

# Lookout Discovers PlainGnome and Bonespy Uzbek Android spyware

By Lookout

Published: 2024-12-11 · Archived: 2026-04-05 19:51:28 UTC

Researchers at the [Lookout Threat Lab](#) have discovered two Android surveillance families dubbed BoneSpy and PlainGnome. They are both attributed to a threat actor associated with Uzbekistan's intelligence service, the State Security Service. This threat actor, Sandcat, was [initially discovered in 2019](#). While our analysis does not point directly to Sandcat, previous reporting on similar activity from Amnesty International in 2019 showed that the State Security Service is involved in domestic surveillance as well as espionage against neighboring countries.

BoneSpy and PlainGnome appear to target Russian-speaking victims across Central Asia in countries including primarily Uzbekistan, as well as in Kazakhstan, Tajikistan, and Kyrgyzstan. Lookout researchers uncovered an indication of possible enterprise targeting using the BoneSpy family in early 2022.

## App Families Analysis

Lookout has tracked BoneSpy since December 2021 and discovered PlainGnome in January 2024. BoneSpy is derived from the Russian open-source [DroidWatcher](#), a surveillance app developed between 2013 and 2014. Conversely, PlainGnome is not based on open-source code, but shares similar theming and C2 server properties with BoneSpy. PlainGnome is also a two-stage deployment while BoneSpy is a self-contained single app. Each of these have broad surveillance capabilities including:

- Attempting to gain root access to the device
- Anti-analysis checks
- Location tracking
- Getting information about the device
- Getting sensitive user data such as:
  - some text
  - SMS messages
  - ambient audio and call recordings
  - notifications
  - browser history
  - contacts
  - call logs
  - photos from the camera
  - screenshots
  - cell service provider information

## Sandcat Used DroidWatcher

A 2019 report from [Amnesty International](#) exposed Uzbekistan government use of an extended version of DroidWatcher to target dissidents domestically. The BoneSpy family later used by Sandcat is a newer version of this spyware.

## Detailed Analysis: Custom DroidWatcher

Sandcat's earlier custom DroidWatcher sample targets Android API 16 (version 4.1), indicating it was targeted against older Android phones. This modified DroidWatcher sample collects the following categories of information:

- Screenshots, using a "Shoter" class similar to BoneSpy
- Device location via GPS and cell (CID, MCC, MNC)
- Device battery level
- SMS messages
- Calendar information including the owning account
- Photos stored on the device
- Photos from the device cameras
- Ambient audio and call recording
- Browser history
- Clipboard contents

The modified DroidWatcher sample also attempted to collect chat messages and other information from various social media apps, via on-device databases, such as:

- WhatsApp
- Viber
- Facebook
- VKontakte
- IMOIM
- Odnoklassniki
- Mail.ru
- Telegram
- Telegram Plus

Exfiltration and command and control occur via a publicly available XMPP client called [Rooster](#).

## Detailed Analysis: BoneSpy

The BoneSpy family showed evidence of continuous development between roughly January and October 2022, after which samples began using consistent lure theming and code structure. Earlier samples from between January and September 2022 used a variety of trojanized apps such as battery charge monitoring apps, photo-gallery apps, a fake Samsung Knox app, and trojanized Telegram apps. Later, Sandcat largely shifted to using trojanized, fully functional Telegram samples titled as "Beta" versions.

Early samples featured a high degree of feature experimentation, with core capabilities to collect the call log, file system, contact list, SMS messages, and emails, while other samples included audio recording functionality. Two early samples used RTMP (Real-Time Messaging Protocol), an open-source streaming protocol, for command and control. Still others checked for root access by attempting to write the string “ZZZ” to a file path only accessible with elevated privileges.

```
public static boolean isPhoneRooted() {
    try {
        java.lang.Process process0 = Runtime.getRuntime().exec("su");
        DataOutputStream dataOutputStream0 = new DataOutputStream(process0.getOutputStream());
        dataOutputStream0.writeBytes("echo \"ZZZ\" >/system/sd/tmp.txt\n");
        dataOutputStream0.writeBytes("exit\n");
        dataOutputStream0.flush();
        try {
            process0.waitFor();
            return process0.exitValue() != 0xFF;
        }
        catch (InterruptedException unused_ex) {
            return false;
        }
    }
}
```

*5bf384e687da92562fcbabac390a88110ddb2755 writes the string “ZZZ” into a text file if it can obtain super user privileges.*

BoneSpy’s surveillance features stabilized by late 2022 along with almost exclusive use of trojanized Telegram samples. BoneSpy samples observed this year had the following surveillance capabilities:

- Browser history
- SMS messages including the addressee, body, and date-time, from inbox and sent messages
- Device location from GPS and cell information
- Contact lists including name, phone number, and email address
- Call logs such as the phone number, date, name, duration, and type of call
- File system information
- List of all installed apps
- Taking photos from device cameras
- Recording phone calls
- Notification content
- Clipboard content
- Device screenshots by abusing media projection
- Device information such as IMEI, SIM cards, carrier information
- Checking for root privileges

A notable capability of BoneSpy is its ability to be controlled via SMS messages. For the extensive list of commands that the surveillance app can receive via SMS see Appendix B.

BoneSpy is based on the Russian-developed, open-source DroidWatcher surveillanceware, featuring nearly identical code, names, and log messages in multiple classes related to the handling of databases containing collected exfil data such as call logs, location tracking, SMS messages, notifications, and browser bookmarks.

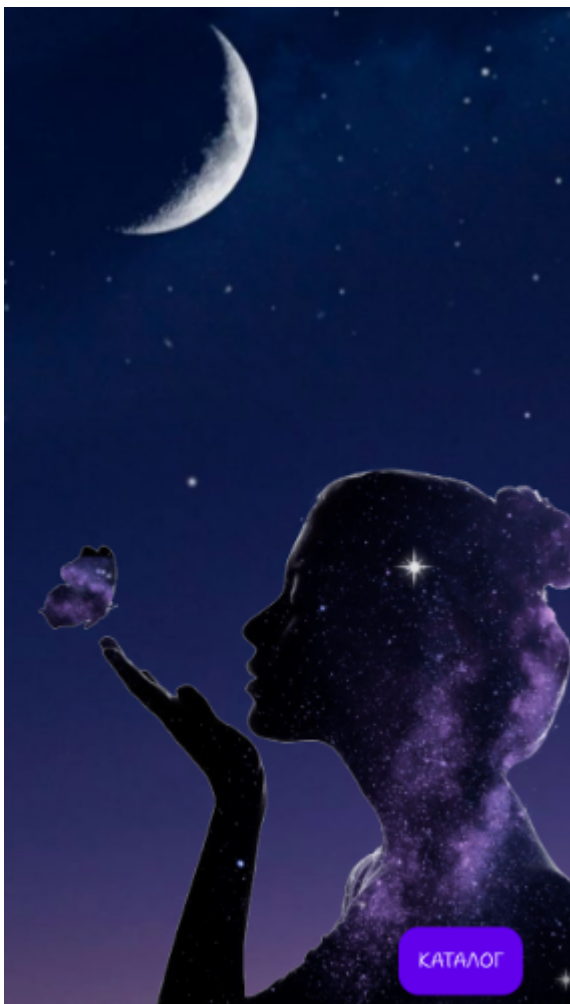
Class names for many entry points (receivers, activities, and services) were either the same or very similar to DroidWatcher Samples.

Unlike BoneSpy, PlainGnome does not share similar entry points. While most of its surveillance capabilities are similar, it appears to have been developed without extensive use of the code of another known surveillance tool.

## **Detailed Analysis: PlainGnome**

PlainGnome consists of a two-stage deployment in which a very minimal first stage drops a malicious APK once it's installed. While the first and second stages use some variation on the Telegram package name, the actual functionality presented to the user is essentially the same as that observed in previous BoneSpy samples using the "image gallery" theme. This lure theme continued through most of PlainGnome's deployment throughout 2024. PlainGnome samples generated by Lookout researchers using an actor-controlled build panel in February 2025 showed the family continued to evolve, with some newer versions deployed as single-stage applications and others designed to be deployed along with a separate app designed simply to start the PlainGnome sample.

Since it must install an APK (i.e. the surveillance payload), the first stage relies on the REQUEST\_INSTALL\_PACKAGES permission. Other than this less common permission, the first stage requests few permissions, and is lightweight in terms of code though notably contains some basic emulator checks. The victim starts the installation of the second-stage by pressing the only available button on the first stage's splash screen, which has the Russian word "каталог" (meaning catalog, listing, or directory).



*First stage app's splash screen with the “καταλογ” button.*

## The Payload

The code of PlainGnome's second stage payload evolved significantly from January 2024 through at least October. In particular, PlainGnome's developers shifted to using [Jetpack WorkManager](#) classes to handle data exfiltration, which eases development and maintenance of related code. In addition, WorkManager allows for specifying execution conditions. For example, PlainGnome only exfiltrates data from victim devices when the device enters an idle state. This mechanism is probably intended to reduce the chance of a victim noticing the presence of PlainGnome on their device.

As opposed to the minimalist first (installer) stage, the second stage carries out all surveillance functionality and relies on 38 permissions. PlainGnome's developers made no effort to obfuscate code and took very basic steps to hinder analysis. PlainGnome supports a total of 19 commands, including functionality to collect

- SMS messages
- Contacts
- GPS location

- Ambient audio
- Call audio
- Photos

A detailed list of commands is in Appendix C.

Once launched, the payload requests approval of permissions from the user until it gains access to a minimum set of permissions:

- READ\_SMS
- READ\_CALL\_LOG
- READ\_CONTACTS
- CAMERA

Notably, PlainGnome has two modes of ambient audio recording - one that automatically stops recording when the screen of the device is activated and one that permits recording regardless of the state of the screen. This is likely because newer versions of Android display a microphone icon in the status bar when the microphone is active, which might help the surveillance victim discover the malware.

## **Infrastructure, Actor Controlled PlainGnome Builder**

With the exception of some early samples, most BoneSpy as well as PlainGnome samples use the No-IP Dynamic DNS service with the ddns[.]net domain for C2 domain hosting, in addition to the \*.pw top-level domain, popularly used in domain names for Uzbekistan.

Most of the resolving IP address space associated with the BoneSpy and PlainGnome C2 domains were owned by Russian ISP [Global Internet Solutions LLC](#) (Russian: ООО Глобал Интернет Решения), [incorporated in Sevastopol, Ukraine](#), in Crimea. One notable exception is the IP 34.98.99[.]30, which resolved goos[.]pw, owned by Google Cloud. The most likely reason for this difference is that neither Global Internet Solutions, nor the later Global Connectivity Solutions (owned by the same individual), were active companies at the time that the goos[.]pw C2 domain was in use.

Attribute	Value
WHOIS Server	rdap.db.ripe.net
Registrar	RIPE
Domain Status	active
Email	<div style="background-color: #e6f2e6; padding: 2px;">+ <a href="mailto:support@gir.network">support@gir.network</a> (registrant)</div> <div style="background-color: #ffe0e0; padding: 2px;">-</div>
Name	<div style="background-color: #e6f2e6; padding: 2px;">+ <a href="#">GLOBAL INTERNET SOLUTIONS LLC</a> (registrant)  <a href="#">Oleg Pischulev</a> (tech)  <a href="#">Igor Gilmutdinov</a> (admin)</div> <div style="background-color: #ffe0e0; padding: 2px;">- <a href="#">ru-permtelecom-1-mnt</a> (registrant)  <a href="#">Oleg Pischulev</a> (tech)  <a href="#">Igor Gilmutdinov</a> (admin)</div>
Organization	-
Street	<div style="background-color: #e6f2e6; padding: 2px;">+ <a href="#">12 malkova</a> (admin, tech)  <a href="#">vn.ter.g. gagarinsky municipal district 13 mayachnaya st.</a> (registrant)</div> <div style="background-color: #ffe0e0; padding: 2px;">- <a href="#">12 malkova</a> (admin, tech)</div>
City	<div style="background-color: #e6f2e6; padding: 2px;">+ <a href="#">sevastopol</a> (registrant)  <a href="#">perm</a> (admin, tech)</div> <div style="background-color: #ffe0e0; padding: 2px;">- <a href="#">perm</a> (admin, tech)</div>

Whois records for 89.185.84[.]81, 81.19.140[.]71, 89.185.84[.]46, and 212.192.14[.]34 show registrant Global Internet Solutions LLC, located in Sevastopol, Crimea, as well as Perm, Russia. Sevastopol is the common registrant location for all BoneSpy and PlainGnome C2 resolving IP addresses.

A number of BoneSpy and PlainGnome C2 domains were hosted on alternate bulletproof provider Global Connectivity Solutions (GCS, autonomous system number 215540), with the latter’s IP infrastructure geolocated in Great Britain. Global Connectivity Solutions, LLP, [is incorporated in the UK and owned by Yevgeniy Valentinovich Marinko](#), a Russian national. Marinko also owns and is general director of Global Internet Solutions, LLC. [Marinko](#), known by aliases Rustam Yangirov or dimetr50, has operated in hacker forums and run stolen-credential trading since at least 2018. In addition, Marinko [was fined by a Sevastopol court](#) for defrauding a Russian-national victim using malware in late 2023.

Examination of BoneSpy and PlainGnome C2 domains revealed significant ties to IP infrastructure controlled by companies in Uzbekistan, as well as ETag values specific to web infrastructure associated with Uzbekistan. A distinctive ETag value, 62580d5e-3, returned in HTTP header responses, was associated with several IP addresses hosted on Uzbekistani providers as well as the previously noted Russian-owned Global Connectivity Solutions (GCS) and Global Internet Solutions (GIS) providers. Two IP addresses returning the ETag value were used to resolve two of the known BoneSpy and PlainGnome C2 domains. This relation is shown in the table below.

IP Address	C2 Domain	ASN	Geolocation
89.185.84[.]146	inspiredflow.ddns[.]net	215540, GCS	UK
195.158.2[.]220	-	8193, BRM	Uzbekistan
147.45.50[.]197	wwkravs.ddns[.]net	215540, GCS	UK
89.23.113[.]132	-	207713, GIR	Russia
195.158.2[.]211	-	8193, BRM	Uzbekistan
185.139.136[.]192	-	201767, UZMOBILE	Uzbekistan

*IP addresses returning ETag 62580d5e-3 with addresses resolving known BoneSpy and PlainGnome C2 domains*

Lookout researchers also noted popular use of the .pw top-level domain in Uzbekistan along with .uz, suggesting that PlainGnome and earlier BoneSpy C2 domains ollymap[.]pw, wleak[.]pw, and goos[.]pw are strongly associated with Uzbekistan.

The IP 185.139.136[.]192 - listed in the above table - hosted a panel to build PlainGnome samples on on port 8888 in the /builder path, as well as a “TG Bot” panel on port 5000. It was not immediately clear what the purpose of the “TG Bot” panel was, other than the probability that it was used to track users of an actor-controlled Telegram bot, which was unidentified.

## Researcher-Generated PlainGnome Samples

Lookout researchers generated multiple PlainGnome samples using the builder panel, which contained Uzbek-language names for various build options. Some options required input of a PNG image to generate the PlainGnome sample; failing to include a PNG caused a detailed Python error message that included an Uzbek-language string: “Faqat PNG formatidagi fayllarni yuklash mumkin!” (“Only PNG format files can be uploaded!”).

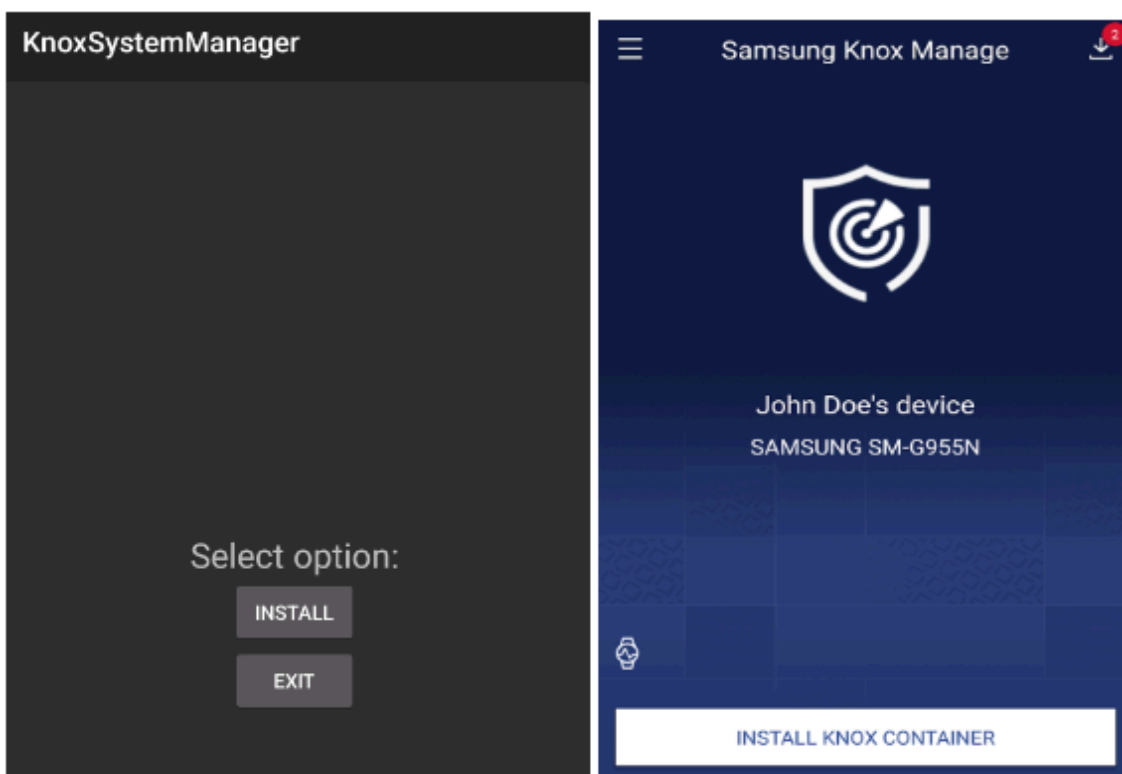
The PlainGnome samples were created through a drop-down menu of “products” with names such as Force, Test, Telegram, Tasbih (“prayer beads” in Uzbek), App Release, or Image Force. One option would build a PlainGnome sample that would open a link entered by the threat actor on the builder panel. The panel also offered an “activity launcher” app, which simply sends an intent for the MainActivity for PlainGnome samples whose package name is com.sjapps.settingswidget, as seen in two of the samples Lookout researchers built from the panel. The activity

starter app, listed separately on the builder panel, has the hash 39efa0d20b740cd45feaac4d25981d72d3e2fa7b. As with prior PlainGnome samples, at least two samples extract a second stage APK from the first stage application. The remaining samples were single-stage versions of PlainGnome. This suggests that PlainGnome has continued to evolve with new single-stage variations as well as those deployed with a separate “starter” app.

Most of the researcher-built samples and their secondary payloads used previously reported PlainGnome command and control domains, ltkwark.ddns[.]net and a secondary hardcoded domain, wleak[.]pw. However, two single-stage samples used a previously unknown C2 domain, wstak[.]pw. This domain is still active at the time of reporting, resolving to 89.23.113[.]10. This IP address, hosted by Global Internet Solutions, is in the same subnet as 89.23.113[.]32, which returned the ETag value noted above.

## Victims

Perhaps the most targeted BoneSpy sample, which has the title “KnoxSystemManager”, attempts to masquerade as Samsung Knox Manage, designed to enable enterprise mobility management on Samsung devices. Since Knox Manage is an enterprise service, this sample suggests that BoneSpy may have been deployed against targeted enterprise victims, with the attacker posing as an internal IT administrator.

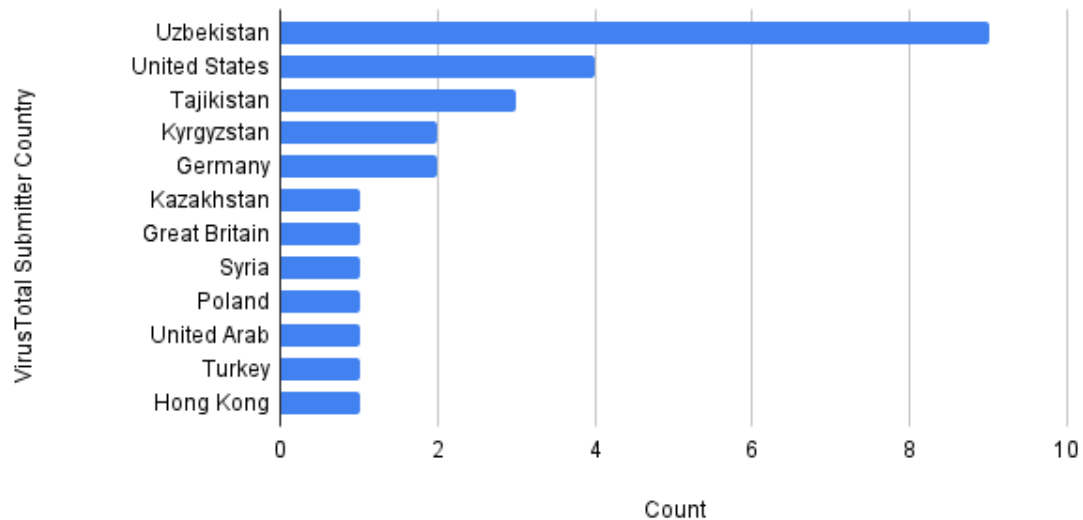


*The KnoxSystemManager BoneSpy sample (left) presents an extremely basic activity with “Install” and “Exit” options; the real Samsung Knox Manage (right) presents full EMM functionality.*

While not a direct indicator of deployment geography, VirusTotal submissions of known BoneSpy and PlainGnome samples indicate targeting in former Soviet states such as Uzbekistan, Kyrgyzstan, Tajikistan. Based on Amnesty International’s 2019 report, Sandcat targeted academic and government organizations in countries neighboring Uzbekistan, as well as human-rights defenders in Uzbekistan.

## VirusTotal Submissions by Country

PlainGnome and BoneSpy Combined Count



*VirusTotal web submissions by country for BoneSpy and PlainGnome, all time*

Additional indicators of targeting are present in use of app lures - particularly Telegram - and Russian-language filenames and promotional strings such as those found in the BoneSpy sample `cd6ee49b224ccb169d5d7f1b85c476cfc253540f`. The actor later apparently shifted away from Russian-language APK filenames. Consistent use of trojanized Telegram samples indicates eastern European targeting to some degree, as the app is highly popular in that region. The table below shows some early BoneSpy samples that were submitted to VirusTotal with Russian-language filenames.

Package Name	SHA1	Filename	Translation
com.project.photogallery	aae4fe35ffcd086253e60825e53269590f917bbc	Личный.apk	Personal
com.project.photogallery	5bf384e687da92562fcbabac390a88110ddb2755	Альбом.apk	Album
org.telegram.messenger.beta	2ec79ffa20d8da7282842925481cbd39c7ef6b46	Фотоальбом.apk	Photo album
im.vector.app	03b18e8e7d414d25930a29ae2976e1ddb5e0fe5c	galareya.apk	Gallery

*Russian-language filenames found in some early BoneSpy samples from 2022.*

## Conclusion

Lookout previously attributed BoneSpy and PlainGnome to Russia-aligned cyber espionage threat group Gamaredon (aka Primitive Bear, Shuckworm). This group was identified as a component of the Russian Federal Security Service (FSB) by the [Security Service of Ukraine \(SSU\)](#) in 2021. Following a tip from the security research community, further research and new evidence showed that this attribution was incorrect and instead pointed to an Uzbekistan-based threat actor. The article has been updated to reflect this new evidence.

## Appendix A - Indicators of Compromise

### Sample SHA-256

2c7827f92a103db1b299f334043fbdc73805bbbee11f4bfac195f672ba0464d22  
 114d2a25bb4c296f8ef5bfca4e8192b5aca9b169099ac6291139e68cfc7e37dc  
 8af63d7aa2142701116207f61e3e01c9e0239731e5bbddf79114889b56ca46ea  
 ce6e5838f3ada452b64ffc6261e9bf74479bd31e83f77c7409c89564846db6a3  
 8407fed605805f0e7ef9628767d0aff1014e7231549b09f3c0d0cb723f07c48a  
 cb648ba5cce810e5ba17b89ca2c346bd3f0ad612834c225ec7b55871c4acc085  
 39cb17cb03a794e69eb4f0694e90e41a8cfb8480b82da82fcbd4a88dfe49930d

fd5fa718a7411b18845b76d7007db6b4431b1a2ce2f8b2cc047c0fff7c46161f  
f0acf9558b7a4fcdad119731ad5fb5bbdf5a704c9be9e929735a4679735989db  
7de055018723b612dfa66a90c83a69afce7db918fb7fa88619833557c4fc61c3  
551b8917f57c5cf8cd0a34c1d500db1dd4aed8ec8f31d28a5fabc4720e5b89a3  
533ff7ba5eb5329cb860486a952259a4dfc0d74654831eb08dbcadc1ae5ca333  
acede5fa46e09803adf9de5e731ca690dc7b02b69a63bacd4836429d289ec4f0  
a3b0c178ab5e6e4b3442d358a78df7409461fa48f6ca8e63b730b0a455a89b18  
7a8ec25f3d4a5c6b4fbd1002ce22ff0352ce65c0f4ddc9567458e8fcb964845  
86e51f1cc8213e173e47080ab45577e922e624006954de73ebae531589c912f4  
2ef72c67cf76e8162f5e4bf0a743ac4ed756e153593c430cedf2043a310b24e8  
5b7b5a2995c102121695225797f12f0b860500150472126b3b465b51ccad07bd  
9dd73c9caa547358b6fe5acddf59443d7bf0ffc5b92867e9b67edd5bb2a9f786  
3b5794ca6051740fff6e1b449db06f169df2749f81aaf4c329e18b12afb9a5c7  
dfaa47ed20021c4f84bf68820a618f9e8a2e077d36b6d7281e8724b2124c7825  
f948b650bdc63cf9b1781d651974a9c54d2b2981d3bf4b882f48c3a406272470  
c82f0a1546bf7025993f2e7da33d1a741d91c78b01268a2d44afa31e66eb2fe8  
de3a0b30b8976da933fe6bf88e6e7ab2386a967ada2599ef1dc1b12100a37694  
bd65dbd61f27a90c0770d5f8cc02cfa7d9552f0fb300868611d69972b42d3f1c  
bed2cf8758d86daaf25475cc6ed1c71fd3f9a922247c42fe246f8542c76d8c15  
255996e1aa2a7514b167d9c940d7c8ff3c34393e97e43bda319eb92ea626c4eb  
46b10de13887c36d61517125bec87c4557f325114221291a3ac7142cbc15de29  
6bfdc285dee8ae3e3dade52a34f5d178163e4a08904b651ff5c906e78ddcce0  
e0c5656ca9877b37e92f5208caf9c65365e9d35ea6eb351915eb3efee235db31  
30429e95b9318816709e23488c77e364a294b6f5f7e3ee414a6a2bef74620ca6  
278c9819583ce64913882d425c1d7634307b290709e0143e9268f8f999dacfba  
3a4fa698536111f377030a5d794851d2e23b18d67e6d440ce883b9906d65037d

629ca39d2c90ff8b343ba1f4cfae11bbc2f61ca6bae80bd093f22efbcf4e4770

633875ce353391ea8bd4c92d8f3f57a525ff0abf9eba8d78528de616b1ee7118

eadd9c3e3f7a1c5e008ca157cb850aa72d283f702da2ab4daf0e4af4d926ab3e

## **C2 Server Domains**

llkeyvost.ddns[.]net

fiordmoss.ddns[.]net

winterknowing.ddns[.]net

weeklyoptional.ddns[.]net

ltkwark.ddns[.]net

ollymap[.]pw

wleak[.]pw

wstak[.]pw

goos[.]pw

## **Appendix B - SMS Commands**

SMS Command String	Result
setSettings	Saves settings values to BoneSpy's shared preferences
getSettings	Gets BoneSpy's settings from shared preferences
calendar	Retrieves calendars and calendar events
<u>deletelogcall</u>	Deletes a call log entry by its phone number
<u>gpson</u>	Enables device GPS via a boolean value stored in BoneSpy's shared preferences.
<u>gpsoff</u>	Disables device GPS via a boolean value stored in BoneSpy's shared preferences.
<u>gpsget</u>	Gets the GPS location of the device
callback	Call an arbitrary phone number
download	Download a file from a specified URL
shell	Open a shell to the victim device; if root access is available, run the shell as root
reset	Clears all settings from BoneSpy's shared preferences
sms	Gets SMS messages from the device
connect	Initiates an authenticated session with the C2 server and starts the main AppService class as a foreground service
record	Enables call recording based on a boolean value stored in BoneSpy's shared preferences
recordstop	Disables call recording based on a boolean value stored in BoneSpy's shared preferences
<u>wifion</u>	Enables Wi-Fi on the device
wifioff	Disables Wi-Fi on the device
<u>gprson</u>	Enable cellular data use on the device
<u>gprsoff</u>	Disable cellular data use on the device
reboot	Reboots the device, if root access is available
restart	Restarts BoneSpy's main AppService foreground service
contact	Gets the victim's contact list
applist	Gets a list of installed apps
<u>sendsms</u>	Send an SMS message with arbitrary content to an arbitrary phone number
sendsms1	Alternative to "sendsms"; send an SMS with a .. ..

	direct call to SmsManager.sendTextMessage()
<u>wipesd</u>	Wipe all files and directories on the SD card
<u>allog</u>	Retrieves all exfil logs including SMS, call log, GPS logs, clipboard contents, notifications, browser history, and installed apps
<u>deletelastsms</u>	Deletes the most recent SMS from the content://sms/ provider
smsblock	Block inbound SMS
screenshoot	<u>Takes</u> a screenshot.
<u>screenshootenable</u>	Enables screenshots based on a boolean value stored in BoneSpy's shared preferences.
<u>screenshootdisable</u>	Disables screenshots based on a boolean value stored in BoneSpy's shared preferences.

<u>screenshootdisable</u>	Disables screenshots based on a boolean value stored in BoneSpy's shared preferences.
whatsappoffline	Unknown, presumably opens WhatsApp in offline mode
requestrecord	unknown/not implemented
whatsapp	Get authentication token for WhatsApp
facebook	Get authentication token for Facebook
vb	Get authentication token for Viber
vk	Get authentication token for VKontakte
imo	Get authentication token for IMOIM
ondoklassniki	Get authentication token for Odnoklassniki
<u>mailruagent</u>	Get authentication token for mail.ru
telegramm	Get authentication token for Telegram
TelegrammPlus	Get authentication token for Telegram Plus
camera	Starts CameraService, which allows for taking photos from device cameras
<u>getsdcardstruct</u>	Gets the file structure of the SD card
<u>frontcameraon</u>	Enables the front camera using a boolean value stored on BoneSpy's shared preferences
<u>frontcameraoff</u>	Disables the front camera using a boolean value stored on BoneSpy's shared preferences
deviceinfo	Gets updated device information and passes it to the C2
debugrealtime	Unknown, probably enables debug-level logging
<u>messangers</u>	Checks for Facebook Messenger
<u>open, close, getconfig, setconfig, autorecoron, autorecoroff</u>	unknown/not implemented

## Appendix C - PlainGnome Commands

Commands supported by PlainGnome. Note that the last two commands appear only in later samples.

Command string	Result
get_sms	Gets SMS messages from the victim device in JSON format, including the address, message body, date, and message type
get_call_log	Retrieves the call logs in JSON format, including the victim device IMEI, contact name, number, direction (inbound/outbound), duration, and date.
get_contacts	Packages victim contacts in a JSON including the victim IMEI, contact name, contact phone number, and date.
upload	Uploads any files to the /upload path on the C2.
gps_on	Starts locationService or locationServiceLow foreground services (depending on victim's Android API level) to capture the device's location.
gps_off	Stops locationService or locationServiceLow.
start_record	Starts the RecordingService foreground service, which captures microphone audio, if RECORD_AUDIO permission is granted.
stop_record	Stops RecordingService.
start_record_screen_off	Enables ambient audio recording only when the device screen is off.
stop_record_screen_off	Disables ambient audio recording when the screen is off, by stopping RecordingService.
start_record_always	Enables constant ambient audio recording
stop_record_always	Disables constant ambient audio recording by stopping RecordingService.
screen_status	Returns a JSON with the screen status (on or off).
gps_status	Returns a JSON containing whether the device GPS is enabled, and whether the locationService is running.
perm_status	Returns a JSON containing whether READ_CONTACTS, READ_SMS, READ_CALL_LOG, WRITE_EXTERNAL_STORAGE, ACCESS_FINE_LOCATION, RECORD_AUDIO permissions are granted, and whether the device is connected to the internet.
start_photo	Takes a photo from the device camera with the takePhoto class and uploads it to the C2.
delete_kesh	Deletes the local cache.
ls	List contents of a path much like the standard

	Unix command
cash	Unclear, but presumably retrieves cached exfil data

---

Source: <https://www.lookout.com/threat-intelligence/article/gamaredon-russian-android-surveillanceware>