SECTRIO

MALWARE REPORT

Silent-Installer

Date: 18/02/2021

Sanjuktasree Chatterjee

www.sectrio.com

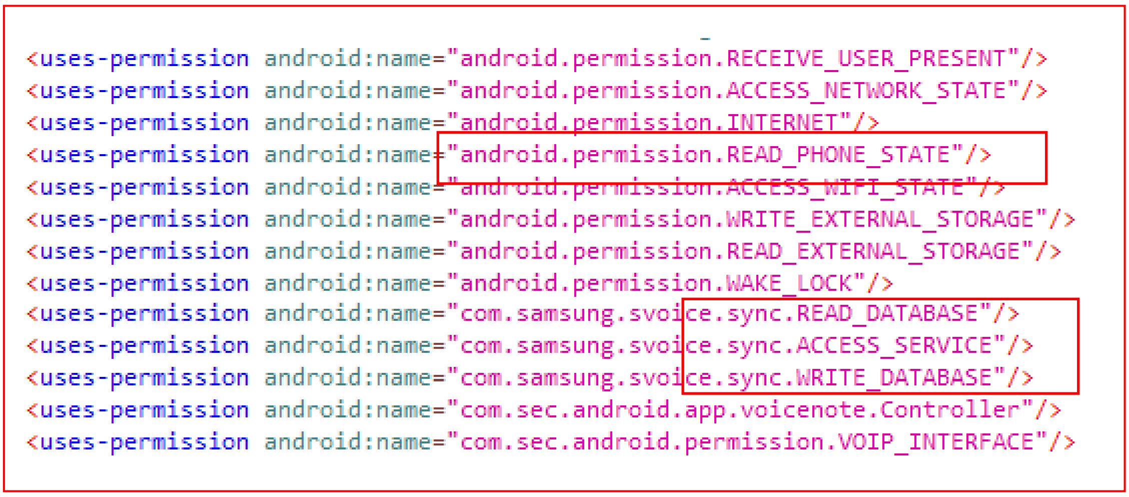
A lot of people do not think about the malware that targets their most-used devices, their smartphones. Android is a very popular OS for the most users. Cybercriminals are actively targeting this platform and the applicationsto conduct the attack with various malware families.

On 16thMarch 2021, we found a zero-day Android package file in our honeypot which is malicious in nature. This file is a variant of a business application 'mobileqqt' which is found to be a malicious installer that slows down theuser's device and collects the user's personal data.

File Hash: e74641d90c3ecb6019c4db1e2278c13c

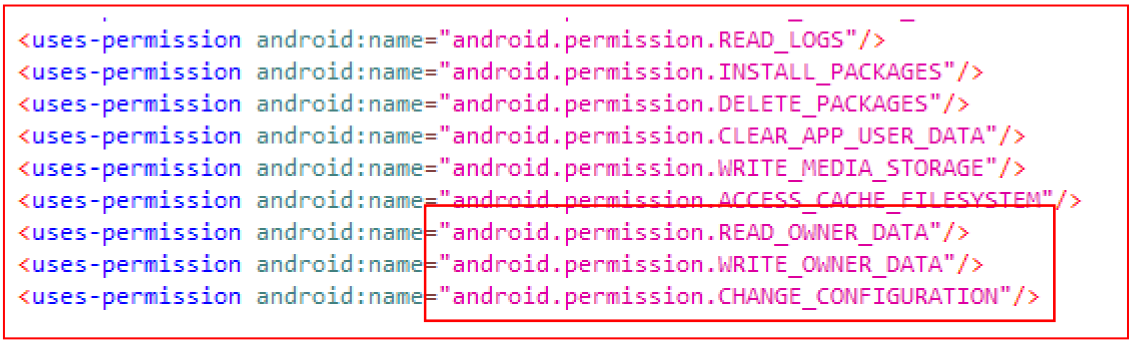
Technical Analysis:

To start the analysis of the sample, first, we need to decompile it. There are permissions for the android file that has been defined in Androidmanifest.xml file. By opening the file, we can see the sample hasa long list of permissions which are not required for that legitimate business app. In Fig: 1, we can see the suspicious permissions for this app like READ_PHONE_STATE, READ_DATABASE, access multiple services, READ_LOGS and it also can access external storage.Also,permissions to read and write owner data and change configuration (fig. 2) that leads to theinstallationofother malicious application.



```
<uses-permission android:name="android.permission.RECEIVE_USER_PRESENT"/>
<uses-permission android:name="android.permission.ACCESS_NETWORK_STATE"/>
<uses-permission android:name="android.permission.INTERNET"/>
<uses-permission android:name="android.permission.READ_PHONE_STATE"/>
<uses-permission android:name="android.permission.ACCESS_WIFI_STATE"/>
<uses-permission android:name="android.permission.WRITE_EXTERNAL_STORAGE"/>
<uses-permission android:name="android.permission.READ_EXTERNAL_STORAGE"/>
<uses-permission android:name="android.permission.WAKE_LOCK"/>
<uses-permission android:name="com.samsung.svoice.sync.READ_DATABASE"/>
<uses-permission android:name="com.samsung.svoice.sync.ACCESS_SERVICE"/>
<uses-permission android:name="com.samsung.svoice.sync.WRITE_DATABASE"/>
<uses-permission android:name="com.sec.android.app.voicenote.Controller"/>
<uses-permission android:name="com.sec.android.permission.VOIP_INTERFACE"/>
```

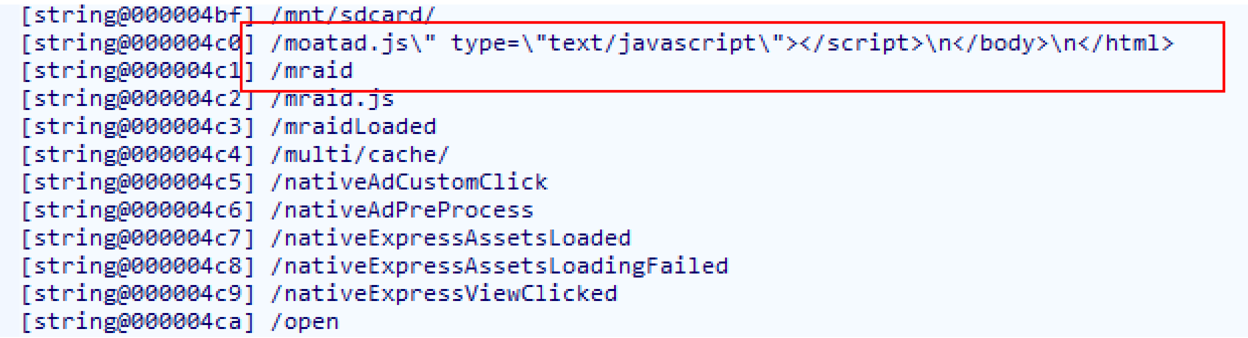
Fig: 1



```
<uses-permission android:name="android.permission.READ_LOGS"/>
<uses-permission android:name="android.permission.INSTALL_PACKAGES"/>
<uses-permission android:name="android.permission.DELETE_PACKAGES"/>
<uses-permission android:name="android.permission.CLEAR_APP_USER_DATA"/>
<uses-permission android:name="android.permission.WRITE_MEDIA_STORAGE"/>
<uses-permission android:name="android.permission.ACCESS_CACHE_FILESYSTEM"/>
<uses-permission android:name="android.permission.READ_OWNER_DATA"/>
<uses-permission android:name="android.permission.WRITE_OWNER_DATA"/>
<uses-permission android:name="android.permission.CHANGE_CONFIGURATION"/>
```

Fig: 2

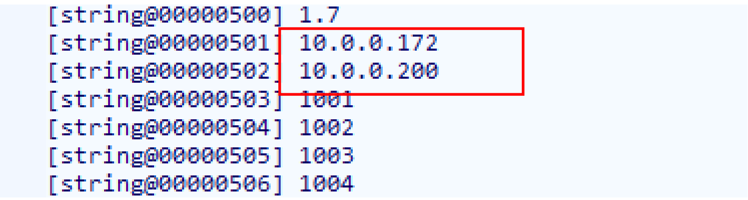
There is a malicious JavaScript code inserted in the sample code itself. When it is executed on the victim's device, it downloadsthe '`moated`' virus which isautomatically installed in the device without the knowledge of the owner.



```
[string@000004bf] /mnt/sdcard/
[string@000004c0] /moatad.js\" type=\"text/javascript\"></script>\n</body>\n</html>
[string@000004c1] /mraid
[string@000004c2] /mraid.js
[string@000004c3] /mraidLoaded
[string@000004c4] /multi/cache/
[string@000004c5] /nativeAdCustomClick
[string@000004c6] /nativeAdPreProcess
[string@000004c7] /nativeExpressAssetsLoaded
[string@000004c8] /nativeExpressAssetsLoadingFailed
[string@000004c9] /nativeExpressViewClicked
[string@000004ca] /open
```

Fig: 3

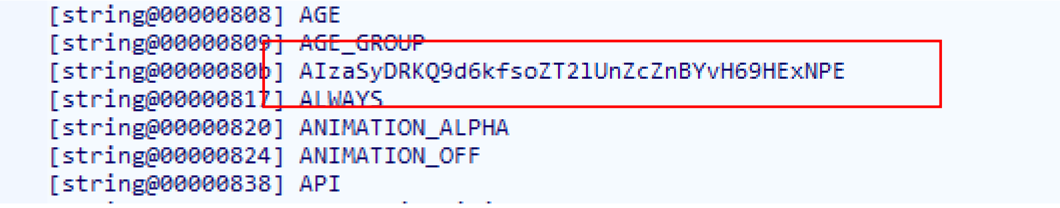
There are few private IPs that are used for downloading the virus from the site.



```
[string@00000500] 1.7
[string@00000501] 10.0.0.172
[string@00000502] 10.0.0.200
[string@00000503] 1001
[string@00000504] 1002
[string@00000505] 1003
[string@00000506] 1004
```

Fig: 4

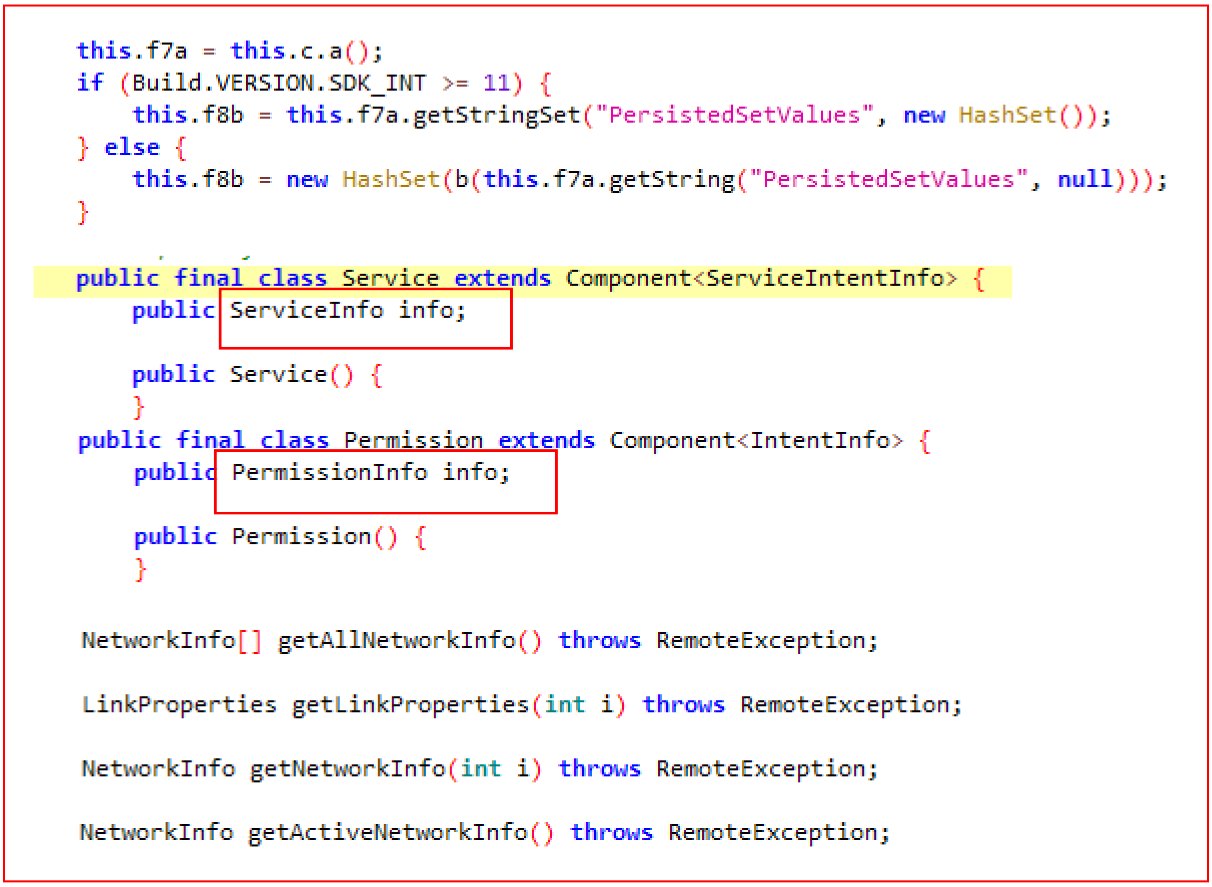
The API key '`AIzaSyDRKQ9d6kfsoZT2lUnZcZnBYvH69HExNPE`' is found openly visible in the code which is a security issue for this application. The attacker can use this information for the exploitation purpose.



```
[string@00000808] AGE
[string@00000809] AGE_GROUP
[string@0000080b] AIzaSyDRKQ9d6kfsoZT2lUnZcZnBYvH69HExNPE
[string@00000817] ALWAYS
[string@00000820] ANIMATION_ALPHA
[string@00000824] ANIMATION_OFF
[string@00000838] API
```

Fig: 5

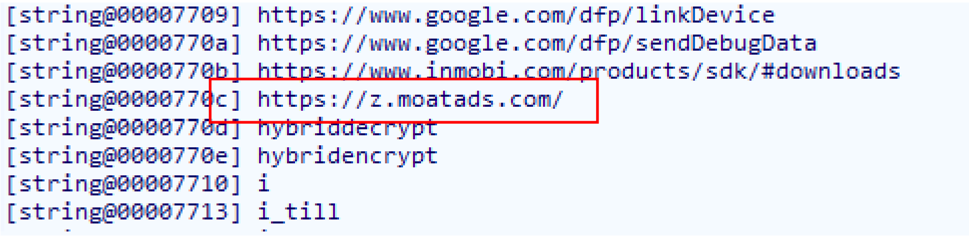
In Fig: 6, these are few API calls that are used to gather information of the owner device and data and from that information, the configuration can be changed to steal those data from the devices. The service info, persisted value, network information and the permission to the application information has been gathered from the victim's device to install malicious virus in it.



```
    this.f7a = this.c.a();
    if (Build.VERSION.SDK_INT >= 11) {
        this.f8b = this.f7a.getStringSet("PersistedSetValues", new HashSet());
    } else {
        this.f8b = new HashSet(b(this.f7a.getString("PersistedSetValues", null)));
    }

    public final class Service extends Component<ServiceIntentInfo> {
        public ServiceInfo info;

        public Service() {
        }
    public final class Permission extends Component<IntentInfo> {
        public PermissionInfo info;

        public Permission() {
        }

    NetworkInfo[] getAllNetworkInfo() throws RemoteException;

    LinkProperties getLinkProperties(int i) throws RemoteException;

    NetworkInfo getNetworkInfo(int i) throws RemoteException;

    NetworkInfo getActiveNetworkInfo() throws RemoteException;
```

Fig: 6

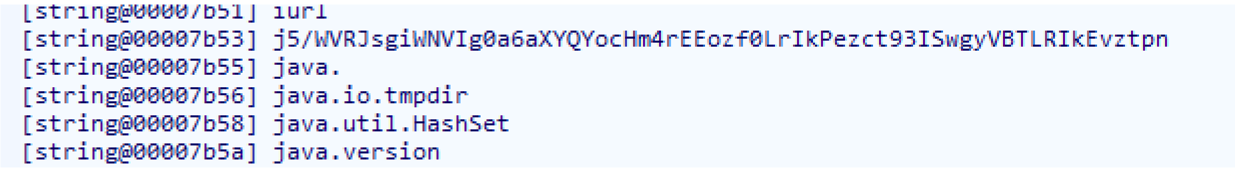
The url ‘`https://z.moatads.com/`’ redirects to the page from where the “moatads” malware has been downloaded.



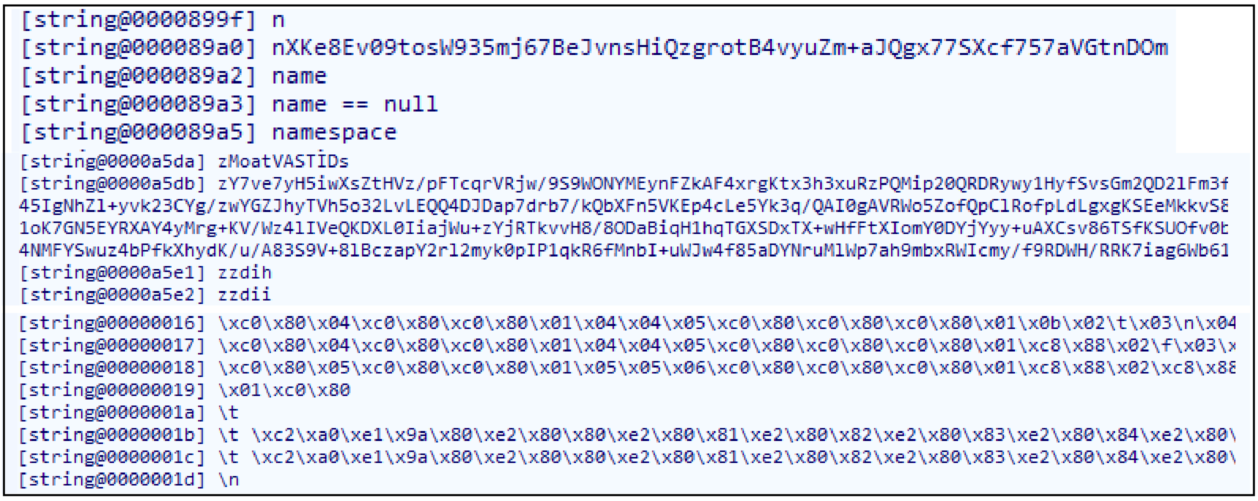
```
[string@00007709] https://www.google.com/dfp/linkDevice
[string@0000770a] https://www.google.com/dfp/sendDebugData
[string@0000770b] https://www.inmobi.com/products/sdk/#downloads
[string@0000770c] https://z.moatads.com/
[string@0000770d] hybriddecrypt
[string@0000770e] hybridencrypt
[string@00007710] i
[string@00007713] i_till
```

Fig: 7

The file contains multiple encoded patterns of strings which is compiled in the JavaClass file. After decoding, it shows the JavaScript information that suggests the malicious embedded file.



```
[string@00007b51] iurl
[string@00007b53] j5/WVRJsgiWNVIg0a6aXYQYocHm4rEEozf0LrIkPezct93ISwgyVBTLRIkEvztpn
[string@00007b55] java.
[string@00007b56] java.io.tmpdir
[string@00007b58] java.util.HashSet
[string@00007b5a] java.version
```



```
[string@0000899f] n
[string@000089a0] nXKe8Ev09tosW935mj67BeJvnsHiQzgrotB4vyuZm+aJQgx77SXcf757aVGtnDOm
[string@000089a2] name
[string@000089a3] name == null
[string@000089a5] namespace
[string@0000a5da] zMoatVASTİDs
[string@0000a5db] zY7ve7yH5iwXsZtHVz/pFTcqrVRjw/9S9WONYMEynFZkAF4xrgKtx3h3xuRzPQMip20QRDRywy1HyfSvsGm2QD21Fm3f
45IgNhZl+yvk23CYg/zwYGZJhyTVh5o32LvLEQQ4DJDap7drb7/kQbXFn5VKEp4cLe5Yk3q/QAI0gAVRWo5ZofQpClRofpLdLgxgKSEeMkkvS8
1oK7GN5EYRXAY4yMrg+KV/Wz41IVeQKDXL0IiajWu+zYjRTkvvH8/8ODaBiqH1hqTGXSDxTX+wHfFtXIomY0DYjYyy+uAXCsv86TSfKSUOfv0b
4NMFYSwuz4bPfkXhydK/u/A83S9V+8lBczapY2rl2myk0pIP1qkR6fMnbI+uWJw4f85aDYNruMlWp7ah9mbxRWIcmy/f9RDWH/RRK7iag6Wb61
[string@0000a5e1] zzdih
[string@0000a5e2] zzdii
[string@00000016] \xc0\x80\x04\xc0\x80\xc0\x80\x01\x04\x04\x05\xc0\x80\xc0\x80\xc0\x80\x01\x0b\x02\t\x03\n\x04
[string@00000017] \xc0\x80\x04\xc0\x80\xc0\x80\x01\x04\x04\x05\xc0\x80\xc0\x80\xc0\x80\x01\xc8\x88\x02\f\x03\x
[string@00000018] \xc0\x80\x05\xc0\x80\xc0\x80\x01\x05\x05\x06\xc0\x80\xc0\x80\xc0\x80\x01\xc8\x88\x02\xc8\x8
[string@00000019] \x01\xc0\x80
[string@0000001a] \t
[string@0000001b] \t \xc2\xa0\xe1\x9a\x80\xe2\x80\x80\xe2\x80\x81\xe2\x80\x82\xe2\x80\x83\xe2\x80\x84\xe2\x80\
[string@0000001c] \t \xc2\xa0\xe1\x9a\x80\xe2\x80\x80\xe2\x80\x81\xe2\x80\x82\xe2\x80\x83\xe2\x80\x84\xe2\x80\
[string@0000001d] \n
```

Fig: 8

IOC's:

10.0.0172
10.0.0.200
https://z.moatads.com/

MITRE Techniques:

Install Insecure or Malicious Configuration(T1478)
Masquerade as Legitimate Application(T1444)
Access Sensitive Data in Device Logs(T1413)
Deliver Malicious App via Other Means (T1476)

Sectrio Protection

Sectrio detects this malware as '`SS_Gen_Silent_Installer_A`**'.**

Our Honeypot Network

This report has been prepared from the threat intelligence gathered by our honeypot network. This honeypot network is today operational in 72 cities across the world. These cities have at least one of the following attributes:

- **Are landing centers for submarine cables**
- **Are internet traffic hotspots**
- **House multiple IoT projects with a high number of connected endpoints**
- **House multiple connected critical infrastructure projects**
- **Have academic and research centers focusing on IoT**
- **Have the potential to host multiple IoT projects across domains in the future**

Over 12 million attacks a day is being registered across this network of individual honeypots. These attacks are studied, analyzed, categorized, and marked according to a threat rank index, a priority assessment framework that we have developed within Sectrio. The honeypot network includes over 4000 physical and virtual devices covering over 400 device architectures and varied connectivity mediums globally. These devices are grouped based on the sectors they belong to for purposes of understanding sectoral attacks. Thus, a layered flow of threat intelligence is made possible.