cases. In 2010, there were approximately 100 employees in the entire company and the analytics team consisted of four people today, the company has over one thousand employees [19].

Typically, after exploratory data analysis, the data scientist is able to more precisely formulate the problem, cast it in within the context of a data mining task, and define metrics for success. For example, one way to increase active user growth is to increase retention of existing users (in addition to adding new users): it might be useful to build a model that predicts future user activity based on present activity. With a precisely-formulated problem in hand, the data scientist can now gather training and test data.

As mentioned before, data mining could be used to find influential users as well. The basic assumption is that when users see their social contacts performing an action they may decide to perform the action themselves. In truth, when users perform an action, they may have any one of a number of reasons for doing so: they may have heard of it outside of the online social network and may have decided it is worthwhile, the action may be very popular, or they may be genuinely influenced by seeing their social contacts perform that action. [20]

Having all these in mind, data mining in our experiment would be an interesting addition. All the valuable data that we collected as shown, can be used in order to cluster the users based on their interests and habits. This can be done by using Data mining tools such as Rapidminer. Rapidminer provides data mining techniques for text processing, Web mining, etc.

However, further analysis is needed, such as hashtags that contained etc. Initially, in order to model training of users, we will use data that collected earlier and especially tweets. Twitter4j has methods in order to collect columns such as tweetText, tweetId, tweetUser, tweetTime, getSource, getURLEntities, getHashTagEntities, userRT, userTalk. This will help us create a vocabulary of words that users use in their tweets.

A tweet can, i.e. contain urls and/ or hashtags, be reposted by another user or even mention a user. In order to achieve better education model, each tweet recovered distinguish tweets that contain links, hashtags, the reference to other users and the usernames of the users of which was reposted.

As soon as we cluster the users, a personalized phishing attack can launched, depending on their interests. In current version, the prize that used as a "bait" is a 2days stay in a hotel. However if users interested in, i.e sports, the bait could be tickets for an important match. By personalizing the phishing attack, the success rates expected to be higher.

Another benefit that would arise from the use of data mining in our experiment would be the information extraction of a phishing attack.

As shown in [21], gender and age are two major factors that affect phishing rates. Specifically, women tend to click on phishing links more often than men do, and also they tend to give more often information to phishing websites. This happens because women have less technical training and less technical knowledge than men.

Another factor is the age: participants aged between ages 18 and 25 are much more likely than others to fall for phishing (as seen by other researchers). This group appears to be more susceptible because participants in this age group have a lower level of education, fewer years on the Internet, less exposure to training materials, and less of an aversion to risks.

By using data mining we could extract information about the habits of male and female, categorize them by age and launch an attack depending on these habits.

Another idea is to launch an attack to specific genders or specific age groups that surveys have proved that are more likely to fall for phishing, instead of using influencer's algorithm, or even combine data mining with our algorithm.

5. FUTURE WORK

During the development and implementation of the crawler, the initial idea was to crawl personal information of a great number of users. However, as mentioned in the Challenges section, this is a process that demands a remarkable amount of time. Therefore, in order to reduce the time needed, the implementation could work in threads. By doing this, all the individual tasks, such as tweet and followers crawling can run simultaneously.

Furthermore, Twitter was able to know every time we tried to send spam tweets to the victims and banned our account temporarily. In order to avoid being banned again, someone could create more similar accounts and send spam tweets from different Twitter accounts.

Twitter API allows a developer to gather emails given their usernames. However this demands the authenticated application to be whitelisted. If an attacker can trick Twitter in order to gain these privileges, then spam messages can also sent to their email accounts.

As for the Data Mining part, the ideal way to implement user clustering is by categorizing them based on their interests. After the categorization of the users, we can send personalized spam messages to every category, depending on their interests.

Finally, some improvements could benefit the website as well. In this current version, the url is https://aegean.000webhostapp.com . It is hosted in 000webhost, a free host. As a result, the url isn't the most realistic one and in our website there is a watermark of the host. In order to be more believable, we could buy a new domain, like aegeanmembersclub.gr, uoaclub.gr etc. By doing this, the result would be more realistic and the phishing rates may improve.

6. CONCLUSIONS

From the experience gained throughout the implementation phase of this thesis one can safely argue that SNS suffer from malicious attacks. Once more, in that kind of attack users are considered to be the most critical part for the success rate of the phishing attack.

As for the crawling part, it was really hard to crawl user personal information, as Twitter is trying really hard to protect the rights of its users and keep their personal information safe. However, that does not mean that it is impossible to have access to this information, something that has been proved by this thesis. In any case, there is always a way for a malicious user to overcome the security of a system and reach its sensitive information.

Furthermore, Twitter was able to detect the spam tweets and after some point, it temporarily banned the account for illegal actions. However, it was really easy to unblock the account, as it was only required an email/sms verification. Twitter didn't detect anything during the crawling procedure, so an attacker would have the time to gather enough information.

Apart from the measures that Twitter should take, there are ways for users to protect themselves. It is impossible to craw user information if the profile is private. In a nutshell, the most important conclusion of this thesis is that the crawler was not able to gather information about users that chose their profile to be protected. Actually, this is the best way for someone to protect its personal information from malicious users and of course, not click on every link that is posted by unknown and unrelated users.

To sum up, the major factor of a phishing attack is the human. Users should be more concerned about their privacy, as internet in general and social media networks specifically, is full of malicious users trying to obtain useful information. Users should never share information with strangers. The best way to do this is by keeping profiles private and share information with users they know in real life only.

Users should also be aware of fake contests and malicious websites, as they used as "bait" of a phishing attack. By using HTTPS connection and by checking the certificates of a specific site, is an effective way to check whether a site is malicious or not.

REFERENCES

[1] http://www.dictionary.com/browse/social--network

[2] https://www.statista.com/statistics/272014/global-social-networks-ranked-by-number-of-users/

[3] https://techradar1.wordpress.com/2008/01/11/facebookmyspace-statistics/

[4] https://en.wikipedia.org/wiki/Instagram

[5] A. Chowdhury, *State of Twitter Spam*, Mar. 2010, accessed Jan. 14, 2014 [Online]. Available: https://blog.twitter.com/2010/state-twitter-spam

[6] L. Tristan, Twitter's Growing Spam Problem, Forbes, Jul. 2013, accessed Mar. 3, 2014 [Online]. Available: http://www.forbes.com/sites/tristanlouis/2013/04/07/twitters-growing-spam-problem/

[7] https://samy.pl/popular/tech.html

[8] https://en.wikipedia.org/wiki/Koobface

[9] Abraham, D., & Raj, N. S. (2014, September). *Approximate string matching algorithm for phishing detection*. In Proceeding of International Conference on Advances in Computing, Communications and Informatics (ICACCI) New Delhi, India: IEEE, 2285-2290

[10] http://docs.apwg.org/reports/apwg_trends_report_q1_2013.pdf

[11] Anupama Aggarwal, Ashwin Rajadesingan, Ponnurangam Kumaraguru (2012, October) *PhishAri: Automatic Realtime Phishing Detection on Twitter*

[12] Noro, T. , Ru, F. , Xiao, F. , & Tokuda, T. (2012). *Twitter user rank using keyword search*. In P. Vojtás, Y. Kiyoki, H. Jaakkola, T. Tokuda, & N. Yoshida (Eds.), Information modelling and knowledge bases XXIV, 22nd european-japanese conference on information modelling and knowledge bases (EJC 2012), prague, czech republic, june, 4-9, 2012 . In Frontiers in Artificial Intelligence and Applications: 251 (pp. 31–48). IOS Press

[13] F. Riquelme, P. González-Cantergiani, *Measuring user influence on Twitter: A survey*, Information Processing and Management (2016), http://dx.doi.org/10.1016/j.ipm.2016.04.003

[14] Nagmoti, R., Teredesai A., & De Cock, M. (2010). *Ranking approaches for microblog search*. In J. X. Huang, I. King, V. V. Raghavan, & S. Rueger (Eds.), 2010 IEEE/WIC/ACM international conference on web intelligence, WI 2010, Toronto, Canada, August 31 - September 3, 2010, main conference proceedings (pp. 153–157). IEEE Computer Society

[15] Cappelletti, R., & Sastry, N. (2012). *IARank: Ranking users on twitter in near real-time, based on their information amplification potential*. In 2012 international conference on social informatics (socialinformatics), Washington, d.c., USA, December 14-16, 2012 (pp. 70–77). IEEE Computer Society

[16] Hajian, B., & White, T. (2011). *Modelling influence in a social network: Metrics and evaluation.* In PASSAT/socialcom 2011, privacy, security, risk and trust (PASSAT), 2011 IEEE third international conference on and 2011 IEEE third international conference on social computing (socialcom), boston, MA , USA , 9-11 oct., 2011 (pp. 497–500)

[17] Jin X., & Wang Y. (2013). *Research on social network structure and public opinions dissemination of micro-blog based on complex network analysis*. Journal of Networks, 8 (7), 1543–1550

[18] C. C. Aggarwal (2011), *Social Network Data Analytics*, DOI 10.1007/978-1-4419-8462-3_12, © Springer Science+Business Media

[19] Jimmy Lin & Dmitriy Ryaboy (2013), *Scaling Big Data Mining Infrastructure: The Twitter Experience*

[20] Francesco Bonchi (2011), *Influence Propagation in Social Networks: A Data Mining Perspective*

[21] Steve Sheng, Mandy Holbrook, Ponnurangam Kumaraguru, Lorrie Cranor, Julie Downs (2010). *A Demographic Analysis of Phishing Susceptibility and Effectiveness of Interventions*, Carnegie Mellon University, Indraprastha Institute of Information Technology

APPENDIX – IMPORTANT PIECES OF CODE

Code 1:

```
public static OAuth2Token getOAuth2Token(String Consumer, String Secret) {
        OAuth2Token token = null;
        ConfigurationBuilder cb;
        cb = new ConfigurationBuilder();
        cb.setApplicationOnlyAuthEnabled(true);
        cb.setOAuthConsumerKey(Consumer).setOAuthConsumerSecret(Secret);
        try {
            token = new
TwitterFactory(cb.build()).getInstance().getOAuth2Token();
        } catch (Exception e) {
            System.out.println("Could not get OAuth2 token");
            e.printStackTrace();
            System.exit(0);
        }
        return token;
    }

    public static Twitter getTwitter(String Consumer, String Secret) {
        OAuth2Token token;
        //   First step, get a "bearer" token that can be used for our
requests
        token = getOAuth2Token(Consumer, Secret);


//    Now, configure our new Twitter object to use application
authentication and
// provide it with  our CONSUMER key and secret and the bearer token we got
// back from Twitter


        ConfigurationBuilder cb = new ConfigurationBuilder();
        cb.setApplicationOnlyAuthEnabled(true);
        cb.setOAuthConsumerKey(Consumer);
        cb.setOAuthConsumerSecret(Secret);
        cb.setOAuth2TokenType(token.getTokenType());
        cb.setOAuth2AccessToken(token.getAccessToken());

        //   And create the Twitter object!
        return new TwitterFactory(cb.build()).getInstance();
    }
```

These functions are responsible for user authentication. They use the key pair of a specific application, in order to authenticate the application, using OAuth2 protocol. It is a necessary process if someone wants to use Twitter API

Code 2:

```
public void getTweetsOfUser(String username, String[] credentials) throws
SQLException, TwitterException {
          PreparedStatement stmt = null;
          PreparedStatement stmt2 = null;
```

```
          String name = null;
          int retweeted_statuses = 0, original_statuses = 0,
retweeted_times = 0;
          int url_counter = 0;
          String sqlc = "CREATE TABLE IF NOT EXISTS USER_TWEETS"
                  + "(USER_NAME CHAR(50)  NOT NULL,"
                  + "USER_ID INT,"
                  + "TWEET TEXT,"
                  + "foreign key(USER_ID) REFERENCES
TWITTER_MEMBER(MEMBER_ID),"
                  + "UNIQUE(USER_NAME, USER_ID, TWEET))";

           stat.executeUpdate(sqlc);
          Twitter twitter = getTwitter(credentials[8], credentials[9]);
          int pageno = 1;
          Paging page;
          int size = 0;
          List statuses = new ArrayList();

         while (true) {
             try {
                 size = statuses.size();
                 page = new Paging(pageno++, 99);
                 statuses.addAll(twitter.getUserTimeline(username, page));
                 if (statuses.size() == size) {
                     break;
                 }
             } catch (TwitterException e) {

// do not throw if user has protected tweets, or if they deleted their
account
                 if (e.getStatusCode() == HttpResponseCode.UNAUTHORIZED
                             || e.getStatusCode()
     =HttpResponseCode.NOT_FOUND)
                 {
                     break;
            // log something here
                 } else {
                     page = new Paging(pageno + 2, 99);
                     statuses.addAll(twitter.getUserTimeline(username,
page));
             //   break;
                     // e.printStackTrace();
                 }
             }
          }

          List<Status> stat = new ArrayList(statuses);
          for (Status stats : stat) {
//     This returns all the various rate limits in effect for us with the
Twitter API
              //   Map<String, RateLimitStatus> rateLimitStatus =
twitter.getRateLimitStatus("statuses");

//     This finds the rate limit specifically for doing the search API call
we use in this program
              // RateLimitStatus searchTweetsRateLimit =
rateLimitStatus.get("/statuses/user_timeline");
  //System.out.println(searchTweetsRateLimit.getRemaining());
```

```
              String text = null;
              int id;
              name = stats.getUser().getScreenName();
              text = stats.getText();
              id = (int) stats.getUser().getId();
              retweeted_times = retweeted_times + stats.getRetweetCount();
              if (stats.getRetweetCount() > 0) {
                  retweeted_statuses++;
              }
             System.out.println("@" + name + " - " + text + " - " +
stats.getRetweetCount());
              if (text.contains("www.") || text.contains("https://") ||
text.contains("http://")) {
                  url_counter++;
  //System.out.println("CONTAINS");
              }

  //     System.out.println(stats.getHashtagEntities());
  //System.out.println(replyStatus);
              stmt = conn.prepareStatement("INSERT OR IGNORE INTO
USER_TWEETS values(?,?,?)");
              stmt.setString(1, name);
              stmt.setInt(2, id);
              stmt.setString(3, text);
              stmt.executeUpdate();
              /*for (Status fav_stats : fav_stat) {
               String name=null,text=null;
               name=fav_stats.getUser().getScreenName();
               text=fav_stats.getText();
               System.out.println("@" + name +  " - "
+fav_stats.getCreatedAt() + " - " + text + " - " +fav_stats.getText());
               */
              //  if
(text.contains("www.")||text.contains("https://")||text.contains("http://")
) {
              //         System.out.println("CONTAINS");
              //         }
              //long replyStatus = stats.getInReplyToStatusId();
  //System.out.println(replyStatus);
          /*    stmt = conn.prepareStatement
               ("INSERT OR IGNORE INTO USER_TWEETS values(?,?)");
               stmt.setString(1, name);
               stmt.setString(2,text);
               stmt.executeUpdate();*/
  //}
          }
          original_statuses = size - retweeted_statuses;
          System.out.println("Total Original: " + original_statuses);
          if (original_statuses < 0) {
              original_statuses = 0;
          }
          System.out.println("Total Retweeted: " + retweeted_times);
          if (retweeted_times < 0) {
              retweeted_times = 0;
          }
          System.out.println("Total Url: " + name);
          stmt2 = conn.prepareStatement("UPDATE USER_METRICS SET
USER_ORIGINAL_TWEETS = ?, USER_URL_TWEETS = ?, USER_RETWEETED_TWEETS = ?
WHERE USER_NAME = ?");
```

```
            stmt2.setInt(1, original_statuses);
            stmt2.setInt(2, url_counter);
            stmt2.setInt(3, retweeted_times);
            stmt2.setString(4, name);
            stmt2.executeUpdate();
        }
```

This piece of code is responsible for collecting all the tweets of a user, by using username and the credential of the authenticate app. As it is impossible to collect tweets of private accounts, for all the public accounts we store username, user id and the text itself. It also checks if a tweet is original, retweet or contains url. It also updates metrics table with the total number of original tweets, retweets and url contained tweets.

Code 3:

```
        public void getInfluence() throws TwitterException, Exception {
            PreparedStatement stmt = null;
            float popularity;
            String sqlc = "CREATE TABLE IF NOT EXISTS USER_INFLUENCE"
                    + "(USER_NAME CHAR(50)  NOT NULL,"
                    + "USER_ID INT,"
                    + "GENERAL_ACTIVITY INT,"
                    + "POPULARITY FLOAT(5),"
                    + "KLOUT FLOAT(5),"
                    + "foreign key(USER_ID) REFERENCES FOLLOWERS(USER_ID),"
                    + "UNIQUE(USER_NAME))";
            stat.executeUpdate(sqlc);
            Klout k = new Klout("8hd5c89qs32kzvh52ca2bcfd");
            ResultSet rs = stat.executeQuery("SELECT *FROM USER_METRICS;");
            while (rs.next()) {
                int id = rs.getInt("USER_ID");
                String name = rs.getString("USER_NAME");
                int original = rs.getInt("USER_ORIGINAL_TWEETS");
                int retweets = rs.getInt("USER_RETWEETED_TWEETS");
                int favorites = rs.getInt("USER_FAVORITED_TWEETS");
                int followers = rs.getInt("USER_FOLLOWERS");
                int friends = rs.getInt("USER_FRIENDS");
                try {
                    int activity = original + retweets + favorites;
                    popularity = (float) followers / (float) (followers +
friends);

  // retrieves klout id with twitter screen name
                    String[]        data        =        k.getIdentity(name,
Klout.TWITTER_SCREEN_NAME); // contains ["635263", "ks]
                    wrapper.User u = k.getUser(data[0]);
                    double score = u.score();
                    stmt  =  conn.prepareStatement("INSERT  OR  IGNORE  INTO
USER_INFLUENCE values(?,?,?,?,?)");
                    stmt.setString(1, name);
                    stmt.setInt(2, id);
                    stmt.setInt(3, activity);
                    stmt.setFloat(4, popularity);
                    stmt.setFloat(5, (float) score);
                    stmt.executeUpdate();
                } catch (FileNotFoundException exception) {
                    System.out.println("Not found");
                }
```

```
        }
        rs.close();
    }
```

This piece of code calculates the influence of a specific user. It calculates the general activity and the popularity as it described in the chapter 3 and it connects with Klout, in order to obtain the klout score.

Code 4:

```
public void getInfluencers(List myList, List myList2) throws
TwitterException, Exception {
        ResultSet rs = stat.executeQuery("SELECT * FROM \n"
                + "(SELECT * FROM USER_INFLUENCE WHERE GENERAL_ACTIVITY>
(select avg(GENERAL_ACTIVITY) from USER_INFLUENCE))\n"
                + " WHERE KLOUT>20 AND POPULARITY > (select
avg(POPULARITY) from USER_INFLUENCE)");

        while (rs.next()) {
            int id = rs.getInt("USER_ID");
            String name = rs.getString("USER_NAME");
            // System.out.println("User " + name );
            myList.add(name);
  //System.out.println(score2);
        }

        ResultSet rs2 = stat.executeQuery("SELECT * FROM (SELECT * FROM
USER_INFLUENCE WHERE GENERAL_ACTIVITY<(select avg(GENERAL_ACTIVITY) from
USER_INFLUENCE)) "
                + "WHERE KLOUT<15 AND POPULARITY < (select
avg(POPULARITY) from USER_INFLUENCE) LIMIT 20");
        while (rs2.next()) {
            int id = rs.getInt("USER_ID");
            String name = rs.getString("USER_NAME");
            // System.out.println("User " + name );
            myList2.add(name);
  //System.out.println(score2);
        }
        rs.close();
    }
```

By using this piece of code, we use our algorithms presented in this paper, in order to find the influencers among our database. This is the final version, where klout score should be greater than 20.

Code 5:

```
public void sendTweet(String username, String[] credentials2) {
        ConfigurationBuilder builder = new ConfigurationBuilder();
        builder.setOAuthConsumerKey(credentials2[0]);
        builder.setOAuthConsumerSecret(credentials2[1]);
        Configuration configuration = builder.build();
        TwitterFactory factory = new TwitterFactory(configuration);
        Twitter twitter = factory.getInstance();
        AccessToken accessToken = new AccessToken(credentials2[2],
credentials2[3]);
        twitter.setOAuthAccessToken(accessToken);
```

```
          Bitly.Provider         bitly         =         as("bitlyminous",
"R_1ff9cacf41194bcaa13b8ca7ba0985d9");
          ShortenedUrl                       shortUrl                        =
bitly.call(shorten("https://aegean.000webhostapp.com/?c=" + username));
          String statusMessage = "@" + username + " Πάρτε μέρος εδώ=> " +
shortUrl.getShortUrl();
          File file = new File(".\\phishing.JPG");
          StatusUpdate status = new StatusUpdate(statusMessage);
          status.setMedia(file); // set the image to be uploaded here.
          try {
              twitter.updateStatus(status);
              System.out.println("Message sent to " + username);
              /*try {
               DirectMessage                directMessage                =
twitter.sendDirectMessage(username, message);
               System.out.println("Message      sent      to      "      +
directMessage.getRecipientScreenName());
               } catch (TwitterException ex) {

Logger.getLogger(AllTogether.class.getName()).log(Level.SEVERE, null, ex);
               }*/
              /*        try
               {
               System.out.println(twitter.getScreenName());
               Status status = twitter.updateStatus(m);
               System.out.println("Successfully updated the status to [" +
status.getText() + "].");
               }catch (TwitterException te) {
               te.printStackTrace();
               System.exit(-1);
               }*/
          } catch (TwitterException ex) {

Logger.getLogger(AllTogether.class.getName()).log(Level.SEVERE, null, ex);
          }

      }
```

This piece of code is used in order to send a tweet mention to a possible victim. This tweet consists of a user mention, a personalized malicious url and the photo about the contest.

CURRICULUM VITAE

Georgios Vlassopoulos

Georgios Vlassopoulos was born in Argos, Greece in 09/06/1993. He is currently an undergraduate student in the Department of Information and Communication Systems Engineering of University of the Aegean, Samos, Greece.