SANS

Analyst Program

Whitepaper

A SANS 2021 Report: Securing Cell Phones

Written by **Heather Mahalik** and **Domenica Lee Crognale**

August 2021

BlackBerry

Cisco Umbrella

Introduction

Mobile endpoints continue to surpass other devices when it comes to browsing traffic on the internet, quickly becoming the devices of choice among users who have only one digital device, likely due to their small footprint, accessibility, ease of use, and low price point. As the number of mobile devices increases, attacks on this platform steadily rise too. Many users might wonder, "Who is responsible for making sure these devices are secure?" and "How much emphasis is put on the user, the company supplying the device, the application developer, the manufacturer, or the mobile platform when it comes to security?"

This paper discusses the following topics:

- Poll results that highlight concerns related to mobile security and trending regarding personal device security practices
- Recent mobile attacks
- The various layers of protection for mobile devices
- Tips for securing mobile devices

Concerns About MobileDevice Security

In a recent poll, we asked users to identify their biggest concerns when it comes to mobile device security. More than 50% of users surveyed reported that data loss and information theft top their lists regarding the safety of their devices (see Figure 1).

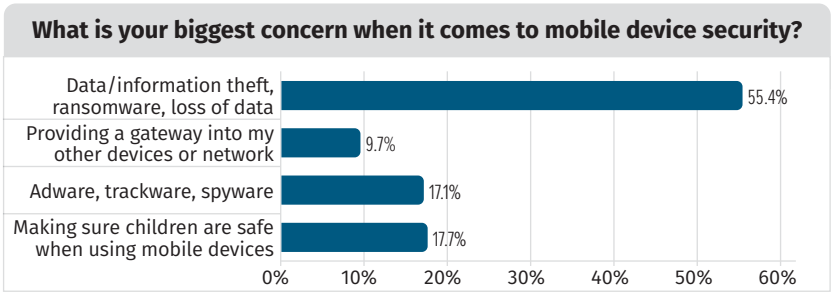

Figure 1. Mobile Device Security Concerns

Examples of Recent Attacks

Unsurprisingly, respondents identified data/information theft, ransomware, and loss of data as their biggest concerns when considering the safety of their mobile devices.

Cyberattacks have dominated the news lately, like the attack on Colonial Pipeline in April 2021,[1] JBS's US beef plants disruption in May 2021,[2] and the posting of over 700 million LinkedIn users' personal details on the dark web that stemmed from a data breach in April 2021.[3] Less publicized are some of the mobile malware variants, such as the Pegasus spyware that is once again making news after being discovered on the iOS devices of prominent officials,[4] such as the Flubot malware variant that targeted Android users in Spain.[5] Both have a seemingly simple attack vector: persuading users to click on links via specially crafted, enticing language via SMS/iMessage or another messaging platform. Attackers also increasingly use the mechanism of "overlay" applications. Designed to look like legitimate applications, these overlays contain trojans developed to steal user data to send to malicious third parties. Contrary to popular belief, mobile malware less often relies on zero-day vulnerabilities, but more commonly leverages known, reported security loopholes, hoping to target unpatched systems or applications, to infiltrate and wreak havoc on mobile devices.

Wherein lies the responsibility for securing mobile devices. Are there existing or recommended ways to help increase the security posture on mobile devices?

Fortunately, mobile device security occurs at many layers. Security often starts with the hardware and software manufacturer, but vendors, application developers, and ultimately end users also play substantial roles when it comes to security.

Users, app developers, and device administrators need to understand the risks and how to mitigate them before a mobile breach occurs.[6]

---

[1] "Hackers Breached Colonial Pipeline Using Compromised Password," www.bloomberg.com/news/articles/2021-06-04/hackers-breached-colonial-pipeline-using-compromised-password

[2] "JBS Paid $11 Million to Resolve Ransomware Attack," www.wsj.com/articles/jbs-paid-11-million-to-resolve-ransomware-attack-11623280781

[3] "Massive data leak exposes 700 million LinkedIn users' information," https://fortune.com/2021/06/30/linkedin-data-theft-700-million-users-personal-information-cybersecurity/

[4] "Private Israeli spyware used to hack cellphones of journalists, activists worldwide," www.washingtonpost.com/investigations/interactive/2021/nso-spyware-pegasus-cellphones/

[5] "Analysis of the FluBot malware variant (locally named Voicemail)," www.infigo.hr/en/analysis-of-the-flubot-malware-variant-locally-named-voicemail-n90

[6] "Combating Insider Threats with People, Processes, and AI-Based Technology," July 2021, www.sans.org/white-papers/40410/ [Registration required.]

Mobile Manufacturer/Vendor

A device's manufacturer often provides the first line of defense when it comes to mobile device security. Apple and the many manufacturers that provide hardware for the Android operating system—such as Samsung, Motorola, LG, and Google—have a multitude of product lines, with new devices in each of the lines released every year. These hardware upgrades take advantage of improvements in technology, including faster processors and increased memory with smaller/less-expensive chips, and they often include a slew of features that end users find desirable.

Usually occurring in tandem, the operating system will also undergo improvements to then leverage the changes to the existing hardware. Depending on your manufacturer, you may get notified immediately upon the availability of a new operating update for your device. For apple users who have selected to download and install automatic updates from **Settings > General > Software Updates** (as shown in Figure 2), the process occurs seamlessly. Generally, the update process occurs as the device charges or when connected to a wireless network. If you are an Android user, you manage this configuration in the

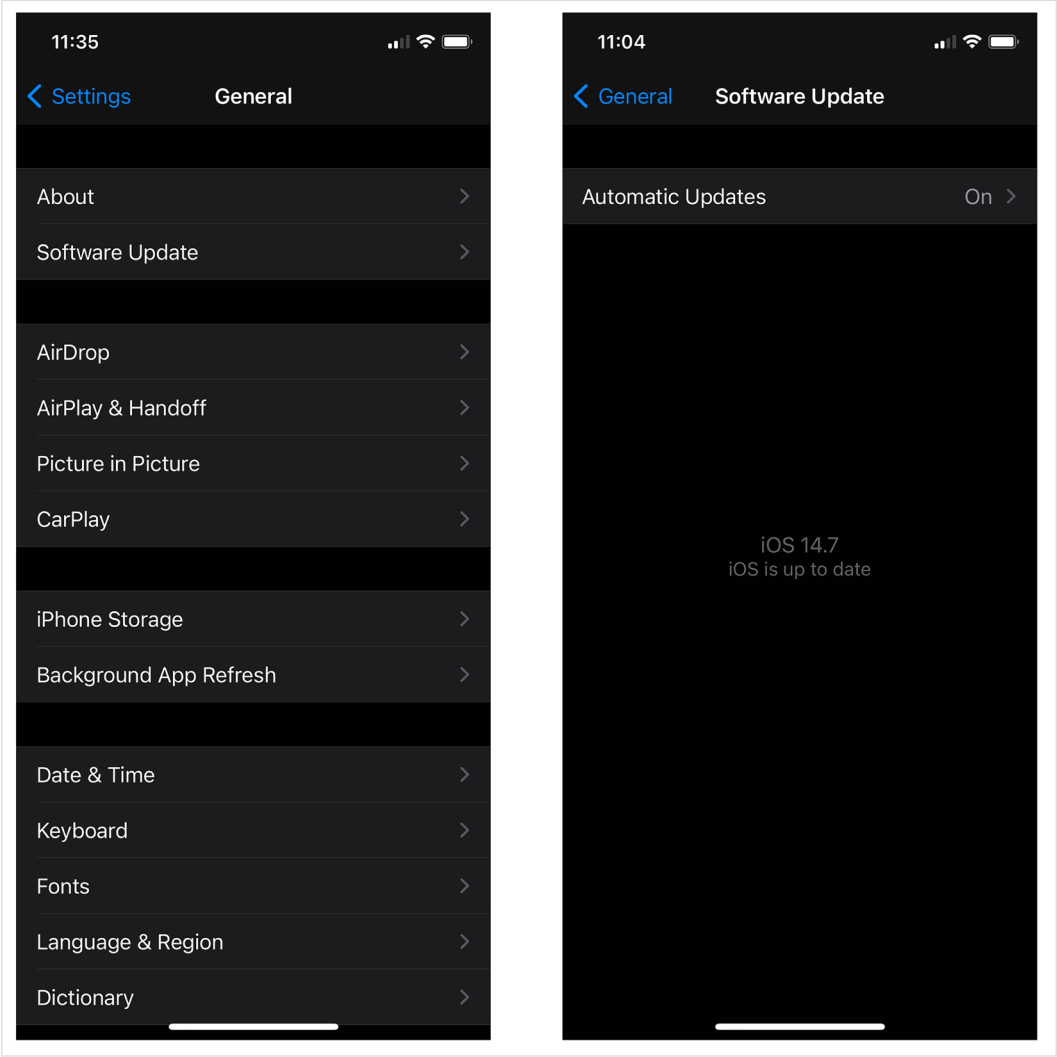

Figure 2. iOS Automatic Firmware Update Settings

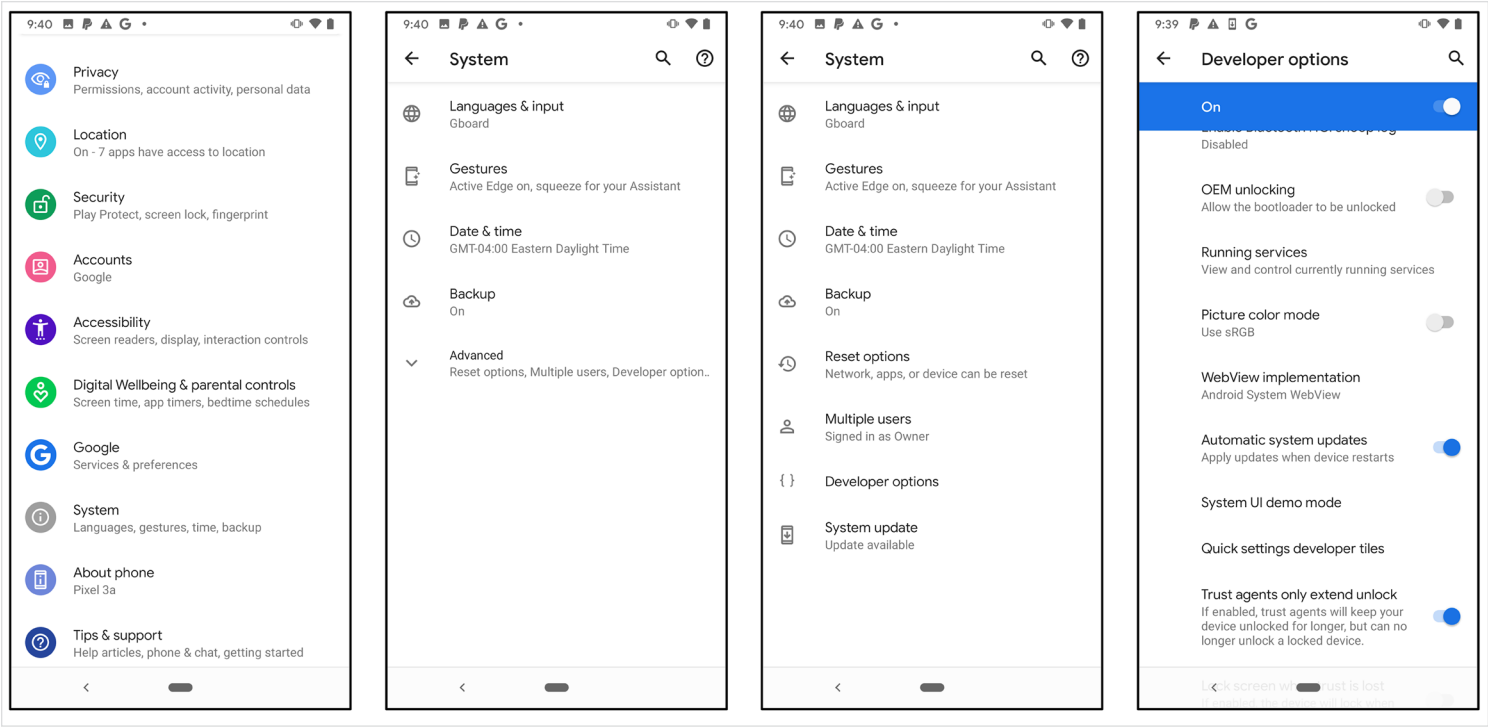

Figure 3. Android Automatic Firmware Update Settings

hidden Developer Options menu. If not already enabled, you must go to **Settings > About device/phone**, find the build number, and tap on it seven times. Once enabled, you can then confirm that you have enabled automatic system updates by going to **Settings > System > Advanced > Developer Options**, as depicted in Figure 3. Fortunately, these described settings are the default, and users need not take any additional steps.

TAKEAWAY

Knowing how to ensure your device is set to update is important. Make sure you review the settings on your mobile device.

As shown in Figure 4, most of the respondents update their mobile endpoint to the latest operating system approved for that device. Even though more than 90% claim to update, the fact that even some express uncertainty or do not update raises concern. The weakest link often offers the vector into a mobile breach.

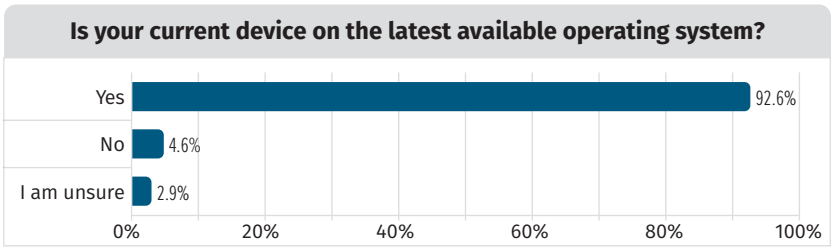

Figure 4. OS Version Status

Setting up devices to remain current is easy, but doing so does require that the device meet minimum hardware requirements. Devices up to approximately six years old can still receive updates to their firmware. In some cases, however, these older devices cannot realize the entirety of device protections offered for the latest models. For example, when Android introduced file-based encryption (FBE), a more efficient way to protect underlying data in the file system, not all devices at the time had compatible hardware to accomplish the task, even though we could update the operating system to utilize this new feature. And in the case of Apple, a hardware vulnerability known as checkm8 still exists in many devices running the newest operating system, because the vulnerability exists in the hardware and we cannot patch it with a firmware update.

Apple controls both the hardware and software for its devices, and so as long as the device is eligible (iPhone 6s and greater) and automatic updates are enabled, the device will receive the new firmware. For Android users who meet those same requirements, a chance exists that their devices won't update, because the update schedule for Android devices can be tied to the device manufacturer, the service provider, or both, prolonging or even prohibiting the update cycle. Android users may have to go directly to the manufacturer or the vendor to see if a firmware upgrade exists for their device, or they might find the latest firmware on developer forums. As shown in Figure 5, 90% of respondents own devices that can receive firmware updates. The small set of respondents who own devices that no longer update should purchase an updated mobile device.

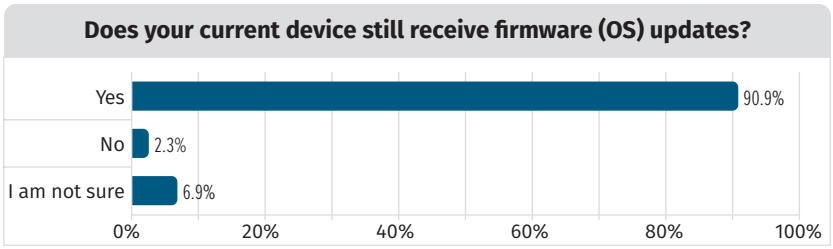

Figure 5. OS Update Status

Application Store

Another line of defense that helps to strengthen the security of mobile devices is the method in which they receive and install third-party applications. By default, Apple directs users to the only trusted repository, Apple's App Store, when they want to install new applications on their devices. Similarly, by default Android devices can install applications from trusted sources only, such as Google's Play Store. However, users can adjust this setting by ticking the box to "Allow installation of apps from unknown sources" from the **Settings > Security** menu.

One benefit of choosing to install applications from the official stores only is the vetting of not only the application but also the application developer. Apple and Google continuously monitor all applications in their stores, performing a combination of both static and dynamic analysis against the applications to ensure that they adhere to their policies. The vendors immediately remove applications that violate policies from their stores and often revoke developer licenses.

This application review extends to the devices themselves. For example, Google Play Protect, if enabled, can monitor your current device for applications that violate current policies, request undesirable or unnecessary permissions, or hide/misrepresent important information. The device receives notifications if an application appears to exhibit unwanted or suspicious behavior, so that we can disable/remove the offending application.

Sandboxing protects applications from reputable vendors from malicious intentions. *Sandboxing* refers to the process of keeping applications containerized or isolated from other applications on the device to control the permissions within each application. Both iOS and Android devices leverage application sandboxing to protect users.

Another aspect of application management that Apple handles particularly well is the minimum operating system version required to install an application. This process of limiting apps to only more current operating systems helps protect the security of the device. While most popular Android applications also adopt this process, Android developers can choose to develop their app with backward compatibility for some of the earliest operating systems.

No methodology is foolproof, and malicious applications have been known to hit both app stores before being identified and subsequently removed. However, using an app store remains safer than installing applications from outside the official stores.

TAKEAWAY

You want to ensure that you have automatic updates turned on for your device, in case the vendor pulls down or makes changes to any suspicious apps that made their way into the store.

Apple users adjust these settings in **Settings > App Store** (see Figure 6). Android users manage this by opening the Google Play Store app, tapping their profile icon, choosing **Settings > Network Preferences > Auto-update apps**, and then choosing from one of the three available options shown in Figure 7 on the next page.

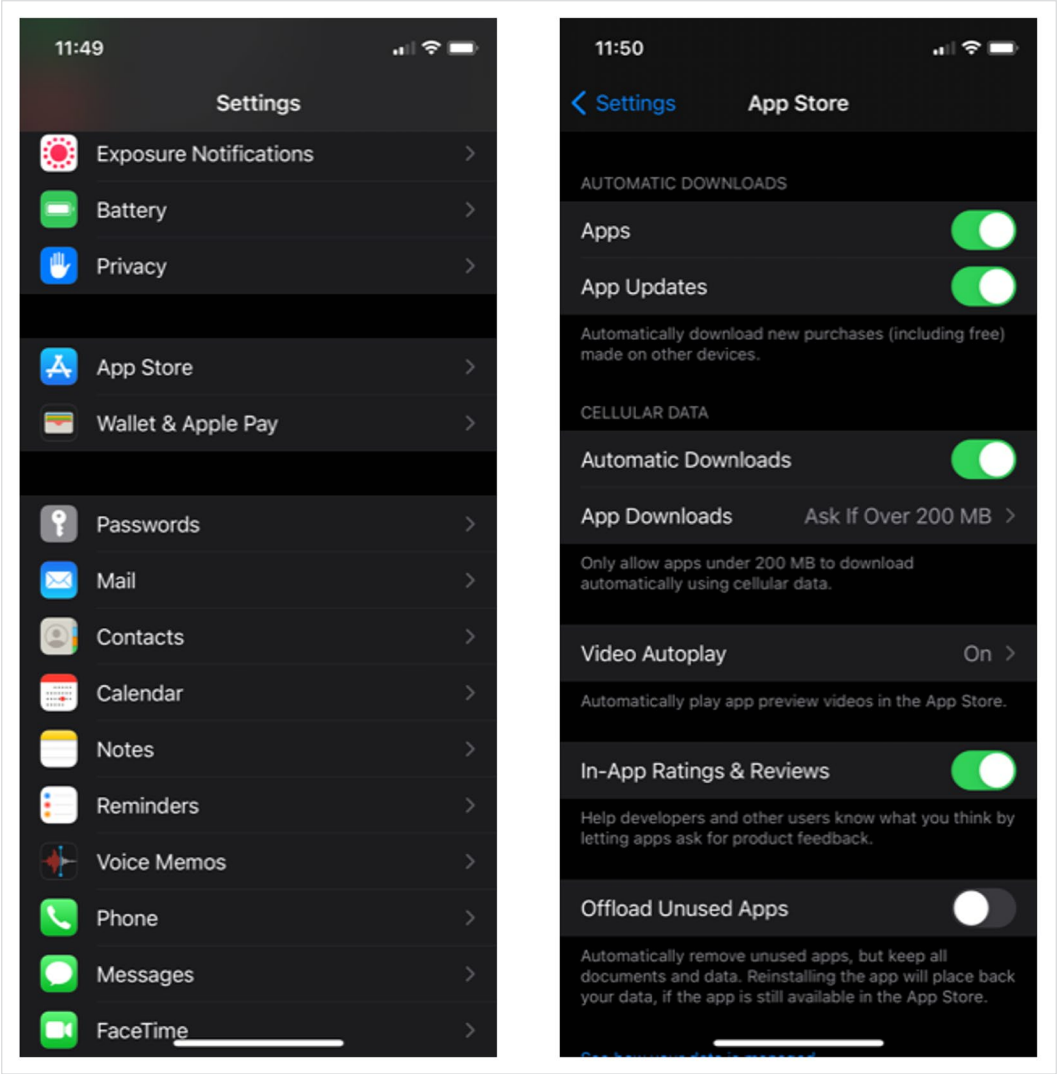

Figure 6. Apple App Store Automatic Application Updates

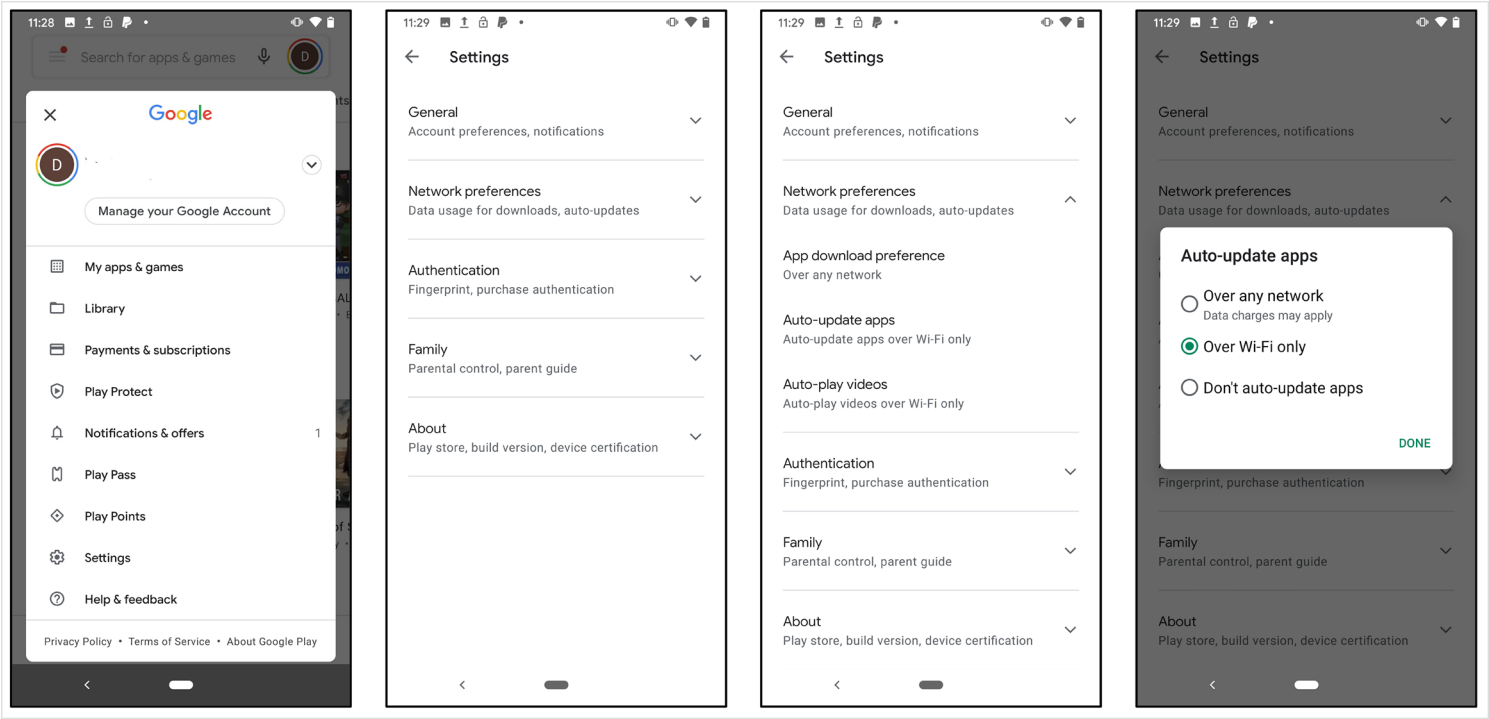

Figure 7. Play Store Automatic Application Updates

Application Development

Application development represents another area where good practices can exponentially affect the overall security posture of the mobile device. This can relate to the application itself—for example, an application that provides end-to-end communications such as Telegram or Wickr—and it can also be used to ensure the security of the user-related data associated with the application.

Secure application development is still gaining traction in the mobile application space, but developer feedback often consists of hurdles in overcoming tight deadlines, unpredictable security release blockers, and constantly evolving mobile operating systems.

Many great resources, including a free checklist from OWASP,[7] will get you started down the right path. As a developer, pay close attention to some common areas that highlight application security in the development phase. They include the highlights in Figure 8, with some also highlighted in the OWASP Top 10 Mobile Vulnerabilities.[8]

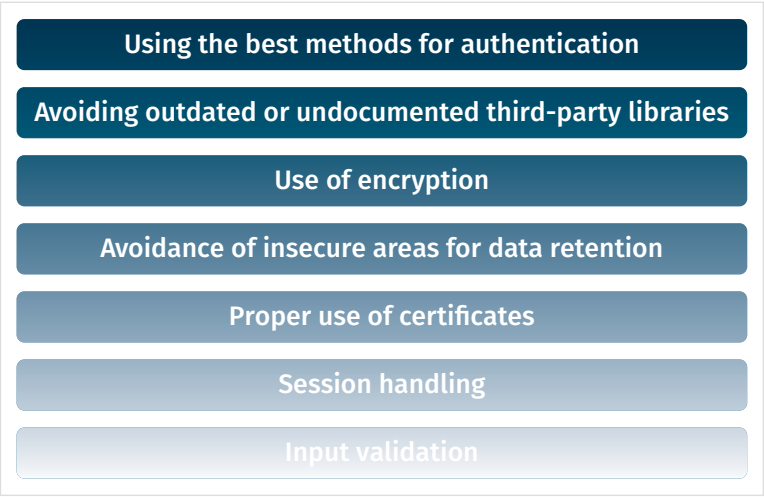

Figure 8. Application Security Highlights

When it comes to secure coding, we have many areas of concern. Fortunately, a number of vendor-supported frameworks, trainings, and certifications assist in making developers more proficient in this process.

[7] https://owasp.org/www-project-secure-coding-practices-quick-reference-guide/migrated_content

[8] https://owasp.org/www-project-mobile-top-10/

Obfuscation of source code is also becoming more common in mobile application development. This prevents an attacker from reverse engineering the application and masquerading as a similar but maliciously designed app as the original. Obfuscation makes the process of reverse engineering an application to gain insight into its processes and methods difficult or nearly impossible if you lack the original context. It essentially makes nonsense of most of the code base, including the names of the variables, classes, and methods, making the reversing process laborious and nearly impossible. Proguard represents an example of an open source tool for Android source code obfuscation. As this process becomes more common, you want to leverage free and paid resources to further protect your sensitive source code, including:

- OWASP: https://owasp.org/www-project-mobile-top-10/
- Android: https://cve.mitre.org/cgi-bin/cvekey.cgi?keyword=Android+
- Apple: https://cve.mitre.org/cgi-bin/cvekey.cgi?keyword=iOS

TAKEAWAY

Application developers must be aware of the latest exploits targeted by attackers.

Knowledge of the lowest hanging fruit for attackers will prevent developers from utilizing some of these weaknesses in their own code. This can often be validated by employing either internal or external penetration testing on the applications. This process should occur any time a major feature release is made available. Penetration testing represents a great way to vet applications before their release, but in some situations bugs or security vulnerabilities still make their way into the finished code. Immediately upon identification of such, the developer should make a patch available. While the vendor can make the patch available in the application store as soon as it's ready, the onus is now on users to ensure that their devices are set to receive application updates automatically or, at the very least, that they periodically monitor their apps for new releases. Users can configure this setting (see Figures 5 and 6).

Mobile Device Management (MDM)

Aside from the protections afforded to us by the manufacturers and application repositories, companies or individuals sometimes take data protection into their own hands and choose to utilize mobile device management (MDM) or mobile application management (MAM) for additional protection on their mobile devices. MDM provides administrator-level capabilities for mobile devices and is often used by agencies and corporations that provision and manage the well-being of devices for a group of users. MDM software, which often targets device features, has evolved and is often included in a more comprehensive utility known as unified endpoint management (UEM), which not only manages devices, features, and applications for mobile devices but also handles similar tasks for other endpoints (desktops, laptops, peripherals) in a network.

Features that MDM can control include policy enforcement, jailbreak/root detection, remote wiping of a device or application, VPN requirements, remote password locks/resets, firmware and application updates, application management, and cataloging. We make these features available by utilizing the device's APIs to interact with the phone. Whereas it might seem like MDM software has a hook into a lot of processes on the device, it is not designed to monitor your device at an application level (and therefore does not provide spy-like capabilities into your activities).

TAKEAWAY

We can greatly increase device security by utilizing MDM software to oversee some of the riskier processes attributed to activities on mobile devices.

For example, if an organization grants permission to access company email on a mobile device, it may insist (by way of policy) that launching the email application will spin up a VPN connection for that session. If you work on sensitive documents on your device and the company wants to restrict you from copying content from one application (company regulated) to a personal app for misuse, they may use MDM to restrict clipboard settings or the ability to copy data out of one application and into another. They may configure the devices to connect only to secure wireless access points to eliminate possible data leakage or theft by connecting to open Wi-Fi networks, and they can restrict certain device settings, such as the ability to create iTunes or iCloud backups on an iPhone or the ability to turn on USB debugging on an Android. This would prevent the ability to retrieve data from the phone via typical USB-data cable connections.

MDM software has become more available in the past few years, and some options meet company-owned or BYOD models. Having a way to prevent users from circumventing or disabling manufacturer or vendor settings is important, because we know that many users may knowingly or unwittingly diverge from the recommended settings on their own devices.

Along the same lines as MDM, some corporations may want to baseline their devices to get an idea of what application or configurations have changed from one point to the next. This can prove helpful in identifying a compromise or providing more stringent controls to devices in the future. Several companies specialize in baselining application code changes or device features to help identify potentially malicious activity on a device.

Users

The final line of defense for many devices is us—the end users. Although developers put many protections in place to help keep us secure, if we relax or adjust those settings, we may open up our devices to vulnerabilities. For instance, we want to keep the software updated, and we can set both applications and firmware to update automatically. As shown in Figure 9, some users choose to automatically update one or the other, but some of those polled still remain unaware of their devices' configurations.

You also want to ensure that your device is new enough to receive updates. If your device is no longer eligible to receive the latest firmware updates, it's time to get a new phone.

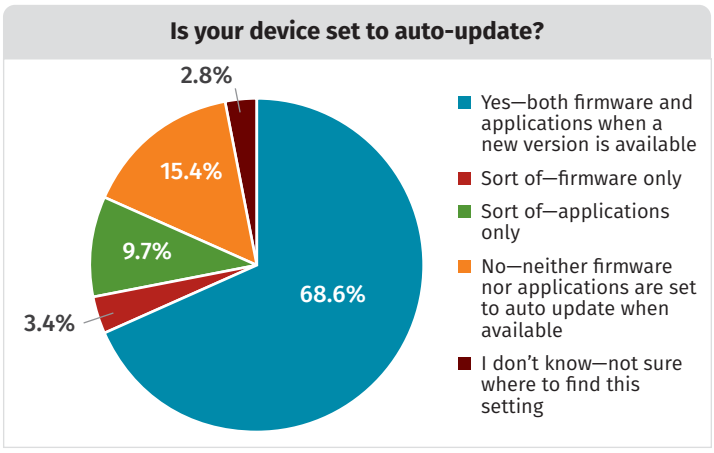

Figure 9. Application/Firmware Updates

Based on a recent survey, most users choose to upgrade their devices after a two-year period (see Figure 10), which means that they continue to take advantage of hardware improvements that can improve overall security. This also means that these devices likely run—or, at the very least, can run—the most current operating systems.

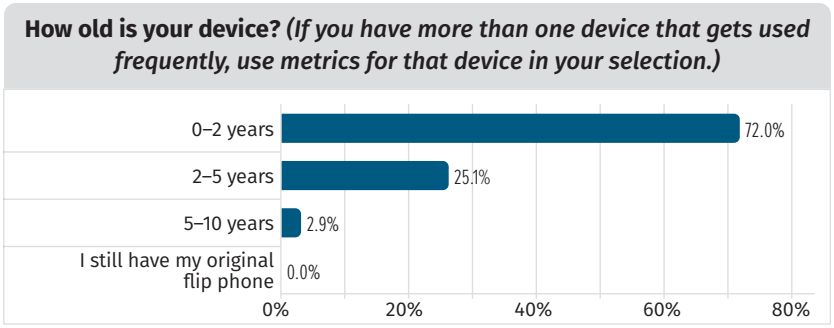

Figure 10. Device Age

You can also increase the overall security of your device by enabling a device passcode. Many different passcode types exist, but not all are created equal. If you have no password at all, make sure to set one up today. While swipe patterns and four-digit pin codes may be easy to remember, they aren't the most secure passwords because they are easier to attack. Biometrics represent a better option, but you want to ensure that when prompted (and you will be) to set up a backup passcode that you choose an alphanumeric passcode—which is more difficult to brute force—as the backup method. Although it seems unfathomable to most of us to forgo a device lock, some users still leave their devices completely open (as shown in Figure 11).

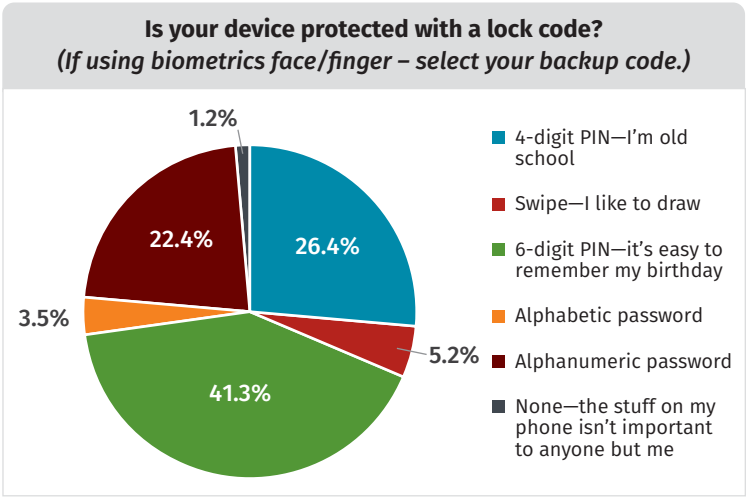

Figure 11. Device Locks

You might not know the terms *rooting* or *jailbreaking*. That's okay, because device manufacturers don't want you to do them anyway. In fact, if you jailbreak an Apple device, you automatically void the warranty. Although still technically acceptable to root an Android device, doing so now often requires unlocking the device's bootloader, which in turn may void a manufacturer warranty. This means that manufacturers know doing this entails risk, and they want your device to be as protected as possible. If you believe that you can get more out of your device by rooting or jailbreaking, you need to pay even more attention to other risky behaviors you engage in on your devices. Fortunately, as indicated by survey results in Figure 12, the majority of users do not root or jailbreak their devices.

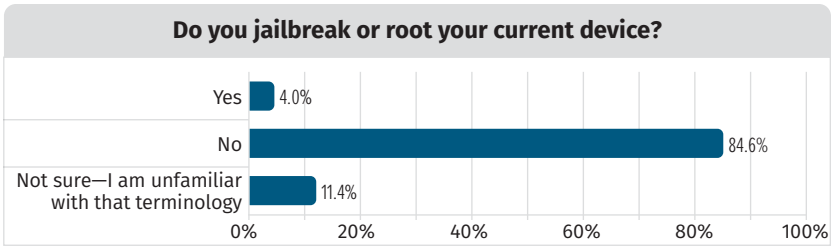

Figure 12. Rooting and Jailbreaking

TAKEAWAY

Keeping your device current is one way to make it more secure.

Where you get your applications also impacts the security of your device. Although application repositories other than the trusted app stores are available for both Android and iOS users, use caution or avoid them altogether. Based on a recent poll, some users still obtain applications outside of the trusted app stores (see Figure 13). Alternative application stores do not undergo the same level of scrutiny (if any) as that provided by the official app repositories.

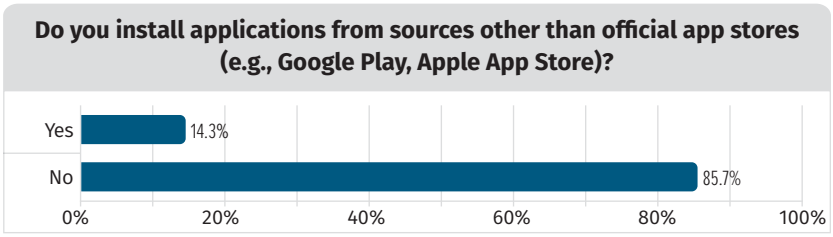

Figure 13. Official App Stores Preferred

Even if you get applications from trusted sources, you should consider a few behaviors when it comes to the applications themselves. First, trust your intuition. Sometimes a message or notification just doesn't look right. Smishing/phishing attempts still represent one of the most common ways for attackers to infiltrate mobile devices. Attackers purposely design these attacks to elicit clicks. At first glance, you might see them as legitimate, but you want to consider the source, the recipients, and the overall context of the message. The examples in Figure 14 are of SMS and email messages designed to get the user to take an action. The messages are usually peppered with "freebies" or "get rich quick claims" and redirect the user by utilizing a shortened but legitimate-sounding link. Phishing can trick users into taking action to dispute a potential unauthorized charge made to one of the user's accounts. In both cases, inspect the links as well as the sender and receiver info before taking any action.

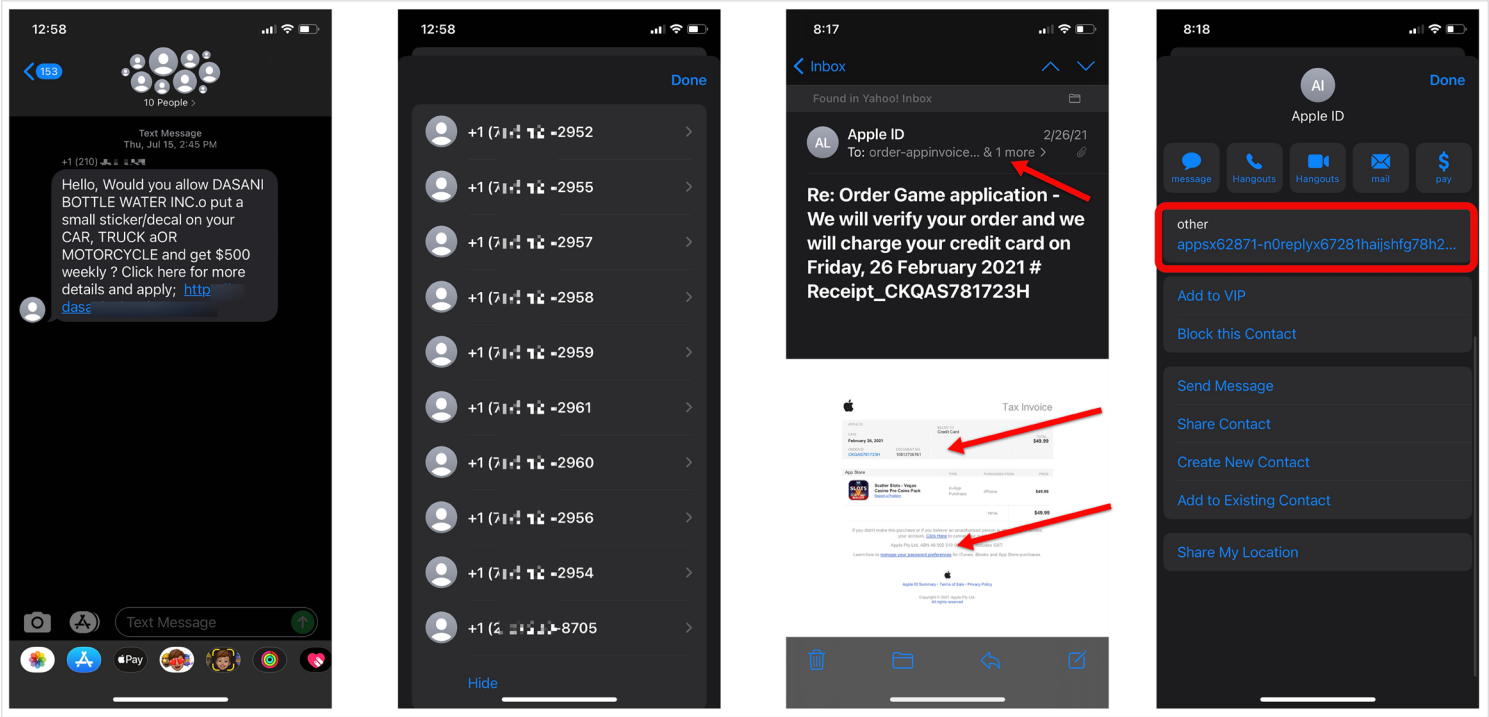

Figure 14. Smishing and Phishing Attempts

Developers create new applications every day, and often the general public has yet to fully vet them. When in app stores, stay away from trial or early access apps, as well as those that don't have a substantial user base. Although they might be benign, you can always find a handful of other applications that have positive user ratings and a large user base from which to choose. Take a look at the permissions that an app is requesting before you install it on your device. Does your fancy flashlight need permission to read and write SMS messages? If the permission doesn't seem consistent with the functionality of the application, consider that an immediate red flag. Lastly, although emerging applications try to entice you to never leave their application by adding feature after feature, avoid using an in-app browser such as those made available in many of your social media applications. Such a browser may have the same look and feel as your favorite mobile browser thanks to webkits, but it's difficult to know whether the browser is kept current. Instead, open your dedicated mobile browser (Safari, Chrome, Firefox, or others) to browse for and enter sensitive information. With mobile browsers you risk multiple vulnerabilities, but you can ensure that your own browser is always current by enabling automatic updates.

Conclusion

It takes a village to properly protect a mobile device, and everyone must work together to ensure the protection of these devices. We have many lines of defense available, and an abundance of additional software and resources enable us to further strengthen our various devices' security postures. Ultimately, end users should remember best practices they learned on other digital platforms and continue to practice good habits when accessing, storing, and utilizing sensitive information on mobile devices. Even though the old adage "If it's not broke, don't fix it" may be a part of your new post-COVID-19 mantra, this proverb doesn't always apply to mobile devices. Keep your devices current and updated and adhere to the security settings already in place to make for a safer overall experience.

About the Authors

Heather Mahalik is a SANS senior instructor and course lead for FOR585: Smartphone Forensic Analysis In-Depth. As the senior director of digital intelligence at Cellebrite, Heather focuses on forensic research and making the community smarter with regard to all aspects of digital intelligence. Her background in digital forensics and e-discovery covers smartphone, mobile device, and Mac and Windows forensics, including acquisition, analysis, advanced exploitation, vulnerability discovery, malware analysis, application reverse engineering, and manual decoding. Prior to joining Cellebrite, Heather focused on mobile device forensics in support of the federal government and served as a technical lead performing forensic examinations for high-profile cases. Heather maintains www.smarterforensics.com, where she blogs and shares presentations.

SANS Certified Instructor **Domenica Lee Crognale** is a co-author and certified instructor of SANS FOR585: Advanced Smartphone Forensics. She has 15 years of experience in cybersecurity, with a focus on digital forensics and mobile device security. Lee developed a love for mobile device forensics while working in both the law enforcement and intel communities, where she was fortunate to work on many high-profile cases. She has provided specialized training to military special forces, the US Coast Guard, and other government agencies, and has tested and validated various forensics utilities, researched artifacts associated with mobile operating systems and numerous third-party apps, and provided security assessments for many mobile applications.

Sponsors

SANS would like to thank this paper's sponsors:

BlackBerry

cisco
Cisco Umbrella