Information Security

Student`s Name:

Institutional Affiliation:

Part I: Mobile Device Security Investigation Report

Section 1: Mobile Device Application

Title: Zoom security issues: Here's everything that's gone wrong (so far)

Author: Paul Wagenseil

Wagenseil, P. (2020). Zoom security issues: Here's everything that's gone wrong (so far). *Toms guide*, 1-3. Retrieved July 9th, 2020, from https://www.tomsguide.com/news/zoom-security-privacy-woes

Brief Summary

The article focuses on discussing the security issues associated with Zoom App, which has been accused of being unsafe, prompting hackers to target its users for personal data, which is exploited and sold on the dark web. Zoom is increasingly becoming a popular platform video-conferencing meetings, classes and social gatherings especially during this period of Covid-19 pandemic. However, according to this article, multiple security and privacy issues have been found in Zoom App which exposes the users to certain threats (Wagenseil, 2020). Zoom's privacy policies give the App a right to manipulate the user's personal data. The author also points out that Zoom's encryption policies are misleading. Furthermore, it is revealed that Zoom creates a large "attack surface", implying that the attackers would continue targeting it. Apparently, the attackers have already registered multiple zoom-related phony domains, thus facilitating the development of the zoom-themed malware.

Information Asset:

The cyber-attackers targeted users' login details (username and passwords) for the Zoom App and put up them on sale in the dark web. The login details had been stored on Zoom databases. This information was retrieved by cyber-attackers after gaining control of Zoom accounts of different users (Wagenseil, 2020).

Security Issues

From the article, it is established that Zoom App has multiple security concerns which need to be handled to prevent attackers from accessing user's personal details. The threats are both internal and external, which are instigated by human actions.

Threat

The Zoom client applications allow attackers to utilize particular developed animated GIF placed in a Zoom meeting chat to hack zoom client software on other people's mobile device. This forces the installation of malware or what is identified as "arbitrary code execution." On 18th May 2020, Zoom App experienced unexplained outage, which rendered it unavailable to multiple users in U.S. and U.K. and it lasted for some hours, before the issue was resolved. It was later revealed that cybercriminals had registered multiple new Zoom-related website addresses (Wagenseil, 2020). The attackers utilized these sites in phishing attacks, thus grabbing victims' Zoom usernames and passwords to hinder its functionality. For example, on 12th May 2020, cyber-attackers hacked the graduation ceremony at Oklahoma City University, replacing the video feed with provocative, racist language and symbols.

Vulnerability

Different vulnerabilities associated with Zoom App have been identified in this article. First, it is pointed out that Zoom creates a large "attack surface", implying that the attackers would continue targeting it. Apparently, the attackers have already registered multiple zoom-related phony domains, thus facilitating the development of the zoom-themed malware. Another flaw found on Zoom App is that The Zoom client applications allow attackers to utilize particular developed animated GIF placed in a Zoom meeting chat to hack zoom client software on other people's mobile device (Wagenseil, 2020). Zoom's encryption policy is another area

that depicts application's vulnerability. In March, Zoom acknowledged that its "end-to-end" encryption was misleading and not real since its servers constantly accessed the contents of the meetings.

Security Incident/Attack

Through these highlighted vulnerabilities, the article has described different security incidences which put the safety and reliability of the Zoom App as far its functionality and usability is concerned.

Suggested Control Measure

Some of the vulnerabilities associated with zoom App which have been identified in this article can be resolved through certain mechanisms. The control measures adopted should be preventive and detective to ensure this mobile application is reliable and stable against the activities of cybercriminals (Lemos, 2015).

Degree of Protection Provided

The vulnerabilities linked to Zoom's end-to-end encryption can be controlled through an encryption tool called Keyabase. Zoom should embark on the purchasing of Keybase and integrate it with the functionalities of Zoom App in a bid to quickly implement actual end-to-end encryption for the meetings (Wagenseil, 2020). Apparently, Keybase technology allows the user-friendly software to securely encrypt messaging and social media posts. Another vulnerability was the flaw that allowed an "arbitrary code execution". This vulnerability can be mitigated by allowing the Zoom client to append a strong of _BigPic.gif to the specified filename (Wagenseil, 2020). This mechanism would prevent cybercriminals from creating a malware file with arbitrary extension.

Limitation of this control Measure

The proposed mechanism to mitigate the vulnerability of "arbitrary code execution" can only resolve this flaw partially (Wagenseil, 2020). The contents of the _BigPic.gif filename are not limited to images only, implying that they are likely to include executable code which could be abused and utilized in the exploitation of other vulnerabilities in the Zoom App.

Section 2: Mobile Device Operating system

Title: Severe MediaTek exploit affects millions of Android devices, some may never be patched

Author: Ben Schoon

Schoon, B. (2020). Severe MediaTek exploit affects millions of Android devices, some may never be patched. *9to5google,* 1-3. Retrieved July 10th, 2020, from https://9to5google.com/2020/03/03/mediatek-security-exploit-android/

Brief Summary

The article discusses about a severe security exploit, which was linked to MediaTek chipsets, ultimately affecting millions of mobile devices using Android OS. This vulnerability originates from MediaTek's Command Queue driver, which according to the developers, it has affected multiple android devices. This security exploit is associated with a critical bug (CVE-2020-0069) which can be exploited by particular crafted file. The bug is primarily a root-access issue.

Information Asset

Passwords, username and other authentication protocols could be targeted by cybercriminals. The CVE-2020-0069 bug allows the attacker to gain root access to an android device without unlocking the bootloader by copying a script to their device, followed by code execution in a shell (Schoon, 2020). This information is stored in the secondary memory of the mobile devices and was constantly targeted by cyber-attackers.

Security Issue

Threats

The mediate bug which was associated with Mediatek chipsets was found to be posing a certain degree of security threat to the millions of mobile devices using Android OS. This security exploit was described as an elevation-of-privilege flaw (CVE-2020-0069), which is more particularly a root-access issue. This vulnerability particularly affects the super-cheap devices including Amazon's Fire tablets, Huawei and honor smartphones (Schoon, 2020). This vulnerability allows an app with root to silently install any other arbitrary application in the

background, ultimately granting them all permissions which can violate user's privacy. The outcome of a successful attack, triggered by this vulnerability can be significant. Through root access, any malicious application can give itself a permission it wants. This makes all files on Android mobile devices, even those which have been stored in private data directories of applications to be accessible, especially with a shell root (Schoon, 2020). Another threat associated with MediaTek bug is that this vulnerability is a severe security exploit which has already affected multiple devices, with some of them being unable to be patched.

Vulnerability

The security exploit addressed in this article comes in the form of a root toolkit. The vulnerability is lodged within the CPU's firmware which allows the creation of a simple script to root to an arbitrary Android device, suing the affected CPUs. This vulnerability originates from MediaTek's Command Queue driver, which according to the developers, it has affected multiple android devices.

Security Incident/Attack

This security exploit is associated with a critical bug (CVE-2020-0032) which can be exploited by particular crafted file. According to the author, the vulnerability particularly affects the super-cheap devices including Amazon's Fire tablets, Huawei and honor smartphones (Schoon, 2020). This vulnerability allows an app with root to silently install any other arbitrary application in the background, ultimately granting them all permissions which can violate user's privacy. This security exploitation can also grant access to a malicious app to inject code directly to the kernel, implying that a normal app could potentially hijack and accomplish the objective of the attacker.

Suggested control Measure

The control measure deployed to resolve this flaw should focus on eliminating an access of the malicious application through shell root. This would prevent hackers from exploiting this vulnerability and gain control of the Android mobile devices (Mayne, 2020).

Type of Control measure

The type of control measure should be both preventive and corrective. The corrective control measure would focus on patching the bug to ensure mobile devices running on Android platform are free from MediaTek vulnerability.

Degree of Protection Provided

The MediaTek has since availed the patch to debug this flaw. Through the help of Google, the MediaTek has closed the patch gap, ultimately securing the millions of devices against the potential critical security exploit of MediaTek-su bug. Android partners and OEMs developed the source code patches for issue, ultimately fixing the bug.

Limitations of this control measure:

The source code patch released to fix the MediaTek-su bug was only limited to certain Android devices. According to XDA Developers, the MediaTek chipsets are in dozens of budget, with mid-tier Android from different vendors. This implies that the patching process is likely to be elongated. In the long run, this would mean many mobile devices running on Android OS would never be patched.

Section 3: Security issue Associated with Mobile Device User Behavior

Title: Why mobile apps require access to your data and device tools?

Author: Shadma Shaikh

Shaikh, S. (2019). Why mobile apps require access to your data and device tools? *Economics Times*, 1-5. Retrieved from https://economictimes.indiatimes.com/small-biz/security-tech/technology/why-mobile-apps-require-access-to-your-dataand-device-tools/articleshow/52138161.cms

Brief Summary

According to this article, close to 40% of mobile phone users never check the app details before downloading, with about 54% rarely or never inspect the terms and conditions or permissions being requested by an App before downloading and installation (Shaikh, 2019). Such user behaviors have increased the security threats to their mobile devices, with a growing

number of hackers targeting and exploiting these user's behaviors to gain access to crucial personal data.

Information Asset

Downloading and installation of Apps without reading terms and conditions can allow malicious applications to gain access to vital data such as messages (including those containing one's bank transactions), passwords and usernames (Shaikh, 2019). Such apps can also be granted permission to access user's pictures, screenshots or messenger images which have been stored in the secondary memory of a mobile device. The apps can also reveal exact location of the user such as house number and one's email account details.

Security Issue

Threat

According to the findings published in this article, it is revealed that the proportion of the mobile phone users who download and install apps without checking the details or never reading the terms and conditions, thus granting potential oblivious app developer or malicious data miners to the. Some of the personal data that can be accessed include messages, passwords, usernames, pictures and exact location of the user (Shaikh, 2019). Allowing apps to access more data on one's phone than it is required could potentially result in security risks, ultimately exposing personal and confidential data to cybercriminals.

Vulnerability

The user's behaviors where they download and install applications without checking the app details before downloading or installing an app without reading the terms and conditions is a critical security exploit associated with mobile devices. When a mobile phone user downloads or install an App from Google's Plays Store, there is a pop-up which lists all the permission it requires (Shaikh, 2019). The scope of the permission include access to messages, phone call details and media files. In order to accomplish certain functionality, the app require access to particular in one's phone. Skipping the terms and conditions before downloading and installing the App implies the user is relinquishing the personal data to an oblivious app developer or malicious data miners.

Security Incident/Attack

Certain Apps such as chat App usually request for access to media files indicating that the user can share the pictures (media files) with the contacts. However, when such apps asks for location details becomes a security incident. A gaming app could ask to pause whenever the user gets a phone call. However, it becomes a mobile security issue if a gaming app requests to access the messages and location.

Suggested control measure:

The suggested control measure for this scenario is the users' behavioral change on how they comply with terms and conditions of various apps before downloading or installing them. Users are advised against clicking on online ads and apps without knowing the features, functionalities and details (Taylor, 2020). This will limit the activities of malicious apps which tend to access users' personal information in a given mobile device.

Type of control measure

This is the preventive control measure since it aims at cautioning the users against the behaviors of downloading and installing apps in their mobile phones without understanding the apps' details or terms and conditions.

Degree of Protection

This type of control measure would significantly protect mobile device users' against hackers who access and steal their personal data to sell in black webs.

Limitations of this control measure:

Not every user would follow the advice and adhere to the requirements of reading the details of an app and its terms and conditions before downloading or installing. This implies that some users are likely to continue being exposed to this mobile device vulnerability.

Part II: Ethics in IT Security

The concept of vulnerability has continually held a critical position in research ethics guidance, thus eliciting mixed reactions across all factions involved. Vulnerability research encompasses the approaches applied by engineering teams to identify flaws existing in software programs which could pave the way for security exploits. According to MacDonald et al. (2017), vulnerability research plays an integral role by allowing software vendors to forecast the exploit landscape. It can therefore be asserted that vulnerability research should be incorporated in the best practices and guidelines of software engineers to enhance the IT security.

There are multiple benefits associated with vulnerability research. Macdonald et al. (2017) argues that vulnerability assessment is highly essential in facilitating the process of identifying and eliminating bugs and security flaws which could otherwise be exploited by cyber-attackers. The source code should be searched and assessed to pinpoint error-prone directives (Schneier, 2008). Failure to conduct vulnerability research can give a leeway to the hackers who can bypass the authentication protocols to gain access and control of the system. A successful exploitation of the pinpointed security flaws can pave the way for the system compromise, theft of confidential data, data corruption, loss of service and data loss. This implies that security engineers should focus on anticipating how the system might fail, what can cause it to fail and what can be done prevent and protect it from the failure.

However, the entire process of vulnerability research should be done under the consent of the owner. It would be unethical for security engineers to test for the security vulnerability without permission from the authority (MacDonald et al., 2017). It is imperative for the security engineers to seek consent with the authority (organization) before conducting vulnerability research to prevent possible legal tussles. In a scenario where the permission to conduct vulnerability assessment has been granted by the relevant authority, it would be a moral obligation to provide the accurate findings (Schneier, 2008). For example, if the assessment on the source code reveals a particular set of vulnerabilities in a system, the security Engineer should report to the vendors and organization of the existence of the flaws which could foster the development of suitable solution to debug or control the vulnerability.

Organizational policy plays a crucial role in determining how the vulnerability research should be conducted. The organizational policy provides frameworks and network infrastructures which can act as checks and control in dealing with the identified flaws (Gerberding, 2018). The

organizational policy also formulates the protocols and procedures that can be applied when it comes to vulnerability assessment and reporting. The organizational policy provides the basis for the continuous assessments and remediation of the vulnerability which in turn enhances the level of data security within an organization.

Regardless of the controversies associated with ethics of research vulnerability, it sets the precedence for enhancement of system security. This approach ensures that an IT system is free of bugs and flaws which can compromise data integrity and reliability. It is however imperative to formulate organizational policies which can guarantee effective conducting of the vulnerability assessment.

References

Gerberding, K. (2018). The Difference Between Vulnerability Assessments and Vulnerability Management. *Hitachi Systems Security*, 1-4. Retrieved from https://www.hitachi-systems-security.com/blog/difference-vulnerability-assessments-vulnerability-management/

Lemos, R. (2015). 3 steps to better security in the API economy. *Tech Beacon*, 1-3. Retrieved from https://techbeacon.com/security/3-steps-better-security-api-economy

Macdonald, M. E., Racine, E., Bell, E., & Bracken-Roche, D. (2017). The concept of 'vulnerability' in research ethics: an in-depth analysis of policies and guidelines. *Health Research Policy and Systems, 15*(8), 1-5.

Schneier, B. (2008). Face-Off: Is vulnerability research ethical? Retrieved from https://searchsecurity.techtarget.com/magazineContent/Face-Off-Is-vulnerability-research-ethical

Schoon, B. (2020). Severe MediaTek exploit affects millions of Android devices, some may never be patched. *9to5google*, 1-3. Retrieved July 10th, 2020, from https://9to5google.com/2020/03/03/mediatek-security-exploit-android/

Shaikh, S. (2019). Why mobile apps require access to your data and device tools? *Economics Times*, 1-5. Retrieved from https://economictimes.indiatimes.com/small-biz/security-tech/technology/why-mobile-apps-require-access-to-your-dataand-device-tools/articleshow/52138161.cms

Wagenseil, P. (2020). Zoom security issues: Here's everything that's gone wrong (so far). *Toms guide*, 1-3. Retrieved July 9th, 2020, from https://www.tomsguide.com/news/zoom-security-privacy-woes