Wroba Android banking trojan targets Japan

avira.com/en/blog/the-android-banking-trojan-wroba-shifts-attack-from-south-korea-to-target-users-in-japan

November 11, 2020

The mobile banking trojan Wroba has been around since 2010. It previously targeted smartphone users, mainly in the U.S, China, South Korea, and the Russian Federation. Cybercriminals have now expanded Wroba's targets, shifting their malware campaign to Japan.

This trojan was first developed as an Android-specific mobile banking trojan, capable of stealing files related to financial transactions. Once it has infected a device, Wroba uses SMS to send messages containing malicious links to the host's stolen contact list.

In this blog, Alexandru Frigioiu, a senior threat researcher at Avira Protection Labs, analyzes a new sample of the Wroba trojan found in the wild. This variant shifts targets from South Korea to Japan. It attempts to compromise banking app users in Japan by displaying a counterfeit version of the Chrome browser with the goal of delivering the payload.

Analysis

We came across a malware sample in the wild. The sample caught our attention because it displayed attributes such as a randomized file name, package name, and activity name. These suggest a packed application and Android malware.

- **Filename:** "vAdOyCy.apk"
- **Package name:** "buhb.uabvv.szxkr"
- **Main activity name:** "zuseoje.QiActivity"

We quickly realized the app is suspicious enough to take a closer look at the app permissions.

Figure 1: *Manifest file*

Looking at the "*Chrome*" application label, we identify that the application was trying to pose as a Chrome browser. However, the discrepancy with the package name made it clear that it was not Chrome and most likely malware.

Additionally, the app was signed with a test certificate from the Android Open Source Project used in the past by several other malware families. Although this certificate is used by some legitimate developers to sign their clean Android apps, this is clearly not the case here.

Installation screen

During our analysis, we saw that the installation screen was trying to pose as a Chrome browser. It was also using the legitimate Chrome icon to make it look familiar.

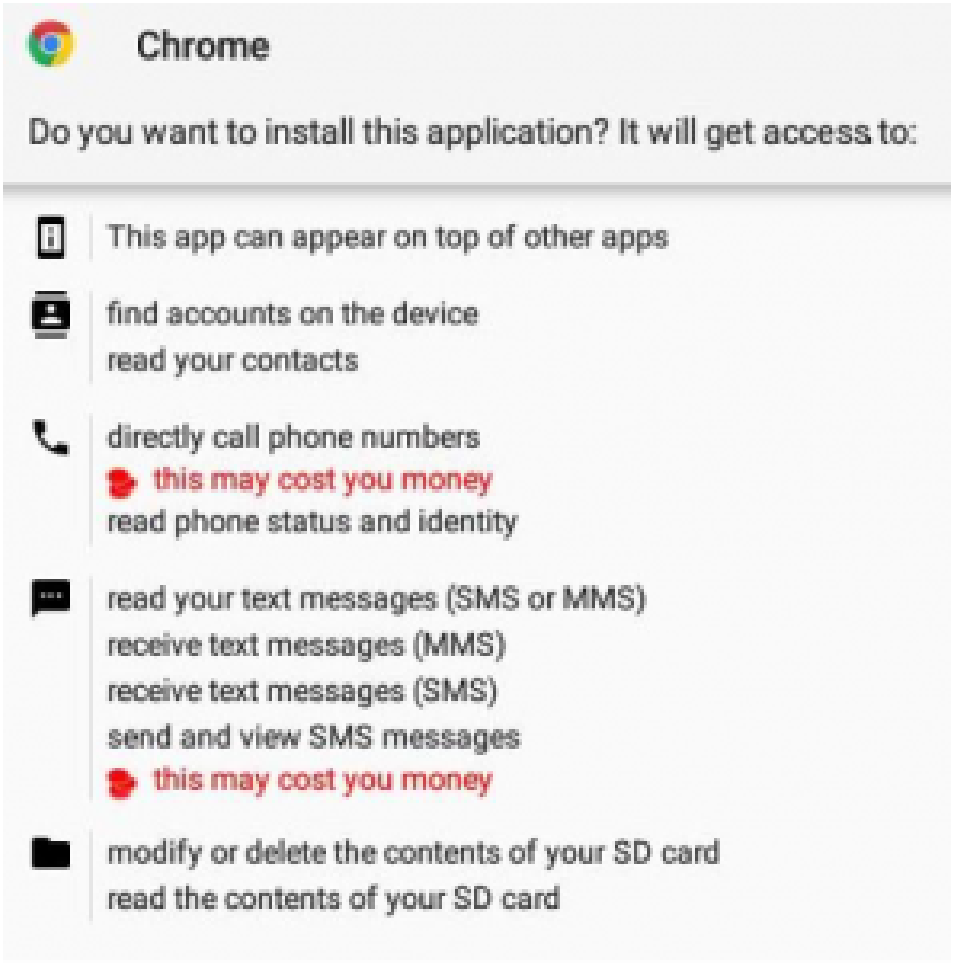


Figure 2: *Chrome application label and icon displayed on the installation screen*

After the permissions were granted, the application was launched, the icon disappeared from the launcher.

APK file structure

After we analyzed the APK file, we saw that apart from typical files found in an android app, there was a random file in the "*assets*" folder:

Figure 3: *APK file structure showing the file with a random name in assets*

This is typically used by malware to hide a second DEX or APK dropped at installation. Then loaded at runtime when the application is launched.

Looking at the contents of the file, we found that it was encrypted. We decompiled the code to find out what the application was doing with this file.

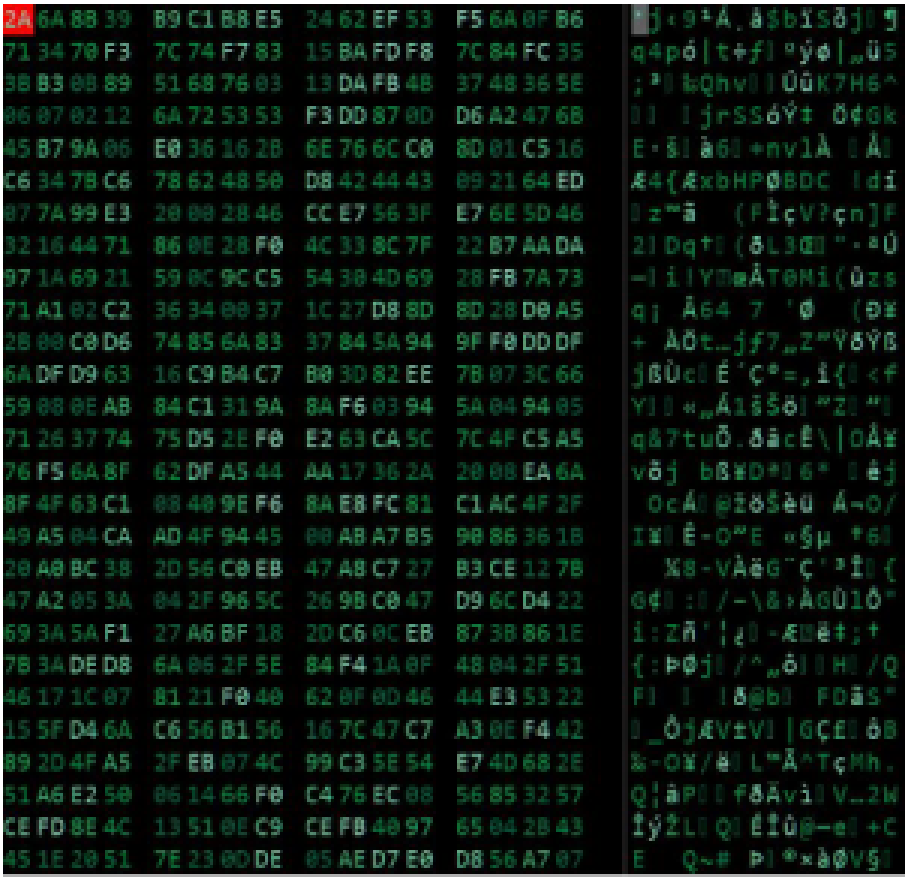


Figure 4: *Encrypted payload*

Hash of the file:
ccdc5c71c18709cea46e8dce04f985e19c054abfcb19a7ee6d875a09e3aa39b1

Main DEX file

The main DEX file is tiny (only 9kb) and we conclude its only function was to decrypt and load the randomly named file from assets.

After decompiling the code, we searched for the file name and found the following method:

Figure 5: *Decryption routine*

We saw that the file from assets is being read and processed to look like a custom decryption routine. After this, a file named “dex” is saved to “*/data/data/buhb.uabvv.szxkr/files/dex*”:

```
private File b() {
        File v1 = new File(this.getFilesDir().getAbsolutePath() + File.separator + "dex");
        if(v1.exists()) {
            v1.delete();
        }
        return v1;
    }
```

Figure 6: *Resulting file name*

We also saw that this file built the string “*dalvik.system.DexClassLoader*” out of separate strings (to avoid detection), which is then used to execute the payload:

```
private static Class<?> d() {
    StringBuffer stringBuffer = new StringBuffer();
    stringBuffer.append("dalvik".substring(0));
    stringBuffer.append(".");
    stringBuffer.append("SYS".toLowerCase());
    stringBuffer.append("Tem.".toLowerCase());
    stringBuffer.append("  Dex".substring(2));
    stringBuffer.append(ClassLoader.class.getSimpleName());
    return Class.forName((String)stringBuffer.toString());
}
```

Figure 7: *building “dalvik.system.DexClassLoader”*

Decryption routine

The function first opens the file from assets, skips the first 4 bytes, and reads the 5th byte used as an XOR key. Then reads the rest of the file in 1024 byte blocks, XOR-es it with the 5th byte, and passes the resulting data to the “this.a” method.

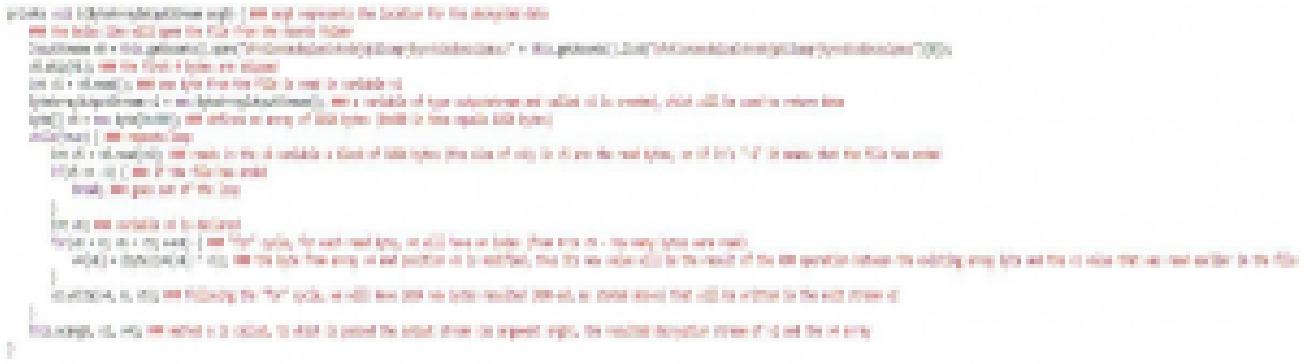


Figure 8: *Analysis of decryption routine*

The next function is decompressing the data using the deflate method:

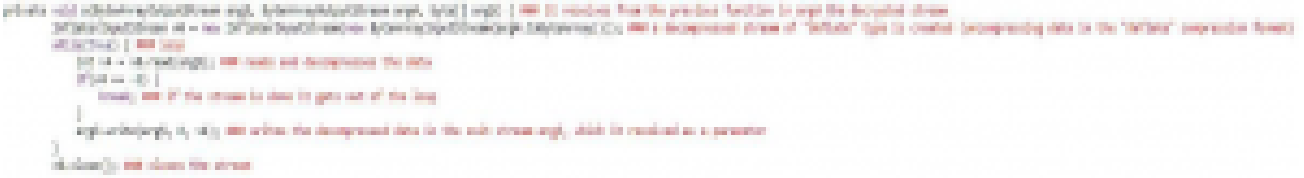
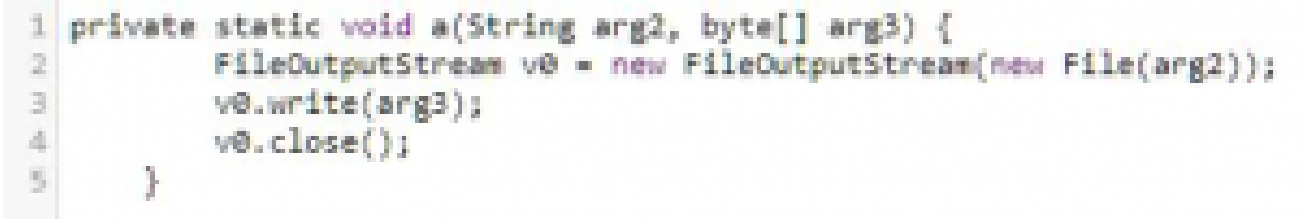


Figure 9: *The deflate method in function*

The data is then base64 decoded:

Figure 10: *Base 64 decode mechanism*

and the result is written to the file:



```
1  private static void a(String arg2, byte[] arg3) {
2          FileOutputStream v0 = new FileOutputStream(new File(arg2));
3          v0.write(arg3);
4          v0.close();
5      }
```

Figure 11: *New file is created*

Now that we know where the decrypted file is stored, we can retrieve it from the device to analyze it:

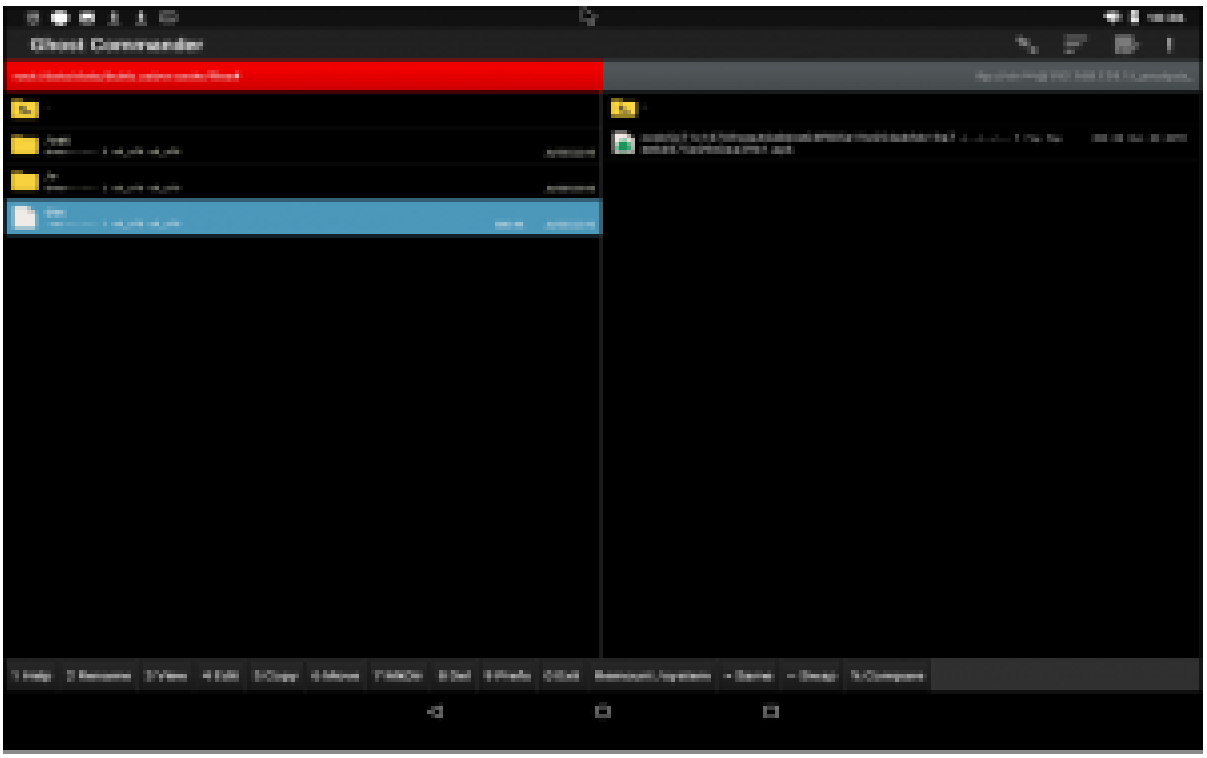


Figure 12: *Decrypted payload (upper left), named “dex”*

Payload analysis

The decrypted dex file (*0cd2b17aa21cd8de63842da21e3464df7bb2bd4a278fffbbfea6b294c3ca9e6d*) turned out to be a malware, as expected.

It was found to be a banking trojan from the Wroba family. This family has been in the wild since 2010, and its name is a concatenation of two terms, “we” and “rob”.

Malware functionality:

- tricks the user that the internet connection will be faster if the (suspicious) permission is granted
- spreads itself directly from the victim’s smartphone by sending phishing messages to the phone’s contact list
- grabs and monitors all incoming and outgoing SMS messages to bypass two-factor authentication

- monitors Android package names that are related to banking apps and overlay over them to capture credentials: *“com.wooribank.pib.smart”, “com.kbstar.kbbank”, “com.ibk.neobanking”, “com.sc.danb.scbankapp”, “com.shinhan.sbanking”, “com.hanabank.ebk.channel.android.hananbank”, “nh.smart”, “com.epost.psf.sdsi”, “com.kftc.kjbsmb”, “com.smg.spbs”.*

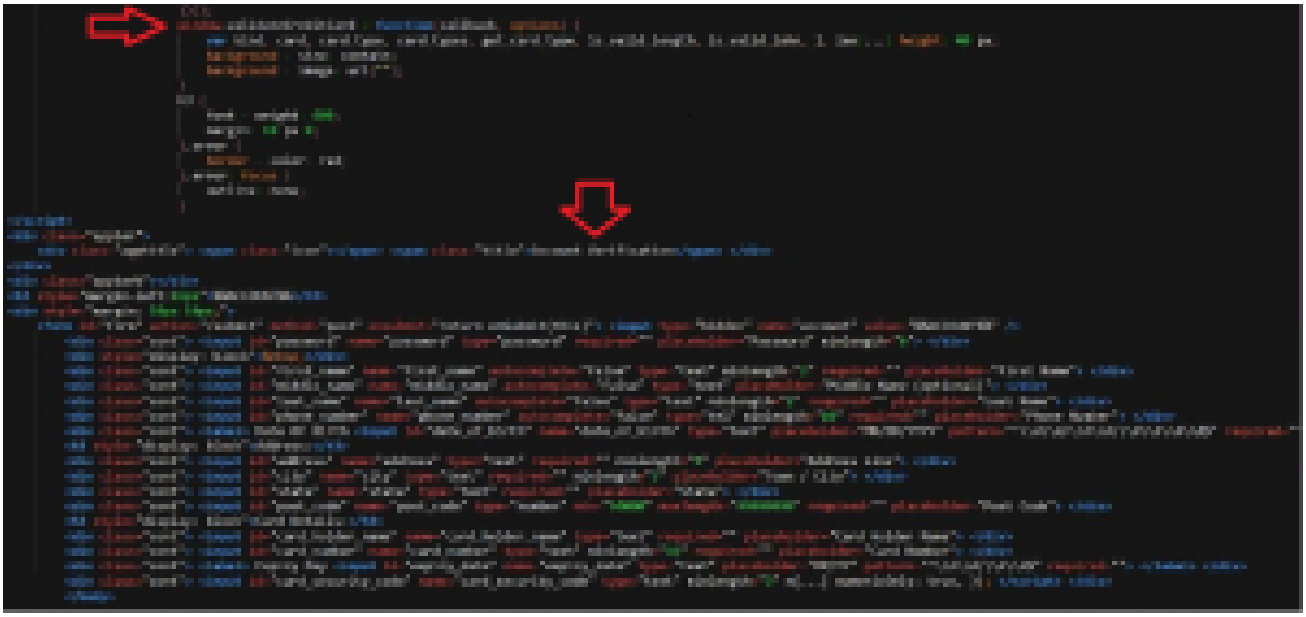


Figure 13: *HTML code of overlay*

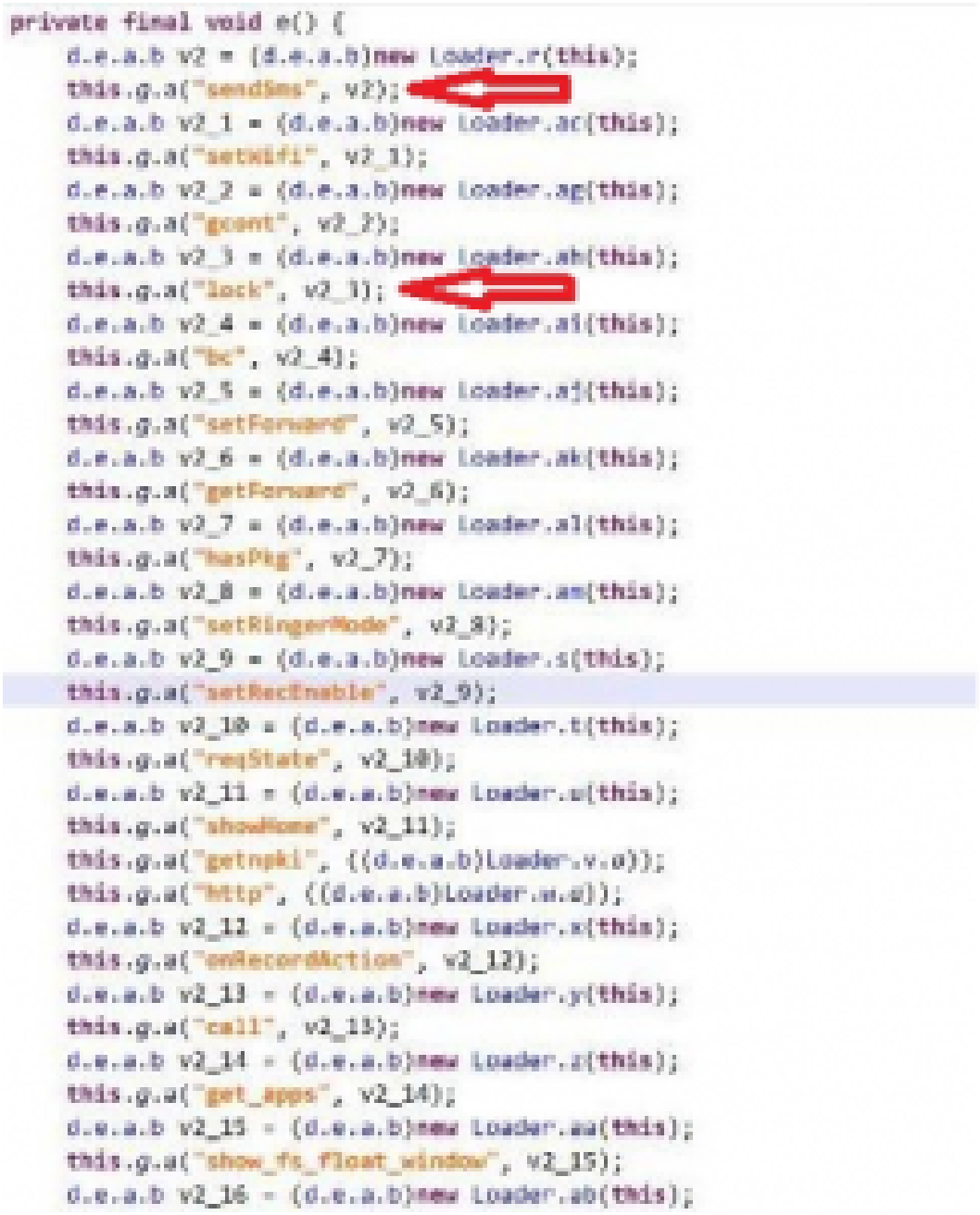


Figure 14: *Class showing some of the malware functionalities*

Figure 15: *Preparing overlay HTML content*

Figure 16: *Targeting Japanese banking apps*

Conclusion

Avira detects the original APK, dex file, decrypted payload, and other APKs from the Wroba family as Android/Wroba.

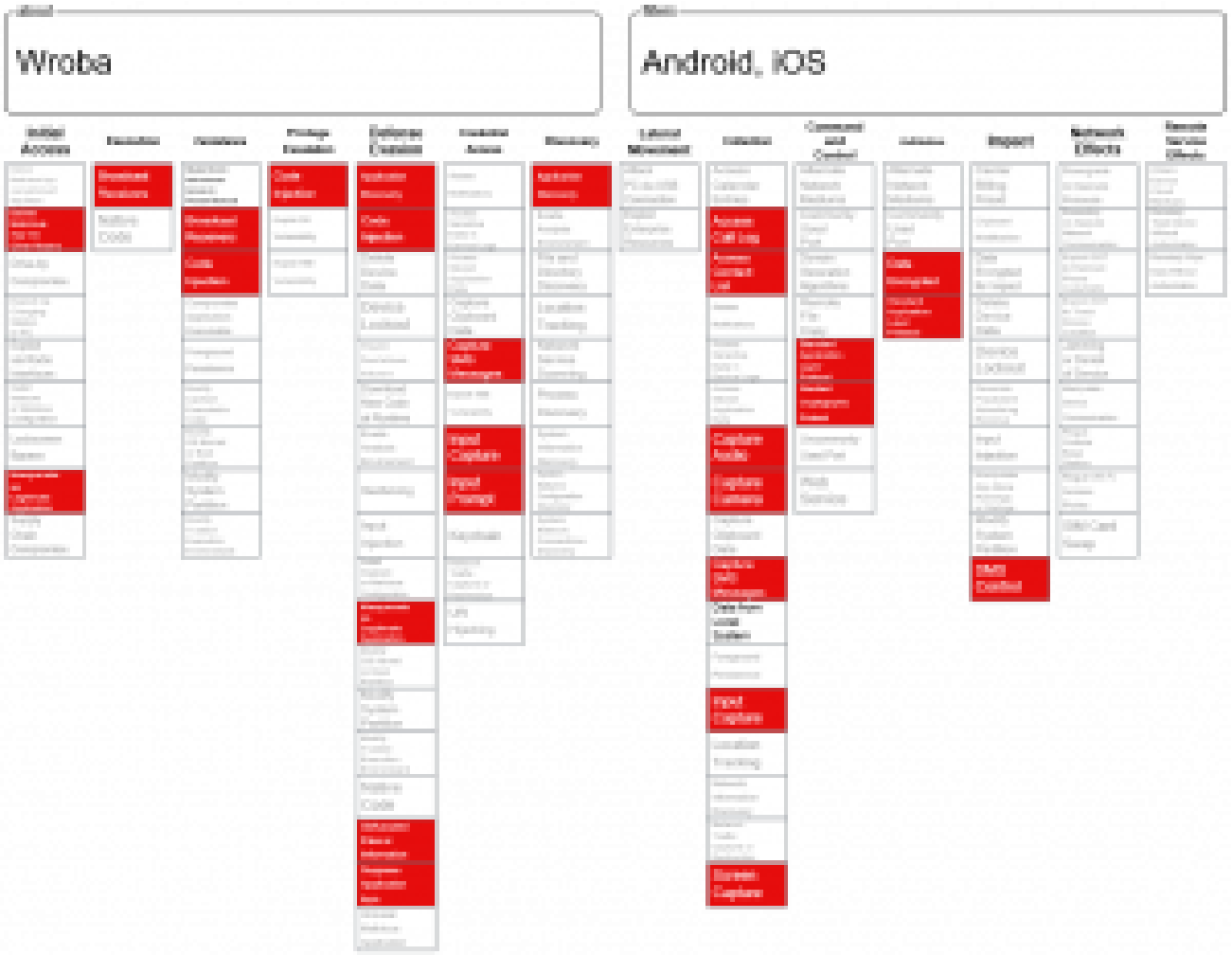
We strongly recommend Android phone users use an anti-virus package. These are available from Avira and other quality digital security providers.

Consumers should always keep the "*Unknown Sources*" option disabled in their Android device's settings to avoid installing apps from unknown sources. Android applications should only be installed from official stores such as Google Play, and even then care should be taken!

Everyone, on any device, Android, Windows, macOS and iOS should avoid clicking on unknown links received through SMS messages, emails, ads, and social media posts from untrusted sources.

Finally, mobile phone providers and network operators need to take steps to protect their customers and consider integrating anti-malware technologies such as Avira's own Anti-malware SDK for Android, and explore threat intelligence solutions that deliver mobile application reputation information.

Mitre attack matrix



IoCs

APKs containing encrypted dex files:

88193107c23011527c499e50ba6cddc8db9a4f9551e0a6e3043393aef65fd006

c4fce4222fca96055d42dde49fd85231c31978f69ebfb888e9c73252e8b7726e

96bb44eb5843debe02d460f95d427ae4122debc752dfd9477c93374065c3052b

c223f07bd07304c0b730111d22946b35b17e5b28d6bf2c2aab3dca98f1002f5d

d8feccdaed372d6cba46db534dffdea63537f7563f501add3c9b9a1087166ca6

2a378736b70a586afcab79541c6b2702ab53233ea1b6faddf8bfe2016f38f167

ccdc5c71c18709cea46e8dce04f985e19c054abfcb19a7ee6d875a09e3aa39b1

811da5d07b631e01a94899db448409d139fb9ab0eedec89dccdc170a5cdb3e0c

Encrypted dex files:

438cd827722e897248695e3c4c8b73acfdc7a6f58133bc13d7a90582ef30a76a

ed57cd7911b42a1d49b45b92776941f2be1aa2aab38f0a0c148acc2dcf9ed6fd

9f1a27161cd02e7a01d6677da4b1c6386031e33728ab5a6d2194a8006d6c047f

e169f1866e3f5acf38d384b5b1448e12bc94d6810b442e5cfd3c89686fac8064

7cc2d90ae871d2d5ca655c277aea99aca7fb401078dce6f8e94bf03065b0bf2d

0bc9de71f90d958acb36e5649ace3cfd747a53da26e71fec378e0c851a0b1d83

f8a74e5ead7b79c2c60cda43200379bfdea2d56bb9a1a44f7fedf615b60ac0f2

87921e3a6306a1fd6258be9c38a24d80b93e99ce590f1235d4e321b6e752c7cb

1ecd07bc8fd3f00825621297b92df88b65b1c5ec02a7f57b2758377bb46ac631

6e24e2663079fb8cda87bfa0dfa4254871b2bb10185caf402b277b3cfe91f926

1de1be19fb1a49f150a899fbd72aca861d12930e5d3f608574f84ba20dd53ddc

bb27ab16b50450004b1eaf1ea1cd93fc8fa16fbc670114e59a4839b254fa7580

e5ac90c1635208b6272af261d63c6d4a1ceecd5d2385564ed01b631439ceb364

3f0116ac154c75ca395ca9a4aa83b5452d4c106742188fac0f4038f9d758cb7d

068e06332271edb928696861a35b0a83b4fb34a9cac2b3ba7484c9107bacb991

77c2b592e454e6ed55605a847c43c80dd2ab662e868b41e3b8a02311ccbbbee4

f17e0353cf75e8c1ee552b811a4b60763210272b3d740f93278124499c8e63f3

79ffef47c6c306c3a3b3fbb99150a2e0006f19077adaa554b613f56df72cefdf

9a7d9c4dabface125e2d2a34e2553e0cfe0db53abd5b1d7e0671e4a197975339

254261568e55775c1a17a8342ca1f9de672cf02a80bf868d92031bd8d8dd7a90

d2c5f5dcbbc1f1236165c89c3caa46078ced0c456a950d12efb2d43095ee7504

457b7e19fd979f7cfb3e3fe07bb70efb0e6c3691c96c08bef0192786c5674237

b610207aa43a24ce7775e34711c448ebe3a3256fd14fe9569099f09d6f9be2e1

b2dd8e20f56defc742506009aca5b9541569d09f01fab31a800d40402f11fbd2

275748b01394bedb904e9b0f71577616c9d2bad921e47f926038cf8bf3121557

ff4fa45f9052fab732fcf7b010dab60a38a80da946adabc47e5fe83d9895f4b1

40098e53c730ec54206580fd65031520c8758fc40789f53dbd106a96d3b561a8

f891a0a57ef1c997b3e1bb7dc2ef6e30ff42b4ee1394c684600070b4e0a71b1b

d2f2bc2702db2ae1b4eafe93e5b16efe2b7bc67d5dc0ce12af3368301074066b

4b3a165f9449ac398df16980a6705412ede7d08debd4212041f36ce29457ab73

2feac3cb2c24571cca42a574bb3e483d30b5732707370807eab399ee9a030ce0

ee19b5db83be3a8c4a8a9c8b3589e3b54c0c7e509f948ddaa718726a16a4454f

626774a5d4eb0444b8b90e8739009586858280fc52bd7a1db796379e289d419a

62449c8c94980399799932ca59fe368ef1f072b7f1c4890071169fa34ee90272

2999de373391b37074d721cba03dfb032180293905f4e4edad754d45dce3c204

6860ad04ce6639cc6d38721cf72ce13d7d4a1c3fbb40aa56a0390618f8510a69

8a14f23121e979408ba7e505b43d934160a087ac9467e02eb54b27bbc43befd1

7dcb2c47f183ea76db8609c3faabdd11b975a098f70d7c028eee3f0aa82518fa

a44b24c3132dea3ca4372252a56c6731246bafcdc46d04379aed936f2bc64857

5eba789000774422d70a56432b6a57f687104b137e01fd7a955701b3914859b3

5bcae1e924e606f3cea5419f3cbdc5284b4ed4c95ffb2f75eaf8b323636fe85b

b65c7fdc5dcbaf82c7687a58ac6a38b173a4f41b92adf4f3d37d883ff18e6fb3

2aa6e8be19b1d189721dce63cb8a8806bb3f1a57547f0ff11e11a7b007dc5242

7f82fab70d14feb0f59b0aa06b4a3e5be8bb071a71fed4b60db9de45e4bce59e

dc1c63cb0e4d5cbbcbeb0d25848df4f1f73d05c1718ffca181857582d831735f

228cf02a878624413ca9baa5a128e0a0afee59e0dec544c9abfefed5e0860673

1f49ddc0674a5a8d75c35e8c85937621e0b286828891b0239c1f14d41bd9d0fb

1693253e9d0710af8468b35855ac66eca3b3c46dd9702a774918efa85e6eff9e

ee285fff7d7fb4871ccd7fc50d45ed890a3483c9cef34363118c86d856ab35cb

e95a46766fa619ef0fafad2ef2f131aa07ec429a9e45bd7aa0a15802b1a5d7ed

bb16d3993b0b77096ef4e5378a0e090df1c0f24ea2e2a62c0dac0d90c7a4b4c7

6efaeafa05e7cb71fc63822a39069f999e74416f57c4d3c43f28209c7b64194c

3d64202745a331c9a249e150342933d713bff6232758f9919961ceac1a6bdf36

742fe3f1d78434b25afc26c43d4f104b1c1f1c1586ef4d0d422b7143c5e56b5d

3da6e34373f0d5c1fde2a0db9ec889eb7417d54860f63bbb3c185ca602e9c4f3

fa03ce066f36007fc4a7f86aec45f2b531c70123c8cdda63925ec7749a1c1617

bebae149ef14fe760b436ec646edd7ce41c2d1b33314289f3aea1604891ec3bd

827c3d4baec64a9bb220eccdf0cf4c98fc25adcca5209a84d5f14d201163e196

27321142bfa0d955232beb03c2b2fad493f8018aff22736c8c602184b91ccc3b

f8919cf1d4edb097f41b52852d1c800833a3ae5ba8549d15e9515f00002390c6

b975858b560c7e0d08437a34b75c09417f60dd2ebb9b393530b7a3561d47d7cf

9aa90cc93be45005ded5366a604797682e41a767e4bb02c0ffb746a82664e675

6084fd74a903dc6f3d0a6d9c29da66c4df2c069556a0f355eb732706e2c5d41d

02169ae226ec5ba1f067d51b8178fcc1196ef0ff822499a27c3cd47da6b03eeb

5e6e70ad3c46a498053723d370fcb725c11e89348466493786723f227863445

fe39450ae10ad2b0469f5f5747e6b0c871137cf89501320995aa4594c40d4a4e

7c351ac692fe7b68490422b441c07ed4302acaae1f5b07c8c4ba1740a5897af8

365ec1625c59f7294c7274caf71437c3396a327e1cf50ad22bdeeb82aebe199b

381dcf5251145678684fc29c85a49e13cae461174b5a0fa3d0139a25327252cc

8c4fedc877d45d33e43f57b5c703b9840eaa0e9d1fd3042e7b62cb7af5388377

cff5e49e54a19267fce98fa6c4e9d3d81865729385c2f85b23991af4ca21c237

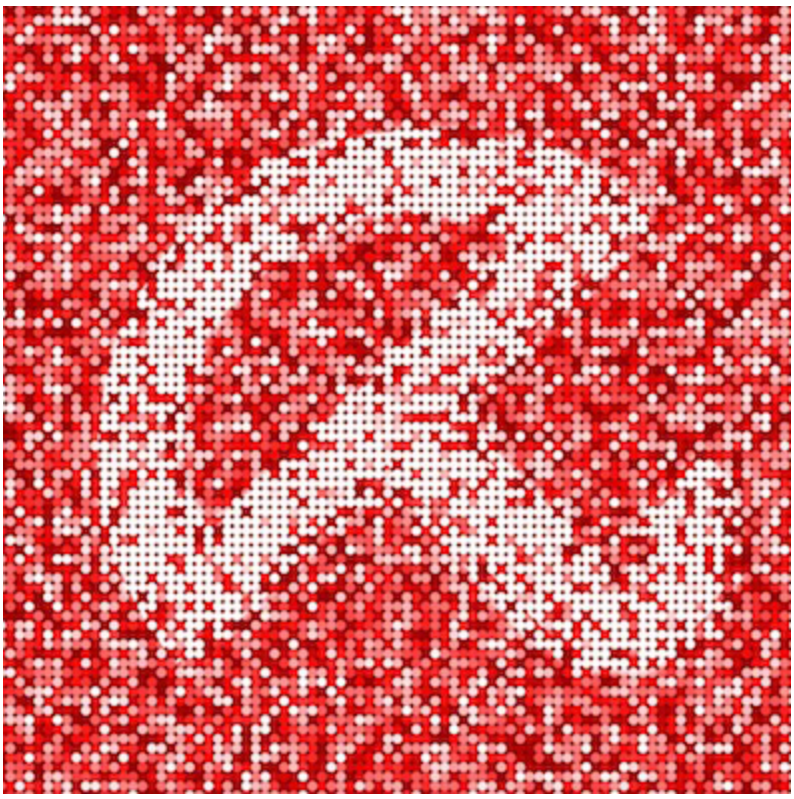
1f07092d63093c2706e5908991be3afdf478c187aeafbf84a4a3233889ae2286

c22a73489397bc63745dc92d41717e904f7c270a75821b223a856caf19d37d6d

Read our Q3 2020 malware threat report to find out the top ten Android families detected this quarter.

Like what you read?
Stay up to date with our monthly Technology Insights blog newsletter

Subscribe now



Avira Protection Labs

Protection Lab is the heart of Avira's threat detection and protection unit. The researchers at work in the Labs are some of the most qualified and skilled anti-malware researchers in the security industry. They conduct highly advance research to provide the best detection and protection to nearly a billion people world-wide.