Analysis and Acquisition of WeChat under Android 6.0

Feng GAO¹, Ying ZHANG¹ and Hong GUO¹,*

¹339 Bisheng Road, Zhangjiang Hi-tech Park, Pudong New District, Shanghai, China
Email: gaofeng@stars.org.com, zhangying@stars.org.com, guohong@stars.org.com

Keywords: Smartphone, WeChat, Android 6.0.

Abstract. Smart phone allows users to exchange instant messages, as well as share videos, audio’s and images instead of relying on desktop or laptop. Consequently it increases the portability and convenience for common users. Take China for example, WeChat has become the most universal Instant Messenger application in recently several years. In the meantime, forensic technicians start to pay attention to data contained in them. Furthermore, the potential evidence could be concealed in those devices and they are generally supposed to be recovered and extracted via utilizing the proper tools. In Android 6.0 and later version, the permissions are granted in runtime instead of installation. The difficulty of acquiring data is greater than before. This paper demonstrates a method on analyzing potential evidences on Android 6.0 Devices regarding WeChat.

Introduction

Smart phones have taken over the market so well that everybody can interact, socialize, as well as share ideas and information while sitting at any corner in the world now. A brand new approach of online communication, known as social network services, emerges and develops rapidly in recent decades. We are used to communicate with people by WeChat in China. Information such as images, videos, activities and events has been exchanged via this application. Therefore, the information transmitted by WeChat is more and more important.

Android 6.0 is the sixth version of the operating system. It was officially released on October 5, 2015. As of 11 September 2017, 32.2% of devices accessing Google Play utilize Android 6.0 [1]. Along with new features and capabilities, Android 6.0 (API level 23) includes a variety of system changes and API behavior changes [2]. This release introduces a new permissions model, where users are able to manage application permissions at runtime directly. This model gives users improved visibility and control over permissions, while streamlining the installation and auto-update processes for app developers. Users can grant or revoke permissions individually for installed applications.

With a complete Android application backup, data could be acquired easily. When Android 6.0 was released, the backup permissions have been quite carefully. Acquiring data from Android system backup has been treated to be the most difficult measure. Therefore, we could not acquire the application data as usually without permission.

For the scope of this paper, we will focus on the concrete procedures regarding acquiring digital evidences of WeChat under Android 6.0. The rest of the paper is organized as follow. The purpose of evidence acquisition is described in section 2. Section 3 explains proper approaches to the evidence acquisition. Test environment and requirements will be described in section 4. Analysis and acquisition will be illustrated in section 5. Conclusion is in section 6.

Method

There are two methods for data acquisition from Android Devices, known as physical acquisition and logical acquisition.

Physical extraction involves a bit-by-bit copy of an entire physical store (e.g. flash memory). All application data of WeChat is stored in the flash memory. This technique can be used when the
device is damaged or to bypass the password protection in some cases. We can acquire data from the image without application permissions.

When android 6.0 was released, some new android devices are supposed to be encrypted in default. The Physical acquisition is meaningless to Full-Disk Encryption, while logical acquisition is the only way to obtain the data of WeChat.

As usually, root permission is required so as to obtain data of WeChat. However, limited to Android phone brand and system version, obtaining root permissions is difficult. The Android system provides backup services and APIs on devices running API Level 8 (Android 2.2) or later version[3]. Therefore, when the phone is not given root privileges, we usually take the backup mechanism to obtain the application data of WeChat and then analyze it.

**Test Environment and Requirements**

Before conducting the experiment, a forensic workstation was set up and configured, while it is required to be isolated from the laboratory’s network. The devices and tools used in the experiment are listed in table 1.

<table>
<thead>
<tr>
<th>No.</th>
<th>name</th>
<th>Versions</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>HTC</td>
<td>version 6.0.1</td>
</tr>
<tr>
<td>2</td>
<td>Android SDK</td>
<td>version 24.03</td>
</tr>
<tr>
<td>3</td>
<td>Android backup extractor</td>
<td>version v20160710</td>
</tr>
</tbody>
</table>

**Implementation**

**Analysis of the Failure in Acquisition**

However, the backup mechanism of application has changed in Android 6.0, and a complete backup can’t be obtained as usual. According to the research, each application owns its “Android Manifest.xml” file in the root directory and permissions involving data acquisition is granted in this file.

**Backup Permission**

Android provides backup and restoration to application data. When backup fails, the console will prompt such messages as “Now unlock your devices and confirm the backup operation”. “AndroidManifest.xml” can be used to enable or disable android applications from being backed up or restored. If the property “android: allow Backup” is set to false, the application backup would be prevented, otherwise it could cause the application data to be saved via adb shell as described in figure 1.
Application Reinstalled

APK is the installation package on android devices. It can be installed via adb shell with the install command on connected device. Because the lower version of the application provides backup permissions, it is used to replace the original one on the device when application backup fails.

The command "adb install -r-d XXX.apk" is usually used to replace the original application via adb shell. But sometimes the installation fails on android 6.0 devices as showed in Figure 2.

The property “versionCode” in the “AndroidManifest.xml” file is an internal version number. The number is used to describe the sequence of version, while a greater number indicates a more recent one. If the value of “versionCode” in the original configuration file is greater than the one which is ready to be replaced, the original application will not be replaced as showed in Figure 3.

Acquisition of WeChat

According to the research and analysis, the backup mechanism of application is restricted to obtain data of WeChat on Android 6.0 devices. Only when the configuration files in the application set the appropriate parameters, backup will be successful. Therefore, reinstalling the application with backup permissions and preserving the original application data on the devices becomes a solution to the problem.
**Get Original Application**

We are able to obtain the storage path of installation files for WeChat via adb shell with pm command on connected device. Then, the original application installation files can be obtained via adb shell on connected device.

**Forced Uninstallation**

In order to bypass the limitations that the version of installed application is higher than the original one, as well as install the application with backup permissions, the original application is enforced to be uninstalled. Nevertheless the data and cache directories are still kept in device after package removal as showed in Figure 4.

![Figure 4. Property “version Code”](image)

**Reinstallation**

When the original one is removed, the application with backup permissions could be installed in the devices without the any limitations. Because the data and cache directories of original application still exist, the new application could keep all the original data.

**Acquisition**

Due to the backup permissions of the new application, we could start the android application backup via adb shell easily. After the application data and cache directories are obtained, more additional application data such as photos, voices and videos from SD card are supposed to be obtained.

When the backup files are extracted, there are differences in aspect of folder structure between the data after decompression and the original one on the devices. The data from the backup folder should be sorted so as to be in accordance with the original one. By merging with additional data from SD card, we can obtain all application data as showed in Figure 5 and Figure 6.

![Figure 5. Backup the application.](image)

![Figure 6. Failure of application installation.](image)
**Reinstall Original Application**

After the acquisition is completed, the original application which was utilized in the first step is required to be reinstalled.

**Conclusion**

Few studies have committed the recovery and analysis of activities performed by applications regarding social network services on smartphones in the past decades. Therefore, the data of WeChat that we obtain from the device will provide us evidence that may be critical in an investigation. The information acquired will help us understand the subject better under scrutiny by identifying, inter alia, their contacts and affiliations, habits and interests, ideas and beliefs.

This paper explores the analysis and acquisition of the WeChat data on Android 6.0 devices. It is hoped that this paper could be able to help investigators acquire relevant application data which are needed.

**Acknowledgment**

This paper is supported by the Special Basic Research, Ministry of Science and Technology of the People's Republic of China, project number: 2016GABJC24.

**References**
