A Survey on Detecting Malicious Facebook Applications using FRAppE

*S.HEMALTHA **HARI KRISHNA

*M.TECH Dept of CSE,VAAGDEVI COLLEGE OF ENGINEERING

**Assistant Professor Dept of CSE,VAAGDEVI COLLEGE OF ENGINEERING

ABSTRACT -Facebook applications are one of the reasons for Facebook attractiveness. Unfortunately, numerous users are not aware of the fact that many malicious Facebook applications exist. With 20 million installs a day[1], thirdparty apps are a major reason for the popularity and addictiveness of Facebook. But, cyber criminals have realized the potential of using apps for spreading malware and spam like unsolicited mail. The problem is already significant, as we find that at least 13% of apps in the sample dataset are malicious. So far, the research community has focused on detecting malicious posts and campaigns. In this paper, we ask the question: given a Facebook application, can we determine if it is malicious? Our key contribution is surveying FRAppE—Facebook's Rigorous Application Evaluator—arguably the primary tool focused on detecting malicious apps on Facebook. There are 2.2 millions of people using Facebook, so in order to develop FRAppE, the information about the posting behavior of Facebook user's is observed and gathered. FRAppE is shown that it can detect malicious apps with 99.5% accuracy, with no false positives and a low false negative rate (4.1%).Strangely, it is found that many apps collude and support each other; in the dataset, it is found 1,584 apps enabling the viral propagation of 3,723 other apps through their posts. Longterm, we see FRAppE as a step towards creating an independent watchdog for app assessment and ranking, so as to warn Facebook users before installing apps.

KEYWORDS: Facebookapps, malicious, OnlineSocialNetworks, spam.

I. INTRODUCTION

Online Social Networks (OSN's) enable and inspire third-party applications (apps) to enhance the user experience on these platforms like FaceBook, Twitter. Interesting or entertaining ways of communicating among on-line friends and diverse activities such as playing games or listening to songs are examples of such enhancements. For example, Facebook provides developers an API [2] that facilitates app integration into the Facebook user experience. There are 500K apps available on Facebook [3], and on average, 20M apps are installed every day [1]. Further-more, many apps have acquired and maintain a really large user database. It has been observed that FarmVille and CityVille apps have 26.5M and 42.8M users to date. Recently, hackers and malicious users have started taking advantage of the popularity of this third-party apps platform and deploying malicious applications [4]–[6]. Malicious apps can provide a lucrative business for hackers, given the status of OSN's, with Facebook leading the way with 900M active users [7]. There are many ways that hackers can benefit from a malicious app:

a) The app can reach large number of users and their friends to spread spam.

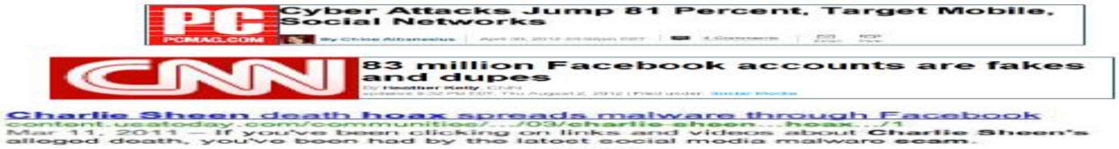
b) The app can obtain users personal information such as e-mail address, home town, and gender, and

c) The app can "reproduce" by making other malicious apps popular. In other words, there is motive and opportunity, and as a result, there are many malicious apps spreading on Facebook every day [8].

Despite the above worries, today a user has very limited information at the time of installing an app on his Facebook profile. In other words, the problem is the following: Given an app's identity number (the unique identifier assigned to the app by Facebook), can we detect if the app is malicious? Currently, there is no commercial service, publicly available information, or research-based tool to advise a user about the risks of an app. Malicious apps are widespread

and they easily spread, as an infected user jeopardizes the safety of all its friends.

So far, the researches has been done regarding spam and malware on Facebook which has focused on detecting malicious posts and social spam campaigns [9]–[11]. At the same time, in a seemingly backwards step, Facebook has dismantled its app rating functionality. A recent study has shown how app authorizations correlate to privacy risks of Facebook apps. Finally, there are some community based feedbacks driven efforts to rank applications, such as WhatApp? [12]; though these could be very powerful in the future, so far they have received little acceptance. The Fig.1 shows how the social malware is rampant on Facebook.



Cyber Attacks Jump 81 Percent, Target Mobile, Social Networks

By Chloe Albanesius April 30, 2012

83 million Facebook accounts are fakes and dupes

By Heather Kelly, CNN
updated 5:32 PM EDT, Thu August 2, 2012 | Filed under: Social Media

Charlie Sheen death hoax spreads malware through Facebook
content.usatoday.com/communities/.../03/charlie-sheen...hoax.../1
Mar 11, 2011 – If you've been clicking on links and videos about Charlie Sheen's alleged death, you've been had by the latest social media malware scam.

Fig.1

II.RELATED WORK

1) Detecting and Characterizing Social Spam Campaigns

Authors: Hongyu Gao, Jun Hu, Christo Wilson,Zhichun Li, Yan Chen, Ben Y. Zhao.

Description: Authors presented a primary study to calculate and analyze spam campaigns launched on online social networks. They calculated a huge anonymized dataset of asynchronous "wall" messages in between Facebook users. System detected generally 200,000 malicious wall posts with embedded URLs, originating from more than 57,000 user accounts. Authors found that more than 70% of all malicious wall posts advertise phishing sites. To study the distinctiveness of malicious accounts, and see that more than 97% are compromised accounts, rather than "fake" accounts formed solely for the principle of spamming. Finally, when adjusted to the local time of the sender, spamming dominates actual wall post in the early morning hours when users are normally asleep.

2) Is this App Safe? A Large Scale Study on Application Permissions and Risk Signals

Authors: Pern Hui Chia, Yusuke Yamamoto, N.Asokan

Description: Third-party applications capture the attractiveness of web and platforms providing mobile application. Many of these platforms accept a decentralized control strategy, relying on explicit user consent for yielding permissions that the apps demand. Users have to rely principally on community ratings as the signals to classify the potentially unsafe and inappropriate apps even though community ratings classically reflect opinions regarding supposed functionality or performance rather than concerning risks. To study the advantages of user-consent permission systems through a large data collection of Facebook apps, Chrome extensions and Android apps. The study confirms that the current forms of community ratings used in app markets today are not reliable for indicating privacy risks an app creates. It is found with some evidences, indicating attempts to mislead or entice users for granting permissions: free applications and applications with mature content request; "look alike" applications which have similar names as that of popular applications also request more permissions than is typical. Authors find that across all three platforms popular applications request more permissions than average.

3) Social Applications: Exploring A More Secure Framework

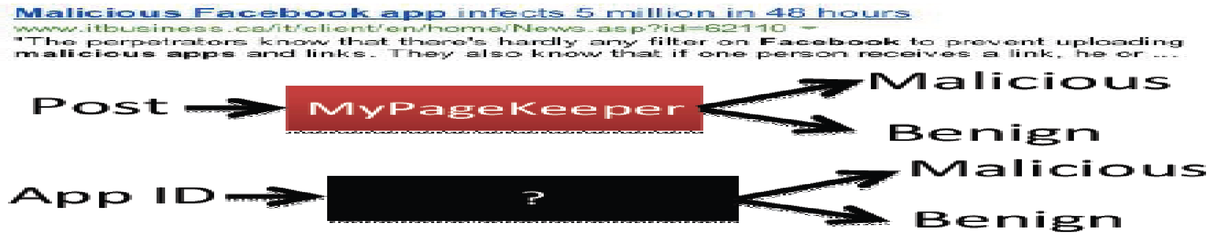
Authors: Andrew Besmer, Heather Richter Lipford,Mohamed Shehab, Gorrell Cheek

Description: OSNs such as Orkut, Facebook and others have grown-up rapidly, with hundreds to millions of active users. A new feature provided on several sites is social applications and services written by third party developers that supply additional functionality linked to a user's profile. However, present application platforms put users at risk by permitting the discovery of huge amounts of personal data and information to these applications and their developers. This paper generally abstracts main view and defines the current access control model gave to these

applications, and builds on it to generate a more secure framework.

III. BACKGROUND

To detect malicious post MyPage-Keeper is used, a security app which was launched by Facebook [13] in June 2011. It monitors the Facebook profiles of 2.2 million users. It crawls user's wall post and news feed continuously and identifies malicious posts and notifies the infected users. Over 111K apps are analyzed that made 91 million posts over 9 months. This review paper presents a comprehensive study focusing on malicious Facebook apps that focuses on quantifying, profiling, and understanding malicious apps and synthesizes this information into an effective detection approach. MyPageKeeper primarily detects malicious posts in Facebook and notify victims. The Sample dataset contains apps for which the ground truth is, they are malicious or not. For collecting sample malicious apps, we use a heurestic: if a post is flagged by MyPageKeeper as malicious which is posted by an app, they app is malicious. Then same amount of benign apps are collected to make the comparison fair. Benign apps are those apps who are not part of malicious apps and also vetted by socialbaker.com, a website that collects app statistics. But the major enabling factor is malicious Facebook app.Fig.2 shows the news about Malicious Facebook app infections and the need for malicious facebook app identification.



Malicious Facebook app infects 5 million in 48 hours
www.itbusiness.ca/it/client/en/home/News.asp?id=62110
"The perpetrators know that there's hardly any filter on Facebook to prevent uploading malicious apps and links. They also know that if one person receives a link, he or ...

Fig.2

The major problem statement is to identify malicious Facebook apps given an app ID?

Facebook enables third-party developers to offer services to its users by means of Facebook applications. Unlike typical desktop and smart phone applications, installation of a Facebook application by a user does not involve the user downloading and executing an application binary. Instead, when a user adds a Facebook application to her profile, the user grants the application server: 1) permission to access a subset of the information listed on the user's Facebook profile (e.g., the user's e-mail address), and 2) permission to perform certain actions on behalf of the user (e.g., the ability to post on the user's wall). Facebook grants these permissions to any application by handing an OAuth 2.0 [14] token to the application server for each user who installs the application.

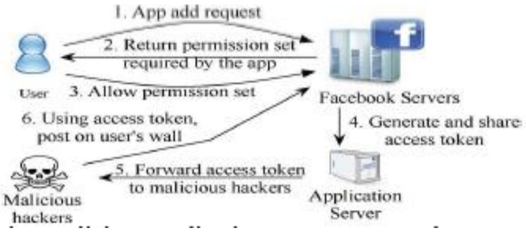

1. App add request
2. Return permission set required by the app
User
3. Allow permission set
Facebook Servers
6. Using access token, post on user's wall
4. Generate and share access token
5. Forward access token to malicious hackers
Malicious hackers
Application Server

Fig.3. Steps involved in hackers using malicious applications to get access tokens to post malicious content on victims' walls.

Thereafter, the application can access the data and perform the explicitly permitted actions on behalf of the user. Fig.3 depicts the steps involved in the installation and operation of a Facebook application.

Operation of Malicious Applications: Malicious Facebook applications typically operate as follows.

- Step 1: Hackers convince users to install the app, usually with some fake promise (e.g., free iPads).

- Step 2: Once a user installs the app, it redirects the user to a Web page where the user is requested to perform tasks, such as completing a survey, again with the lure of fake rewards.

- Step 3: The app thereafter accesses personal information (e.g., birth date) from the user's profile, which the hackers can potentially use to profit.

- Step 4: The app makes malicious posts on behalf of the user to lure the user's friends to install the same app (orsome other malicious app).

This way the cycle continues with the app or colluding apps reaching more and more users. Personal information or surveys can be sold to third parties [15] to eventually profit the hackers.Malicious hackers make posts into compromised user's wall. Their friends see the post, click the link which leads to the malicious app installation page as shown in Fig.4.Once installed, they redirect users to different pages for collecting victims personal information and Make her complete surveys so that they can earn money. Once the app is installed, hackers get permission to post any time on the victims wall. So, they make the same post and appears victims friends news feed and thus the cycle repeats and the app spreads in Facebook.

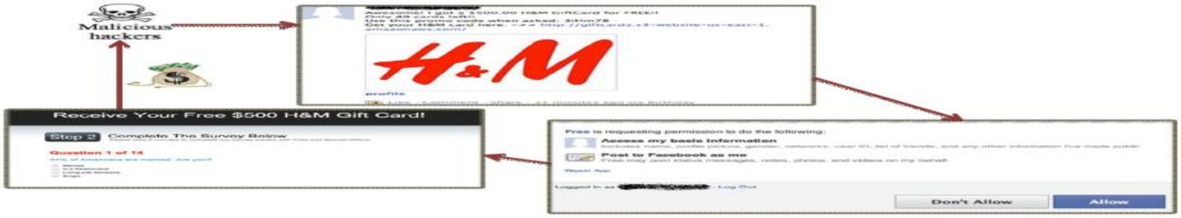

Malicious hackers
Awesome! I got a $500.00 H&M GiftCard for FREE!!
H&M
Receive Your Free $500 H&M Gift Card!
Step 2 Complete The Survey Below
Question 1 of 14
Free is requesting permission to do the following:
Access my basic information
Post to Facebook as me
Don't Allow
Allow

Fig.4

IV.PREVALENCE OF MALICIOUS APPS

The driving motivation for detecting malicious apps stems from the suspicion that a significant fraction of malicious posts on Facebook are posted by apps. It is found that 53% of malicious posts flagged by MyPageKeeper were posted by malicious apps. The prevalence of malicious apps can be quantified in two different ways.

60% of malicious apps get at least a hundred thousand clicks on the URLs they post: The malicious apps are quantified by determining a lower bound on the number of clicks on the links included in malicious posts. For each malicious app in the sample dataset, it is identified all bit.ly URLs in posts made by that application. It is focused on bit.ly URLs because bit.ly offers an API [16] for querying the number of clicks received by every bit.ly link; thus, our estimate of the number of clicks received by every application is strictly a lower bound.Across the posts made by the 6273 malicious apps in the Sample dataset, it is found that 3805 of these apps had posted 5700 bit.ly URLs in total. When queried bit.ly for the click count of each URL. Fig.5 shows the distribution across malicious apps of the total number of clicks received by bit.ly links that they had posted. We see that 60% of malicious apps were able to accumulate over 100K clicks each, with 20% receiving more than 1M clicks each.

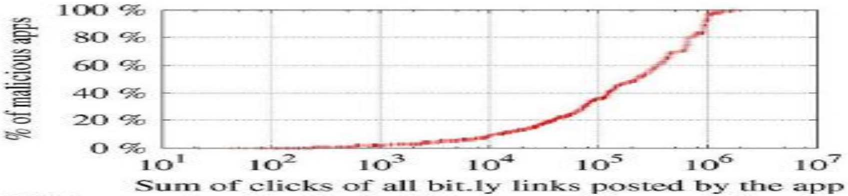


Fig. 5. Clicks received by bit.ly links posted by malicious apps.

40% of malicious apps have a median of at least 1000 monthly active users: The reach of malicious apps is examined by inspecting the number of users that these applications had. To study this, the Monthly Active Users (MAU) metrics used provided by Facebook for every application. The number of Monthly Active Users is a measure of how many unique users are engaged with the application over the last 30 days in activities such as installing, posting, and liking the app.Fig.6 plots the distribution of Monthly Active Users of the malicious apps in the sample dataset. For each app, the median and maximum MAU values over the three months are shown. We see that 40% of malicious applications had a median MAU of at least 1000 users, while 60% of malicious applications achieved at least 1000 during the 3-month observation period. The top malicious app here—”Future Teller”—had a maximum MAU of 260 000 and median of 20 000. Although it would be interesting to find the bit.ly click-through rate per user and per post, the data is not obtained for the number of users who saw these links. We can query bit.ly’s API only for the number of clicks received by a link.

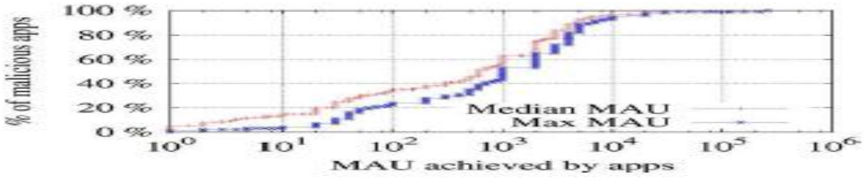


Fig. 6. Median and maximum MAU achieved by malicious apps.

This paper makes the following key contributions.

. Malicious Facebook apps are prevalent

13% of observed apps are malicious. The malicious apps are prevalent in Facebook and reach a large number of users.13% of apps in the dataset of 111K distinct apps are malicious. Also, 60% of malicious apps endanger more than 100K users each by convincing them to follow the links on the posts made by these apps, and 40% of malicious apps have over 1000 monthly active users each.

. Malicious and benign app profiles significantly differ.

A striking observation is the “laziness” of hackers; many malicious apps have the same name, as 8% of unique names of malicious apps are each used by more than 10 different apps (as defined by their app IDs). Overall, the apps can be profiled based on two classes of features:

1) Those that can be obtained on-demand given an application’s identifier (e.g., the permissions required by the app and the posts in the application’s profile page), and

2) Others that require a cross-user view to aggregate information across time and across apps (e.g., the posting behavior of the app and the similarity of its name to other apps).

Furthermore, hackers use fast-changing indirection: Applications posts have URLs that point to a Web site, and the Web site dynamically redirects to many different apps. These observed behaviors indicate well-organized crime: One hacker controls many malicious apps, which we will call an appnet, since they seem a parallel concept to botnets.

. Malicious hackers impersonate applications.

It is surprised to find popular good apps, such as FarmVille and Facebook for iPhone, posting malicious posts. On further investigation, a lax authentication rule in Facebook that enabled hackers to make malicious posts appear as though they came from these apps.

V.PROFILING MALICIOUS AND BENIGN APPS

Given the significant impact that malicious apps have on Facebook, we try to identify malicious applications. Towards this, we compare malicious and benign apps with respect to various features. When Face-book is crawled and observed several features for every application are obtained from the sample dataset. We divide these features into two subsets: on-demand features and aggregation-based features. We find that malicious applications significantly differ from benign applications with respect to both classes of features.

A. On-Demand Features

The on-demand features associated with an application refer to the features that one can obtain on demand given the application's ID. Such metrics include app name, description, category, company, and required permission set.

1) **Application Summary: Malicious apps typically have in-complete application summaries.** First, malicious and benign apps re compared with respect to attributes present in the application's summary—app description, company name, and category. Description and company are free-text attributes, either of which can be at most 140 characters. On the other hand, category can be selected from a predefined (by Facebook) list such as "Games," "News," etc., that matches the app functionality best. Application developers can also specify the company name at the time of app creation. For example, the "Mafia Wars" app is configured

with description as “Mafia Wars: Leave a legacy behind,” company as “Zynga,” and category as “Games.”

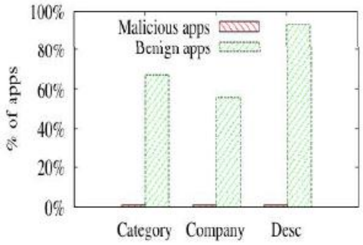
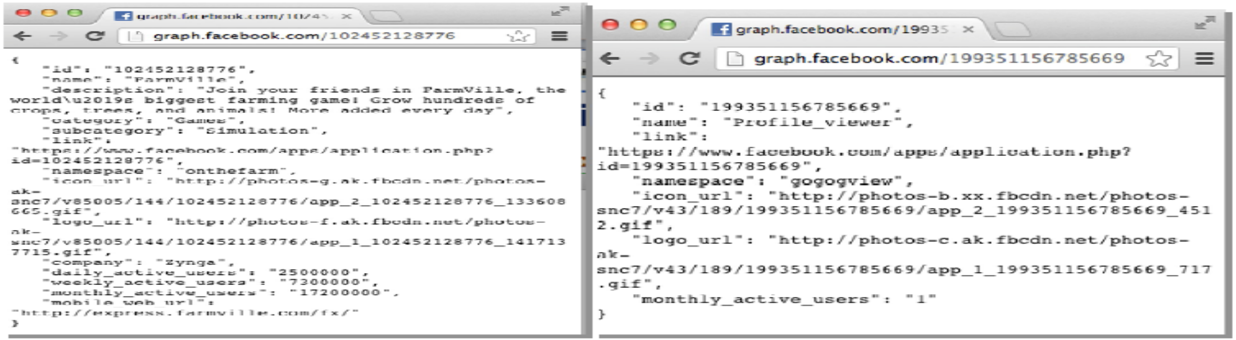


Fig.7

Fig.7 shows the fraction of malicious and benign apps in the sample dataset for which these three fields are nonempty. We see that, while most benign apps specify such information, very rarely malicious apps do so. For example, only 1.4% of malicious apps have a nonempty description, whereas 93% of benign apps configure their summary with a description. We find that the benign apps that do not configure the description parameter are typically less popular (as seen from their monthly active users). For example a popular app, FarmVille contains different information such as category, description, company etc.A malicious app “Profile_viewer” contains no such information as shown in Fig.8.

2) Required Permission Set: 97% of malicious apps require only one permission from users. Every Facebook app requires authorization by a user before the user can use it. At the time of installation, every app requests the user to grant it a set of permissions that it requires.Fig.9 shows the request for permission which by unnoticed allowed by the users.These permissions are chosen from a pool of 64 permissions predefined by Facebook [17]. Example permissions include access to information in the user’s profile (e.g., gender, e-mail, birthday, and friend list), and permission to post on the user’s wall.


graph.facebook.com/1024...
graph.facebook.com/102452128776
graph.facebook.com/19935...
graph.facebook.com/199351156785669

```
{
    "id": "102452128776",
    "name": "FarmVille",
    "description": "Join your friends in FarmVille, the
world\u2019s biggest farming game! Grow hundreds of
crops, trees, and animals! More added every day",
    "category": "Games",
    "subcategory": "Simulation",
    "link":
"https://www.facebook.com/apps/application.php?
id=102452128776",
    "namespace": "onthefarm",
    "icon_url": "http://photos-g.ak.fbcdn.net/photos-
ak-
snc7/v85005/144/102452128776/app_2_102452128776_133608
665.gif",
    "logo_url": "http://photos-f.ak.fbcdn.net/photos-
ak-
snc7/v85005/144/102452128776/app_1_102452128776_141713
7715.gif",
    "company": "Zynga",
    "daily_active_users": "2500000",
    "weekly_active_users": "7300000",
    "monthly_active_users": "17200000",
    "mobile_web_url":
"http://express.farmville.com/fx/"
}
```

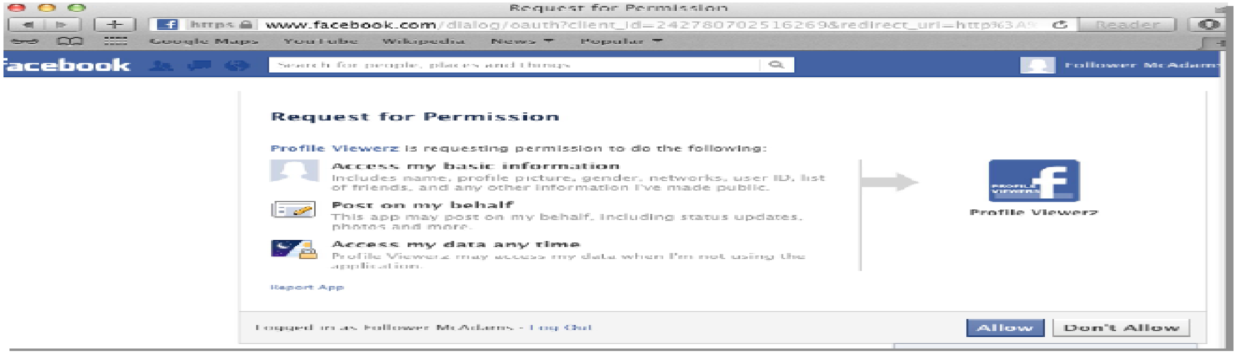
```
{
    "id": "199351156785669",
    "name": "Profile_viewer",
    "link":
"https://www.facebook.com/apps/application.php?
id=199351156785669",
    "namespace": "gogogview",
    "icon_url": "http://photos-b.xx.fbcdn.net/photos-
snc7/v43/189/199351156785669/app_2_199351156785669_451
2.gif",
    "logo_url": "http://photos-c.ak.fbcdn.net/photos-
ak-
snc7/v43/189/199351156785669/app_1_199351156785669_717
.gif",
    "monthly_active_users": "1"
}
```

Fig.8

Fig. 10 shows the top five permissions required by both benign and malicious apps. Most malicious apps in the sample dataset require only the “publish stream” permission (ability to post on the user’s wall). This permission is sufficient for making spam posts on behalf of users.

We believe that this is because users tend not to install apps that require a larger set of permissions; Facebook suggests that application developers do not ask for more permissions than necessary since there is a strong correlation between the number of permissions required by an app and the number of users who install it [18]. Therefore, to maximize the number of victims, malicious apps seem to follow this hypothesis and require a small set of permissions.

# of permissions requiredα1/# of users install



Request for Permission

Profile Viewerz is requesting permission to do the following:

Access my basic information
Includes name, profile picture, gender, networks, user ID, list of friends, and any other information I've made public.

Post on my behalf
This app may post on my behalf, including status updates, photos and more.

Access my data any time
Profile Viewerz may access my data when I'm not using the application.

Report App

Logged in as Follower McAdams - Log Out

Fig.9

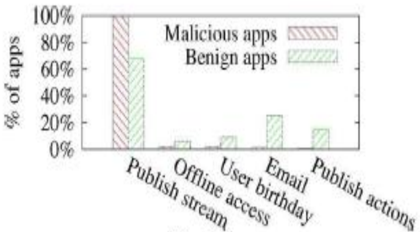


Fig.10

App installation URL contains the list of permission it requires. For example, “Profile viewez” malicious apps request for two permissions, “publish stream” which is the ability to post any time in users wall and “offline access” which gives the ability to access users data any time.

3) Redirect URI: Malicious apps redirect users to domains with poor reputation. In an application’s installation URL, the “redirect URI” parameter refers to the URL where the user is redirected to once she installs the app.The redirect URI parameter from the installation URL for apps is extracted in the sample dataset and queried the trust reputation scores for these URIs from WOT(Web Of Trust-Online) [19].WOT assigns a score between 0 and 100 for every URI,

and we assign a score of 1 to the domains for which the WOT score is not available.It is observed that 80% of malicious apps point to domains for which WOT does not have any reputation score, and a further 8% of malicious apps have a score less than 5. In contrast, it is found that 80% of benign apps have redirect URIs pointing to the apps.facebook.com domain and therefore have higher WOT scores. We speculate that malicious apps redirect users to Web pages hosted outside of Facebook so that the same spam/malicious content, e.g., survey scams, can also be propagated by other means such as e-mail and Twitter spam. Furthermore, it is found that several instances where a single do-main hosts the URLs to which multiple malicious apps redirect upon installation. For example, thenamemeans2.com hosts the redirect URI for 138 different malicious apps in the sample dataset.

4) Client ID in App Installation URL: 78% of malicious apps trick users into installing other apps by using a different client ID in their app installation URL. For a Facebook application with ID , the application installation URL is https://www.facebook.com/apps/application. php?id=A. When any user visits this URL, Facebook queries the application server registered for app to fetch several parameters, such as the set of permissions required by the app. Facebook then redirects the user to a URL that encodes these parameters in the URL. One of the parameters in this URL is the “client ID” parameter. If the user accepts to install the application, the ID of the application that she will end up installing is the value of the client ID parameter. Ideally, as described in the Facebook app developer tutorial [18], this client ID should be identical to the app ID , whose installation URL the user originally visited. However, in the sample dataset, it is found that 78% of malicious apps use a client ID that differs from the ID of the original app, whereas only 1% of benign apps do so. A possible reason for this is to increase the survivability of apps.

5) Posts in App Profile: 97% of malicious apps do not have posts in their profiles. An application’s profile page presents a forum for users to communicate with the app’s developers (e.g., to post comments or questions about the app), or vice versa (e.g., for the app’s developers to post updates about the application).Typically, an app’s profile page thus accumulates posts over time. We examine the number of such posts on the profile pages of applications in our dataset.

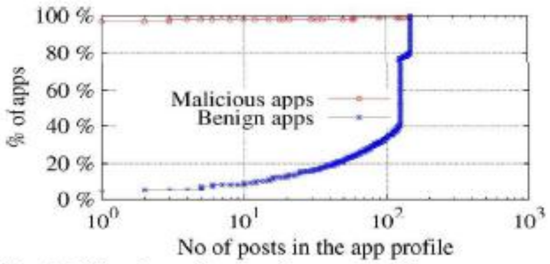


Fig.11 Number of posts in app profile page.

From Fig. 11, which shows the distribution of the number of posts found in the profile pages for benign and malicious apps, it is identified that 97% of malicious apps do not have any posts in their profiles. For the remaining 3%, their profile pages include posts that advertise URLs pointing to phishing scams or other malicious apps. For example, one of the malicious apps has 150 posts in its profile page, and all of those posts publish URLs pointing to different phishing pages with URLs such as http://2000forfree.blogspot.com and http://free-offers-sites.blogspot.com/. Thus, the profile pages of malicious apps either have no posts or are used to advertise malicious URLs, to which any visitors of the page are exposed.

B. Aggregation-Based Features

Next, we analyze applications with respect to aggregation-based features. Unlike the features we considered so far, aggregation based features for an app cannot be obtained on demand. Instead, we envision that aggregation-based features are gathered by entities that monitor the posting behavior of several applications across users and across time. Entities that can do so include Facebook security applications installed by a large population of users, such as MyPageKeeper, or Facebook itself. Here, we consider two aggregation-based features: similarity of app names, and the URLs posted by an application over time. We compare these features across malicious and benign apps.

1) App Name: 87% of malicious apps have an app name identical to that of at least one other malicious app. An application's name is configured by the app's developer at the time of the app's creation on Facebook. Since the app ID is the unique identifier for every application on Facebook, Facebook does not impose any restrictions on app names. Therefore, although Facebook does warn app developers not to violate the trademark or other rights of third parties during app configuration, it is possible to create multiple apps with the same app name. Hackers can also attempt to "typo-squat" on the names of popular benign applications. For example, the malicious application "FarmVile" attempts to take advantage of the popular "FarmVille" app name, whereas the "Fortune Cookie" malicious application exactly copies the popular "Fortune Cookie" app name. However, it is found that a large majority of malicious apps in sample dataset show very little similarity with the 100 most popular benign apps in our dataset.So the sample data seems to indicate that hackers creating several apps with the same name to conduct a campaign is more common than malicious apps typo-squatting on the names of popular apps.

2) External Link to Post Ratio: Malicious apps often post links pointing to domains

outside Facebook, whereas benign apps rarely do so. Any post on Facebook can optionally include an URL. When the URLs included in posts made by malicious and benign apps are analyzed, for every app in our sample dataset, the posts seen by MyPageKeeper and the URLs seen across these posts are aggregated. Every URL pointing to a domain out-side of facebook.com is called as an external link.

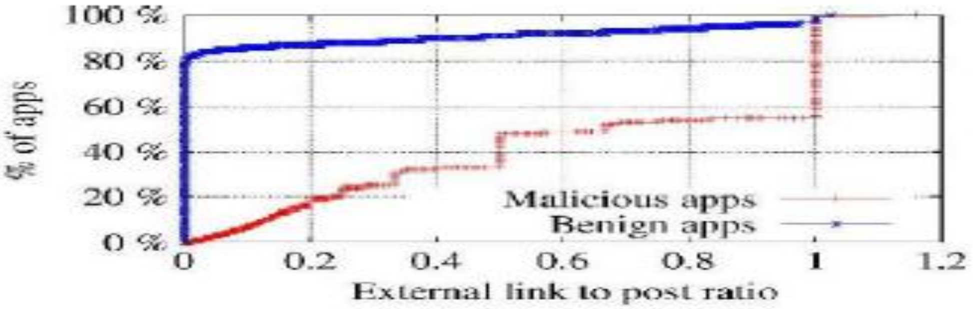


Fig. 12. Distribution of external links to post ratio across apps.

The “external-link-to- post ratio” measure is defined for every app as the ratio of the number of external links posted by the app to the total number of posts made by it.

External-link-to-post-ratio= # of external links /Total # of posts by the app

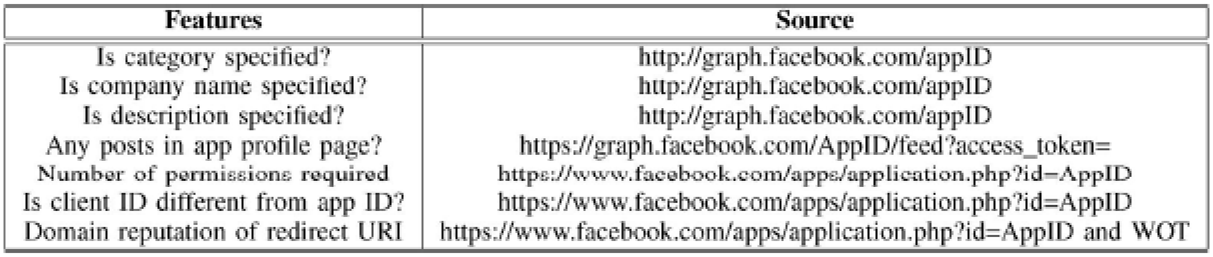
Fig.12 shows that the external link to post ratios for malicious apps are significantly higher than those for benign apps. We see that 80% of benign apps do not post any external links, whereas 40% of malicious apps have one external link on average per post. This shows that malicious apps often attempt to lead users to Web pages hosted outside Facebook, whereas the links posted by benign apps are almost always restricted to URLs in the facebook.com domain.

VIFRAppE: DETECTING MALICIOUS APPS

After analyzing the differentiating characteristics of malicious and benign apps, these features are used to develop efficient classification techniques to identify malicious Facebook applications. Here two variants of malicious app classifier—FRAppE Lite and FRAppE are presented.

A. FRAppE Lite

FRAppE Lite is a lightweight version that makes use of only the application features available on demand. Given a specific app ID, FRAppE Lite crawls the on-demand features for that application and evaluates the application based on these features in real time. FRAppE Lite can be incorporated, for example, into a browser extension that can evaluate any Facebook application at the time when a user is considering installing it to her profile. Table I lists the features used as input to FRAppE Lite and the source of each feature. All of these features can be collected on demand at the time of classification and do not require prior knowledge about the app being evaluated.



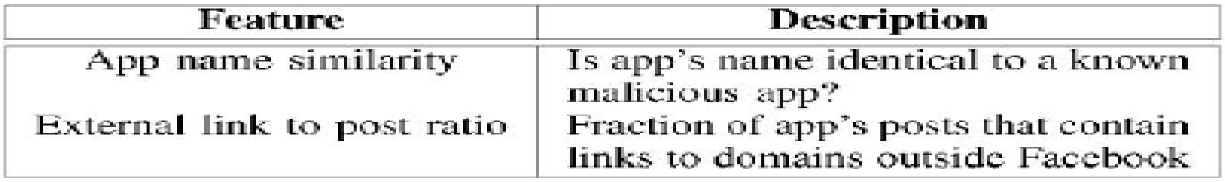
Features	Source
Is category specified?	http://graph.facebook.com/appID
Is company name specified?	http://graph.facebook.com/appID
Is description specified?	http://graph.facebook.com/appID
Any posts in app profile page?	https://graph.facebook.com/AppID/feed?access_token=
Number of permissions required	https://www.facebook.com/apps/application.php?id=AppID
Is client ID different from app ID?	https://www.facebook.com/apps/application.php?id=AppID
Domain reputation of redirect URI	https://www.facebook.com/apps/application.php?id=AppID and WOT

TABLE I List of Features Used in FRAppE Lite

Support Vector Machine (SVM) [20] is used in classifying malicious apps. SVM is widely used for binary classification in security and other disciplines [21], [22].5-fold cross validation is used on the sample dataset for training and testing FRAppE Lite’s classifier. In 5- fold cross validation, the dataset is randomly divided into five segments, and tested on each segment independently using the other four segments for training. Accuracy, false positive (FP) rate, and true positive (TP) rate are used as the three metrics to measure the classifier’s performance.Accuracy is defined as the ratio of correctly identified apps (i.e., a benign/malicious app is appropriately identified as benign/malicious) to the total number of apps. False positive rate is the fraction of benign apps incorrectly classified as malicious, and true positive rate is the fraction of benign and malicious apps correctly classified (i.e., as benign and malicious, respectively). It is expected that FRAppE Lite offer roughly 99.0% accuracy with 0.1% false positives and 95.6% true positives in practice.It can be used on user-side.

B. FRAppE

Next, we consider FRAppE a malicious app detector that utilizes aggregation-based features in addition to the ondemand features. Table.II shows the two features that FRAppE uses in addition to those used in FRAppE Lite. Since the ag-gregation- based features for an app require a cross-user and cross-app view over time, in contrast to FRAppE Lite, we en-vision that FRAppE can be used by Facebook or by third-party security applications that protect a large population of users



Feature	Description
App name similarity	Is app’s name identical to a known malicious app?
External link to post ratio	Fraction of app’s posts that contain links to domains outside Facebook

Table II.Additional Features used in FRAppE

Here, we again conduct a 5- fold cross validation with the D-Complete dataset for various ratios of benign to malicious apps. In this case, it is found that, with a ratio of 7:1 in benign to malicious apps, FRAppE's additional features improve the accuracy to 99.5% (true positive rate 95.1% and true negative rate 100%), as compared to 99.0% with FRAppE Lite. Furthermore, the true positive rate increases from 95.6% to 95.9%, and there is no single false positive.

The expected output from the above two methods is the user on facebook can only get request from benignapps and they provide security to user profiles from malicious apps.

Finally we try to conclude with the Recommendations to Facebook. The most important message of the review work is that there seems to be a parasitic eco-system of malicious apps within Facebook that needs to be understoodand stopped. However, even this initial work leads to the following recommendations for Facebook that could potentially be useful to other social platforms.

1) Breaking the cycle of app propagation. We recommend that apps should not be allowed to promote other apps. This is the reason that malicious apps seem to gain strength by self-propagation. Note that we only suggested against a special kind of app promotion where the user clicks the app A installation icon, app A redirects the user to the intermediate installation page of app B, and the user cannot see the difference unless she examines the landing URL very carefully where client ID is different. At the end, the user ends up installing app B although she intended to install app A. Moreover, cross promotion among apps is forbidden as per Facebook's platform policy [23].

***2)* Enforcing stricter app authentication before posting.** We recommend a stronger authentication of the identity of an app before a post by that app is accepted. As we saw, hackers fake the true identify of an app in order to evade detection and appear more credible to the end user.

VII.CONCLUSION

This paper is written as a survey of the base paper "Detecting Malicious Facebook Applications" by Sazzadur Rahman, Ting-Kai Huang, Harsha V. Madhyastha, and Michalis Faloutsos. Applications present convenient means for hackers to spread malicious content on Facebook. However, little is understood about the characteristics of malicious apps and how they operate. In this paper, an analysis of a large corpus of malicious Facebook apps is observed and it is found that malicious apps differ significantly from benign apps with respect to several features. For example, malicious apps are much more likely to share names with other apps, and they typically request fewer permissions than benign apps. Leveraging our observations, FRAppE is developed, an accurate classifier for detecting malicious Facebook applications. We hope that Facebook will benefit from our recommendations for reducing the menace of hackers on their platform.

REFERENCES

[1] C. Pring, "100 social media statistics for 2012," 2012 [Online].

[2] Facebook,PaloAlto,CA,USA, "Facebook Opengraph API," [Online].

[3] "Wiki: Facebook platform," 2014 [Online]. Available: http://en. wikipedia.org/wiki/Facebook_Platform

[4] "Pr0file stalker: Rogue Facebook application," 2012 [Online].

[5] "Which cartoon character are you—Facebook survey scam," 2012 [Online].

[6] G. Cluley, "The Pink Facebook rogue application and survey scam," 2012 [Online].

[7] D. Goldman, "Facebook tops 900 million users," 2012 [Online].

[8] HackTrix, "Stay away from malicious Facebook apps," 2013 [Online].

[9] M. S. Rahman, T.-K. Huang, H. V. Madhyastha, and M. Faloutsos, "Efficient and scalable socware detection in online social networks," in *Proc. USENIX Security*, 2012, p. 32.

[10] H. Gao *et al.*, "Detecting and characterizing social spam campaigns," in *Proc. IMC*, 2010, pp. 35–47.

[11] H. Gao, Y. Chen, K. Lee, D. Palsetia, and A. Choudhary, "Towards online spam filtering in social networks," in *Proc. NDSS*, 2012.

[12] "WhatApp? (beta)—A Stanford Center for Internet and Society Website with support from the Rose Foundation," [Online].

AUTHOR 1:-

* S.Hemalatha completed her B-Tech in S.R Engineering College in 2014 and pursuing M-Tech in Vaagdevi College of Engineering

AUTHOR 2:-

**Mr.Hari Krishna is working as Assistant Professor in Dept of CSE, Vaagdevi College of Engineering