The Privacy of Snapchat

Dan Defossez
Advisor: Ming Chow

Abstract

Snapchat is a mobile app that offers users a fun way to share photos and videos temporarily with their friends and family. The service lets users show these photos and videos for 1-10 seconds, and after this time period the photos appear to be gone forever. However, there are numerous ways to bypass this time limit, including some that do not notify the sender that the image/video was saved. In addition, although Snapchat does encrypt the media before sending it to the internet, the encryption key is the same for all users, and is stored in plain text on users' devices. Furthermore, while there is no official API, Snapchat has been reverse engineered and there is documentation of an unofficial API widely available online. Because of all these things, the resulting security of the app is very weak. This paper will detail the many privacy and security concerns of the app, and discuss how this affects the users of the service.

Introduction

Snapchat is a mobile application that lets users send and receive time-limited pictures and videos (snaps). Users choose how long they want the recipient(s) to be able to view the image (from 1-10 seconds) and after that time limit the images appear to be deleted forever. In theory, this sounds like a fun way to safely share pictures and videos that one doesn't want saved, but there are numerous security flaws in Snapchat that create privacy concerns.

One of the biggest privacy leaks is that the Snapchat API was widely documented online by Gibson Security [1]. This allows anyone with programming knowledge to directly interact with the Snapchat servers, without using the official mobile app. Snapchat claims to have updated their API to detect and reject 3rd-party (unofficial) applications that use the API, but this has already been bypassed [2]. In this paper, I won't be focusing on the details of the API, but rather the privacy concerns with this application, both as a result of the exposed API and as a result of other factors.

To The Community

I chose this topic because Snapchat was a service I used daily, and I wanted to find out just how secure my snaps were. According to Snapchat, hundreds of millions of snaps are sent daily, so this topic affects many people [3]. In addition, the content of the media sent on Snapchat can be very personal, so it's important that this information is kept safe.

Vulnerabilities

1. **3rd Party Applications**

One of the biggest vulnerabilities that arises from an exposed API is the unsanctioned, 3rd party applications that developers create using the API. There are a number of plugins for Snapchat that are available for jailbroken iPhones, like Ghostprefs and Phantom. These plugins add many new features to Snapchat, such as the ability to bypass the viewing time limit, save images and videos without notifying the sender, and view a snap without notifying the sender [4][5]. They represent a huge security vulnerability for Snapchat, because they defeat the purpose of the original app.

Snapsaved.com was a website that used the Snapchat API to save snaps onto its server [6]. Users would enter their Snapchat credentials on Snapsaved.com, and all of their incoming snaps would be saved to the server, allowing the users to save them or view them again later. In October 2014, the Snapsaved.com servers were breached, and tens of thousands of snaps were retrieved [6]. The people behind Snapsaved.com promptly took the server offline and apologized for the incident [6].

This breach illustrates an important point regarding the use of 3rd party applications. By using 3rd party applications, a user is trusting the security and integrity of not only the main Snapchat service, but also that of the 3rd party application. Even if Snapchat's security were virtually impenetrable, a 3rd party application could use a user's credentials to save all their incoming snaps without their knowledge, among other things.

2. Analog Loophole

The analog loophole is a privacy concern that affects virtually all audio and video sources in the world. If the contents of a message are viewable/hearable by humans, they can therefore be viewed/heard (and saved) by a camera too. Screenshotting falls under this category, but at least the Snapchat app notifies the sender that their snap was screenshotted by the recipient. Users are not

notified, however, if the receiver users a camera to capture the snap sent to them. Though the quality of the media is lessened, this is still a leak of potentially personal information. There is no way to stop this from happening.

3. Find My Friends

In 2013, Snapchat introduced a new feature called Find my Friends. This feature allowed users to find friends on Snapchat by sending the phone numbers in their address book to Snapchat to see if any of their contacts had a Snapchat account. Initially, there was no way to opt out of this service, and Snapchat automatically uploaded users' phone numbers onto their server without consent. On New Years Eve 2013, hackers released a database of usernames and an incomplete version of their associated phone number [7].

This release showed that Snapchat wasn't taking security and privacy as seriously as they should. Less than a week prior to the database release, Snapchat even acknowledged on their blog that,

> *"Theoretically, if someone were able to upload a huge set of phone numbers, like every number in an area code, or every possible number in the U.S., they could create a database of the results and match usernames to phone numbers that way."* [8]

This is a classic instance of naïveté and neglect with regards to security and shows disregard towards their users. This line of thinking is one of the most common reasons people and organizations are

vulnerable to attacks. They think they can't be attacked because it's an involved process to attack them, or because they've never been attacked before.

Less than a week after the attack, Snapchat posted a blog post describing some updates to Find my Friends, including being able to opt-out and improving rate-limiting to prevent another attack of this nature [9]. Though it's encouraging to see Snapchat make an effort to improve security in the wake of an attack, the damage was done. Phone numbers are sensitive information, and should not be taken and uploaded without a user's consent, for this among many other reasons.

4. Hard Coded Encryption Key

Snapchat uses a single, hard-coded encryption key for all media sent through their application. Furthermore, this key (`M02cnQ51Ji97vwT4`) is stored in plaintext on Android devices, in a file `com.snapchat.android.util.AESEncrypt` [10]. Hard-coding a key is a big security flaw; if a malicious user is able to figure out the key (which is trivial in this case) for one user, they are able to decrypt media from any user.

If they got access to the Snapchat servers, they could do even more. Snapchat has stated that they store unopened snaps for 30 days before deleting them from their servers, which would be enough time for the malicious user to intervene [11]. They could view anyone's unopened snaps without notifying the users who took them. In addition, they could change or replace the snaps with a different photo or video, again without notifying anybody involved [10].

Another result of this encryption system is that snaps can be intercepted and viewed over a wireless network, provided the phones are connected to a proxy server on the network and have installed a specific, known Certificate Authority (CA) [12]. Although it's not possible to head to a coffee shop and intercept anyone's snaps on the coffee shop network (because of the need for the

proxy/CA), using a proxy server with one's own phone and computer is another way to save snaps sent to one without notifying the sender.

Action Items

There are a number of steps Snapchat should take to maximize the security and privacy of their users. They should begin by restricting access to their API. By restricting the use of the API to only the official Snapchat app, they eliminate the risk of users' information being leaked by third party applications. One way to implement this is to use OAuth 1.0a [13]. OAuth 1.0a is a secure signature based protocol, and it is widely used in the industry. Snapchat posted to their blog in November that they are taking steps to prevent unauthorized applications from using their API, but these security measures already have been bypassed [14][2]. It is encouraging to see that Snapchat is taking action towards securing their service, but they shouldn't try to "roll their own" security, as history has shown that that is very ineffective.

Snapchat should also use some system of dynamic encryption keys for encrypting media before it's sent. Hardcoding and storing the encryption key in plaintext is hardly better than no encryption at all. In a system with dynamic keys, if someone figured out the key for a single snap, they would only be able to decrypt that specific snap, rather than every snap in the entire service (as it is now).

Looking Ahead

Snapchat recently revealed a new feature in November called Snapcash. Snapcash allows users to send money over the internet using their Snapchat account [14]. Snapchat doesn't handle the actual financial/bank details, though. They've partnered with Square, a reputable credit card processing company, to deal with that [15]. However, given Snapchat's track record, I am steering clear of that feature. The company has a history of neglecting security, so it's not out of the question that something could go wrong with this feature. However, in this case there would be real financial consequences if this happened, instead of a simple invasion of privacy (as with leaking photos and phone numbers).

Conclusion

Snapchat is not as secure as it would like to make its users believe. There are many ways to save snaps without notifying the sender, which eliminate one of the main draws to the application. In addition, there was a leak of millions of phone numbers due to negligence and naïveté. Another act of neglect and carelessness is their hard-coded encryption key stored in plaintext. Snapchat needs to take security more seriously, or else there could be an even more catastrophic security breach in the future.

References

1. Gibsonsec. (2013, December 23). Snapchat - GibSec Full Disclosure (no ed.) [online]. Available: http://gibsonsec.org/snapchat/fulldisclosure/
2. Surur. (2014, November 24). 6Snap updated to beat Snapchat's 3rd party bans (no ed.) [online]. Available: http://wmpoweruser.com/6snap-updated-to-beat-snapchats-3rd-party-bans/
3. Snapchat. (2013, October 14). Who Can View My Snaps and Stories (no ed.) [online]. Available: http://blog.snapchat.com/post/64036804085/who-can-view-my-snaps-and-stories
4. (n.d.). Ghostprefs (no ed.) [online]. Available: http://moreinfo.thebigboss.org/moreinfo/depiction.php?file=ghostprefsDp
5. (n.d.). Phantom (no ed.) [online]. Available: http://apt.thebigboss.org/onepackage.php?bundleid=com.cokepokes.phantom
6. M. Isaac. (2014, October 17). A look Behind the Snapchat Photo Leak Claims (no ed.) [online]. Available: http://bits.blogs.nytimes.com/2014/10/17/a-look-behind-the-snapchat-photo-leak-claims/?_php=true&_type=blogs&ref=technology&_r=0
7. Snapchat. (2014, January 2). Find Friends Abuse (no ed.) [online]. Available: http://blog.snapchat.com/post/72013106599/find-friends-abuse
8. Snapchat. (2013, December 27). Finding Friends with Phone Numbers (no ed.) [online]. Available: http://blog.snapchat.com/post/71353347590/finding-friends-with-phone-numbers
9. Snapchat. (2014, January 9). Find Friends Improvements (no ed.) [online]. Available: http://blog.snapchat.com/post/72768002320/find-friends-improvements
10. Gibsonsec. (2013, August 27). Snapchat Security Advisory (no ed.) [online]. Available: http://gibsonsec.org/snapchat/snapchat_gibsonsec.txt
11. Snapchat. (2013, May 9). How Snaps Are Stored And Deleted (no ed.) [online]. Available: http://blog.snapchat.com/post/50060403002/how-snaps-are-stored-and-deleted
12. B. Bain. (2014, October 17). Snapception (no ed.) [online]. Available: https://github.com/thebradbain/snapception
13. Stormpath. (2013, April 17). Secure Yout REST API… The Right Way (no ed.) [online]. Available: https://stormpath.com/blog/secure-your-rest-api-right-way/
14. Snapchat. (2014, October 14). Third Party Applications and the Snapchat API. (no ed.) [online]. Available: http://blog.snapchat.com/post/99998266095/third-party-applications-and-the-snapchat-api
15. Snapchat. (2014, November 17). Introducing Snapcash (no ed.) [online]. Available: http://blog.snapchat.com/post/102895720555/introducing-snapcash