PEOPLE AND OS SECURITY IN SMARTPHONES

Aseel Alnabri, Mada Alshammry, Bedour Alrashidi, Kusum Yadav
College of Computer Science & Engineering, University of Hail, Saudi Arabia.
E-mail: sullya1219@gmail.com; madaalfahad@gmail.com; b.alrashidi@uoh.edu.sa; kusumasyadav0@gmail.com
(Presented at ICSC, 2019, KSA)

ABSTRACT— *The smartphone in this era is one of the necessities for people. Most of the people prefer smartphones that have good security to makes them feel safer. They depend entirely on the OS of the smartphone to protect them from dangerous permissions as it is shown by our statistics and study. There are many ways beyond system protection that access the user's data most of the users doesn't have the basic knowledge about it. To highlight this issue we will mention a case study of penetration testing. In our result, we conclude that most of the users are unaware of various methods of unauthorized access possibilities of their smartphones.*

Keywords- Cybersecurity; Smartphones OS; Penetration Test; Social Engineering

I. INTRODUCTION

Many people trust their OS security in smartphones, but they do not know that each system even if it has a high level of protection it full of gaps. Through those gaps, it can be easy to access the smartphone. for example, iOS is known for its security but it's not enough because there is no system free of error even iOS. There are much application asks for permission to important information in user smartphones which the user does not know how dangerous this permission is for example location, phone book and gallery and the most dangerous one of them. Also, the play store is filled with of virulent application which can gain gateway to information. This information can then pursue to go across the network for virulent use. The application can also forcefully navigate through users to phishing sites and have the accesses to bypass the two-step authentication process used to access an ever-increasing number of online services such as email or banking [1].

If we assume that the applications have high protection, Are the people safe from hacking?

social engineering takes advantage of people that have less knowledge of the system they using. Some hackers use this to take from their targets important information such as IP address and their account password. Therefore no matter how powerful the os security is because it can be hacked in several ways.

II. RELATED WORK

For hacking, many doors may arise from poor system or lack of security awareness for people or even malicious programs and network surveillance.

Does the system solve all security problems and protect users from hacking?

Let's take the most recent OS of smartphones on the scene now Android and iOS, both systems have vulnerabilities that expose users to the risk of hacking or cannot protect them from hacking attacks on social engineering sometimes.

A. Android Security

Due to its prominence for Android security also user privacy, the furlough system give attracted lots of research interests why?

There are many studies on how the furlough is used by Android applications. In [10], Barrera et al. did an analysis of the furlough based security models by analyzing 1,100 most popular Android applications using the Self-Organizing Map (SOM) algorithm. They found that among the defined permissions only a little portion of them are used actively by developers. Also, the study find that the requested permissions or accesses are not highly correlated with application categories [11].

Felt *et al.* did a survey of 100 paid and 856 free apps from the Android Market in [11].

It was observed that show 93% of free and 82% of paid apps have at least one serious permission. They also declare a tool called (Stowaway) that detect whether a compiled Android application requests much permissions than it necessary i.e. overprivileged [12].

Among the application they verification, about one-third were actually overprivileged. In [13], Wei *et al.* studied the permission development in the Android ecosystem. One of their key remarking is that the set of serious level license always outnumbers other license types in all versions of the Android platform and it is still increasing. Frank et al. studied the permission demand patterns of Android apps using pattern mining technique [14].

They have an attempt to relate the permission request pattern with the application reputation which can be served as a pointer of application goodness. Although all these works revealed something about the permission request patterns of Android application they didn't a trial to identify prospect malware. Latterly, the applications ear request manner has been used to beget a risk signal for warning prospect malicious activities. Enck et al. proposed an alight heaviness application certification employ called (Kirin) that uses a rule-based strategy to distinguish suspicious application based on their demand permissions[15].

However, because the principles were founded manually they can't conform to the changing advantage of influx permissions and application. For example, the 9th rule of Kirin is no longer valid because the permission Set Preferred Application has been deprecated since Android API level 7. In [16], Zhou *et al*. suggest a system called DroidRanger to find malicious application in formal and alternative for Android store. The first ingredient in DroidRanger permission-based filtering which uses some serious permissions such as RECEIVES SandSENDS to find possibility malicious applications[6].

It shows that only 0.66% of the application needed further analyses after the permission-based filtering step. Chia *et al.* They scanned several signals overall the adjusted community rating, the availability of the developer's website and the number of apps published by the developer. However, none of those signals was found to be trustworthy. In [18], Sarma *et al.* suggest a set of risk signals by checking the permission request manner from applications in the Android store and the collected malignant application. The suggest risk signals contain rare critical permissions RCP, rare pairs of critical permissions

(RPCP), a combination of RCP and (RPCP), and category-based RCP (CRCP). The RCP signal is made if at least one of the critical permissions is requested by not more than a certain percentage of the Android Market apps. The RPCP signal is triggered if, for a pair of critical permissions, any single permission occurs more frequently than they occur as a pair. The (CRCP) signals the combination of category input with RCP. Though RCP has shown outstanding performance compared with Kirin in terms of warning and detection rates [18], the users don't have any idea about the risk levels of requested permissions by an application and the application itself as there is no quantitative assessment of the risk levels.

B. iOS Security

No OS free from error even iOS. The third-party application which is Standalone programs their disadvantages are more than their benefits for examples Jelbrak. iOS application sandbox is beheld as the fundamental mechanisms that protect users from security and privacy take advantage of that. Each iOS third party application is required to do a vetting process before published on the official iTunes application Store which is the only source of obtaining applications without jailbreaking an iOS device. Although the details of the vetting process are still secret, it is mostly regarded as highly eff ective since no harmful malware on non-jailbroken devices has been reported on the iTunes application Store [19].

Only (gray wares) which stealthily collect important user data, were found on iTunes Store. These (gray wares) were immediately removed from the store upon discovery [19] When an application is downloaded and installed on an iOS device, it is given a limited set of privileges [19], which are enforced by the application sandbox. With the sandbox restrictions, an application cannot access files and folders of other applications. In order to gateway the required user data or control system hardware (e.g. Bluetooth or WiFi), applications need to call respective iOS APIs which are hooked by the sandbox so that validations of these API invocations are performed dynamically. However, including vetting process and application sandbox is not officially documented. As a result, it shows there is no systematic security analysis even iOS platform, people believed as one of the most secure operating systems [19].

We provide an attack vector which exploits the weaknesses of both vetting process and iOS application sandbox. The attack vector consists of two attack stages and used to construct displeasure attacks that work on non-jailbroken iOS devices. We include seven proof-of-concept attacks with the attack vector proposed. We embed these attack codes into multiple applications we implemented and all the applications are able to pass the vetting process and appear on official iTunes Store.

C. Social Engineering

Social engineering is effective because of people less knowledge in security [4].

A proficient social engineer has the capability to establishing confidence and usually disguise as someone the victim would trust for example employees. Much of the information important to commit social engineering aggression is publicly available. Reverse smartphone lookup directories, such as www.reversephonedirectory.com, are much free obtainable on the Internet. Once a phone number and address are gained from the vectom, other useful information can be gained easy. One common thing that social engineering technique is to call the main switchboard of an organization and ask to be transferred to an employee and that used to attack the company . Arthurs [21], give some examples of social engineering attacks:

- IT Support:

A social engineer deceives to be from the company's (IT) support group phones a user and clarify that he is locating an error in the company network and ask for some information like ID and password from an employee in the department to identify the problem. Except if the user has been duly educated in security practices, he will likely give the "trouble-shooter" the requested information[20].

- Manager:

A social engineer He represents that he is from an authority, phones the ask for assistance desk demanding to know why he cannot log in in the system with his password then ask for the password or giving him a new password. He may also menace to report the assistance desk employee to his supervisor[20].

- Trusted Third Party:

A social engineer phones the help desk allegation to be the vice president's administrative assistant. He says that the vice president was entitled to her to collect the information. If the help desk employee inaccuracy, he menace to inform the employee's supervisor[20].

III. METHODOLOGY AND EXPERIMENT

The methodology of this work can be categorized into three main parts:

A. The Way of Access Possibilities :

In the first part we study a methodology of different OS :

Fig.1. shows the iOS and Android OS can be accessed and attack by all access possibilities. After the user accepts the terms and conditions of the application's license while downloading the application either in iOS or Android-based smartphone. Both can be the victims of vulnerable access of the IP, Gallery, Contact details and location by hackers.

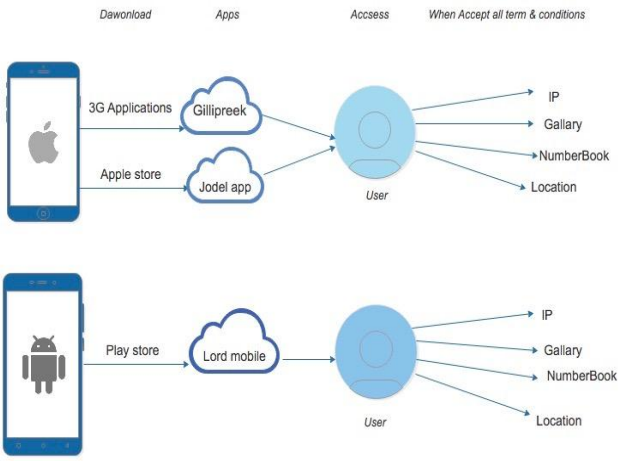


Fig. 1. The Way of Access Possibilities.

B. Case Study Of Penetration Testing:

In the second part, we highlighted the penetration testing by using Social Engineering techniques. We have 2 actors one

is a hacker and another as a user and the scenario as shown in Fig.2. as the following steps:

1. The Hacker asks the user to get the user's information by using different social engineering techniques for instance: phishing and pretexting.
2. The user as a victim provides the information asked by the hacker.
3. The hacker starts the all possible ways of penetration testing to access the user's data.

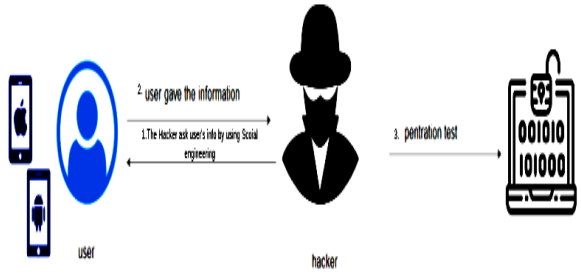


Fig. 2. Case Study Of Penetration Testing.

C. A Survey Methodology in Extent the smartphone users know the Access Possibilities:

In the third part, we design a survey to investigate the level of user's smartphones knowledge about access possibilities.

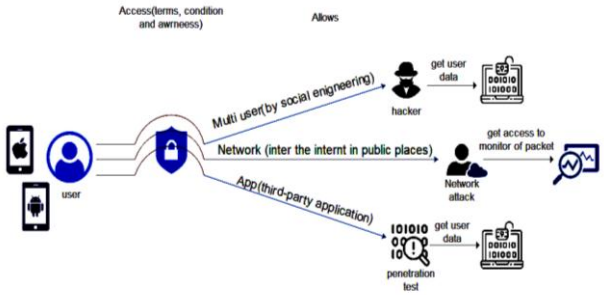


Fig. 3. A Survey Methodology in Extent the smartphone users know the Access Possibilities.

A survey was conducted for different smartphone users aged between 15 to 40 with 18 questions. In Fig. 4 we asked about the user's knowledge about cybersecurity to get a general overview. To emphasize our study that most of the users believed that the high security in smartphones especially iOS is more powerful and sufficient to protect their data. Fig.5 was an investigation about the user's opinion on smartphones OS security. To examine the user's awareness about the risks of application access possibilities Fig .6. shows different questions on access approved and agrees on terms and conditions. Fig. 7. shows the importance Survey in the user's background of networking Security. Finally, Fig. 8. illustrates A Survey in the user's knowledge about social engineering.

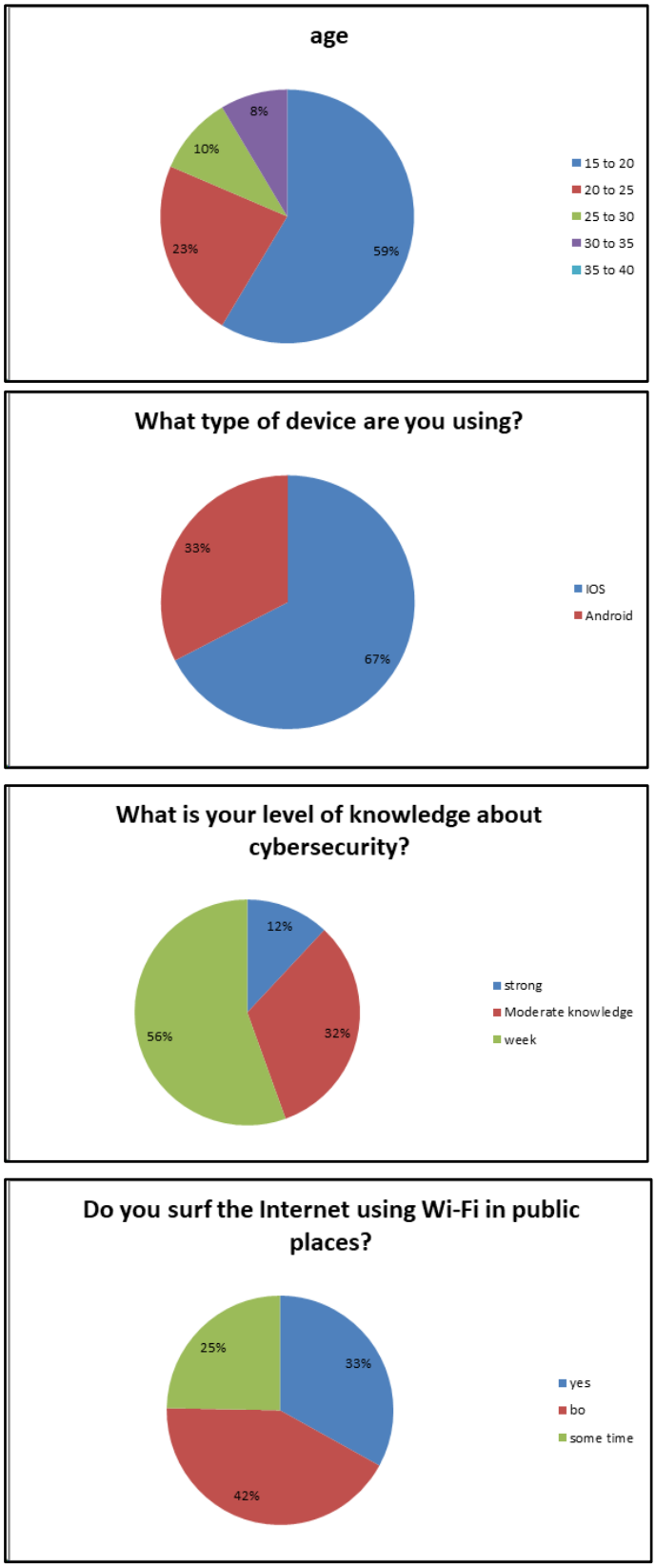
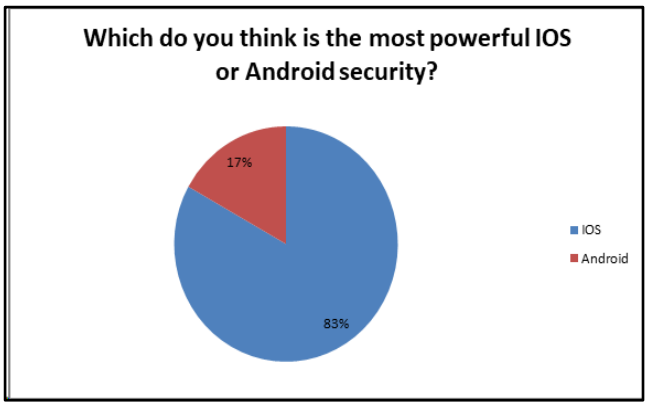
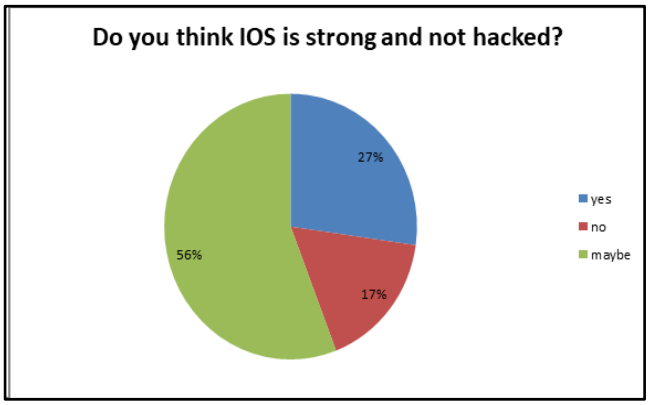
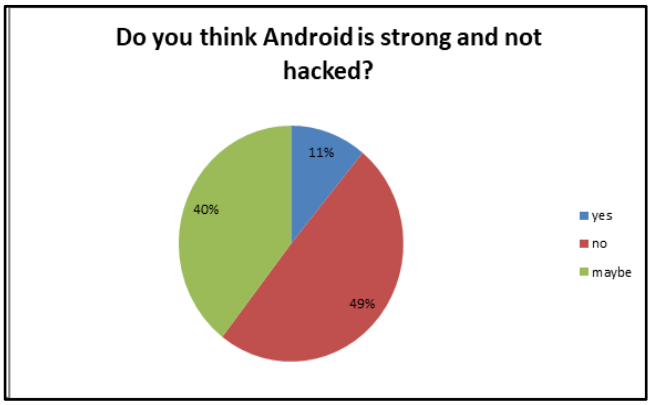


Fig. 4. A Survey in the user's background of the cybersecurity.







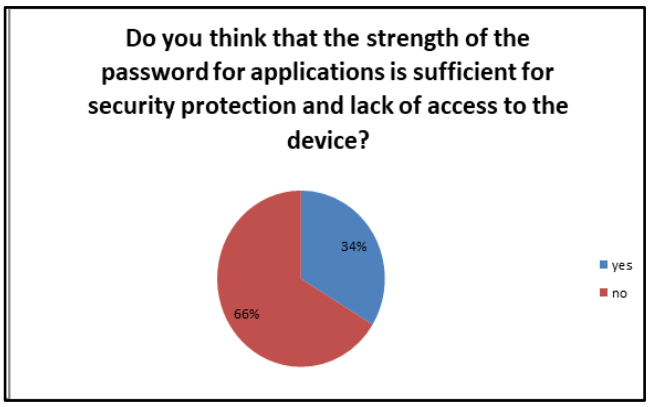
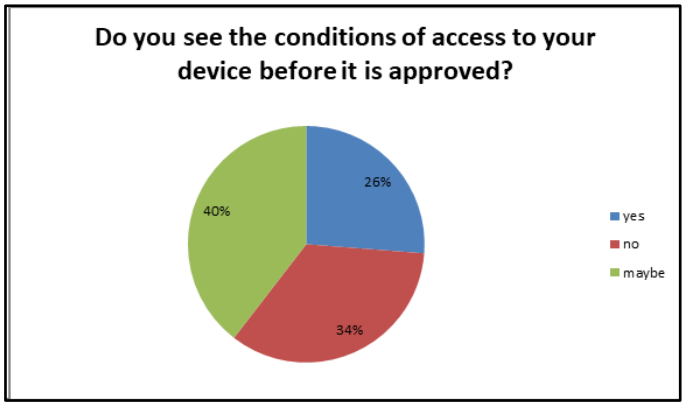
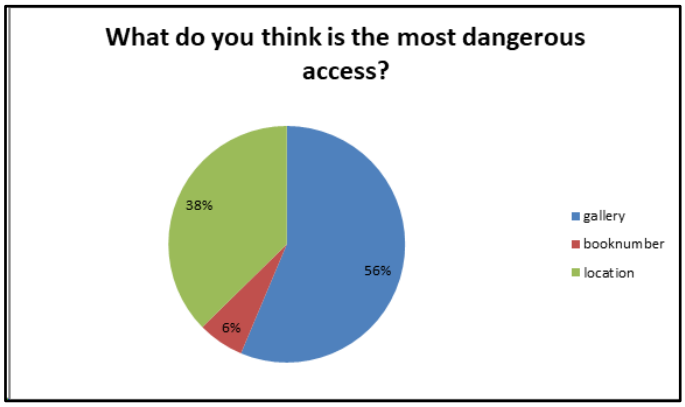
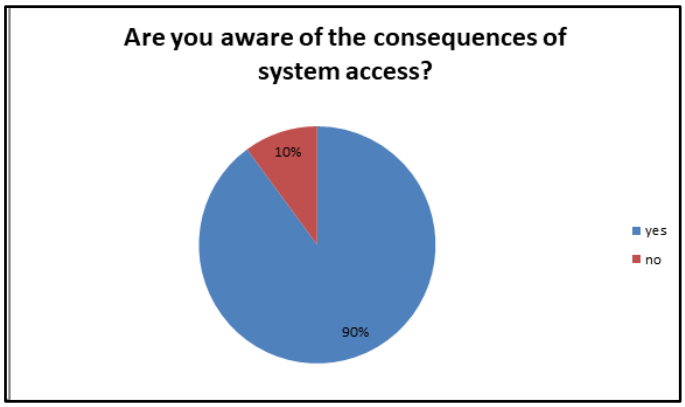


Fig. 5. A Survey in the user's opinion about smartphones OS securi*ty*.







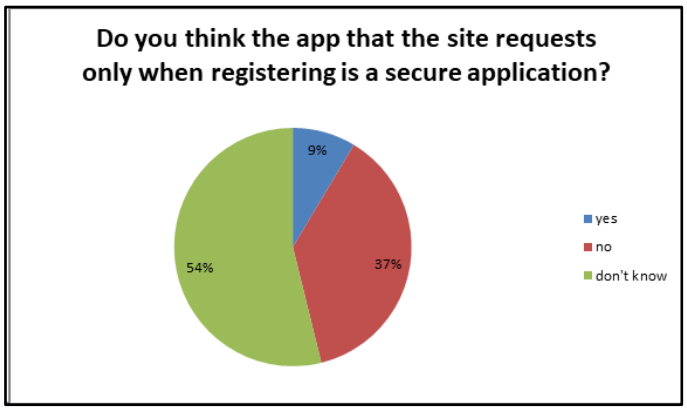
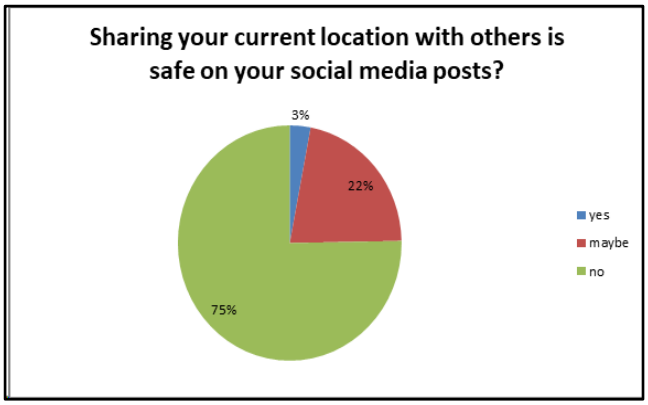
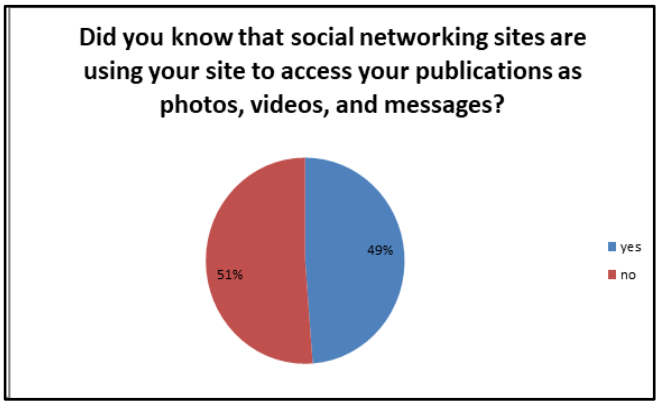
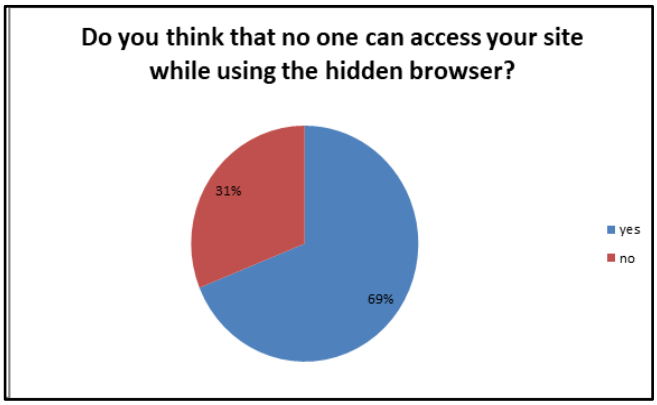


Fig. 6. A Survey in the user's awareness about application access possibilities.







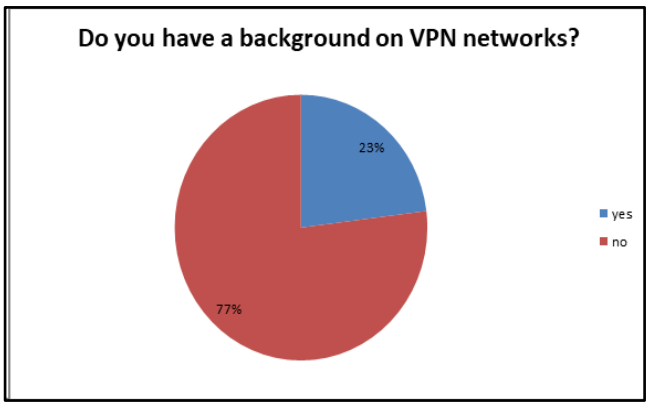


Fig. 7. A Survey in the user's background of networking Security.

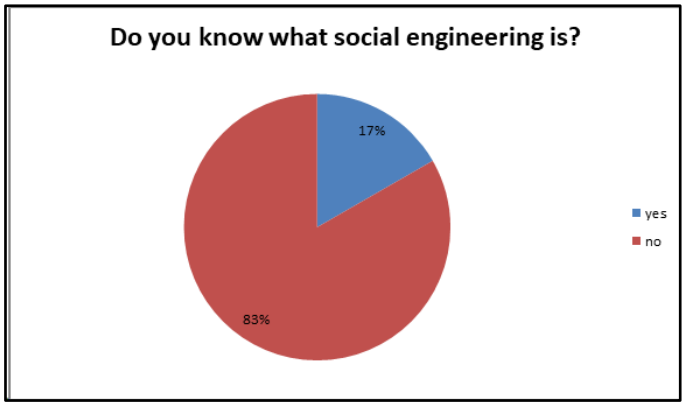


Fig. 8: A Survey in the user's knowledge about social engineering.

I. RESULT AND DISCUSSION

We received 301 responses, After analyzing the results even in a very short span of time. the participants showed so much curiosity and interest to contribute in this study and this impels to their curiosity in cybersecurity fields. We conclude that people between the ages of 15 and 40 do not have sufficient awareness of breaking off their systems with smartphones. They believe that the system alone is sufficient to protect them from penetration attacks. In addition, they believe that iOS as an OS is more powerful than Android. We also find a great awareness of the most important and easiest tools of penetration in social engineering that the OS however the level of strength it is it couldn't protect them from it.

II. CONCLUSION AND FUTURE WORK

The smartphone and cybersecurity become one of necessity and priority for people. This study starts with an introduction about iOS and Android OS Security with Social Engineering. Most of the people prefer smartphones with excellent security features to makes them feel safer. Then the methodology and experiments part have three main parts which are: The ways of access possibilities, a case study of penetration testing and a survey methodology in extent the smartphones users to know the access possibilities. Depends on the OS of the smartphone to protect the user's data from dangerous permissions our statistics and study shows different percentages based on their opinion. On the other hand, a good percentage shows that people are aware of the risks for access possibilities in applications. However, most of the participants believe that the OS security of smartphones is sufficient to protect their data. Finally, we recommend in the future more studies and awareness campaign should be designed and executed for enhancing the knowledge of the user about the coming war of cybersecurity to protect their crucial data from undesired and unauthorized access.

REFERENCES

[1] A. Arabo, "Mobile App Collusions and Its Cyber Security Implications," 2016 IEEE 3rd International Conference on Cyber Security and Cloud Computing (CSCloud), Beijing, 2016, pp. 178-183.

doi: 10.1109/CSCloud.2016.9

[2] Reed, B., "Android market share nears 50% worldwide," Available at:http://www.networkworld.com/news/2011/080111-canalys.html, Aug., 1st, 2011.

[3] Nguyen, Thi-Tra-My & Nguyen, Dong-Son & Tong, Van & Tran, Duc & Tran, Hai-Anh & Mellouk, Abdelhamid. (2018). Mining Frequent Patterns for Scalable and Accurate Malware Detection System in Android. 370-375.

[4] Launching Generic Attacks on iOS with Approved Third-Party Applications M. Jacobson et al. (Eds.): ACNS 2013, LNCS 7954, pp. 272–289, 2013.

[5] X. Li, D. Zeng, W. Mao and F. Wang. Online Communities: A SocialComputing Perspective. Intelligence and Security Informatics 2008Workshops, 2008, pp. 355-365, doi:10.1007/978-3-540-69304-8.

[6] International Journal of Advanced Research in Computer Science and Software Engineering, 3, 966-972. Low, C., Chen, Y., & Wu, M. (2011). Understanding the determinants of cloud computing adoption. Industrial Management & Data Systems, 111, 1006-1023. Massadeh, S. A., & Meslah, M. A. (2013).

[7] J. Bort, Liar, Liar, Client Server Computing^ vol. 4(5), 1997.

[8] A. Dolan, Social engineering (www.sans.org/rr/catindex.php7cat_id =51).

[9] Michael Hoeschele and Marcus Rogers, DETECTING SOCIAL ENGINEERING ,Hoeschele & Rogers ,2016.

[10] Barrera, D., Kayacik, H.G., van Oorschot, P.C., Somayaji, A.: A methodology for empirical analysis of permission-based security models and its application to android. In: Proceedings of the 17th ACM Conference on Computer and Communications Security, CCS 2010, pp. 73–84 (2010).

[11] Felt, A.P., Greenwood, K., Wagner, D.: The effectiveness of application permissions. In: Proceedings of the 2nd USENIX Conference on Web Application Development, WebApps 2011, p. 7 (2011).

[12] Felt, A.P., Chin, E., Hanna, S., Song, D., Wagner, D.: Android permissions demystified. In: Proceedings of the 18th ACM Conference on Computer and Communications Security, CCS 2011, pp. 627–638 (2011).

[13] Wei, X., Gomez, L., Neamtiu, L., Faloutsos, M.: Permission evolution in the android ecosystem. In: Proceedings of the 2012 Annual Computer Security Applications Conference, ACSAC 2012 (2012).

[14] Frank, M., Dong, B., Felt, A.P., Song, D.: Mining permission request patterns from android and facebook applications. In: Proceedings of the IEEE International Conference on Data Mining, ICDM 2012 (2012).

[15] Enck, W., Hong Tang, M., McDaniel, P.: On lightweight mobile phone application certification. In: Proceedings of the 16th ACM Conference on Computer and Communications Security, CCS 2009, pp. 235–245 (2009).

[16] Zhou, Y., Wang, Z., Zhou, W., Jiang, X.: Hey, you, get off my market: detecting malicious apps in official and alternative android markets. In: Proceedings of the19th Network and Distributed System Security Symposium, NDSS 2012 (2012).

[17] Chia, P.H., Yamamoto, Y., Asokan, N.: Is this app safe?: a large scale study on application permissions and risk signals. In: Proceedings of the 21st International Conference on World Wide Web, WWW 2012, pp. 311–320 (2012).

[18] Sarma, B.P., Li, N., Gates, C., Potharaju, R., Nita-Rotaru, C., Molloy, I.: Android Permissions: a perspective combining risks and benefits. In: Proceedings of the 17th ACM Symposium on Access Control Models and Technologies, SACMAT 2012, pp.13–22 (2012).

[19] J. Han et al. ,M. Jacobson et al. (Eds.): ACNS 2013, LNCS 7954, pp. 272–289, 2013. c Springer-Verlag Berlin Heidelberg 2013.

[20] ADVANCES IN DIGITAL FORENSICS, Michael Hoeschele and Marcus Rogers ,DETECTING SOCIAL ENGINEERING ,2016.

[21] W. Arthurs, A proactive defense to social engineering(www.sans.org /rr/catindex.php?cat_id=51).