UNIVERSITY OF TORONTO

MUNK SCHOOL OF GLOBAL AFFAIRS

Join the Global Conversation

The Citizen Lab

Research Brief
May 2015

A Chatty Squirrel:
Privacy and Security Issues with UC Browser

Authors: Jakub Dalek (lead), Katie Kleemola (lead), Adam Senft (lead),
Christopher Parsons, Andrew Hilts, Sarah McKune, Jason Q. Ng,
Masashi Crete-Nishihata, John Scott-Railton, Ronald Deibert

SECTION 1 - INTRODUCTION & OVERVIEW

UC Browser is the most popular mobile web browser in China and India, boasting over 500 million users. This report provides a detailed analysis of how UC Browser manages and transmits user data, particularly private data, during its operation. Our research was prompted by revelations in a document leaked by Edward Snowden on which the Canadian Broadcasting Corporation (CBC) was preparing a story. The CBC contacted us requesting our comment. The document, apparently prepared in 2012 by Canada's signals intelligence agency, the Communications Security Establishment (CSE), noted the existence of security vulnerabilities in UC Browser. Given the Citizen Lab's ongoing research into popular Asian communications tools, and the possibility of vulnerabilities affecting a large number of users, we decided to conduct an independent investigation of UC Browser. While media outlets are publishing a story about the CSE document, we cannot determine if the problems we identify in UC Browser and that are described in this report are identical to those referenced in the 2012 CSE document.

SUMMARY OF FINDINGS

We have identified a series of major security and privacy issues in the English language and Chinese language editions of the Android version of UC Browser. Our notification to the parent companies is described below in detail. We found that both versions of the application leak a significant amount of personal and personally-identifiable data; as a result, any network operator or in-path actor on the network can acquire a user's personally identifiable information (including cellular subscriber information, mobile device identifiers, geolocation data, and search queries) through trivial decrypting of traffic or by observing unencrypted traffic. Specifically, the issues we found include:

Transmission of personally identifiable information and user search queries without encryption:

- User data, including IMSI, IMEI, Android ID, and Wi-Fi MAC address are sent without encryption to Umeng, an Alibaba analytics tool, in the Chinese language version.
- User geolocation data, including longitude/latitude and street name, are transmitted without encryption by AMAP, an Alibaba mapping tool, in the Chinese language version.
- User search queries are sent without encryption to the search engine Shenma (in the Chinese language version) or Yahoo! India and Google (in the English language version).
- Reason for concern: The transmission of personally identifiable information, geolocation data and search queries without encryption represents a privacy risk for users because it allows anyone with access to the data traffic to identify users and their devices, and collect their private search data.

Transmission of personally identifiable information and geolocation data with easily circumvented encryption:

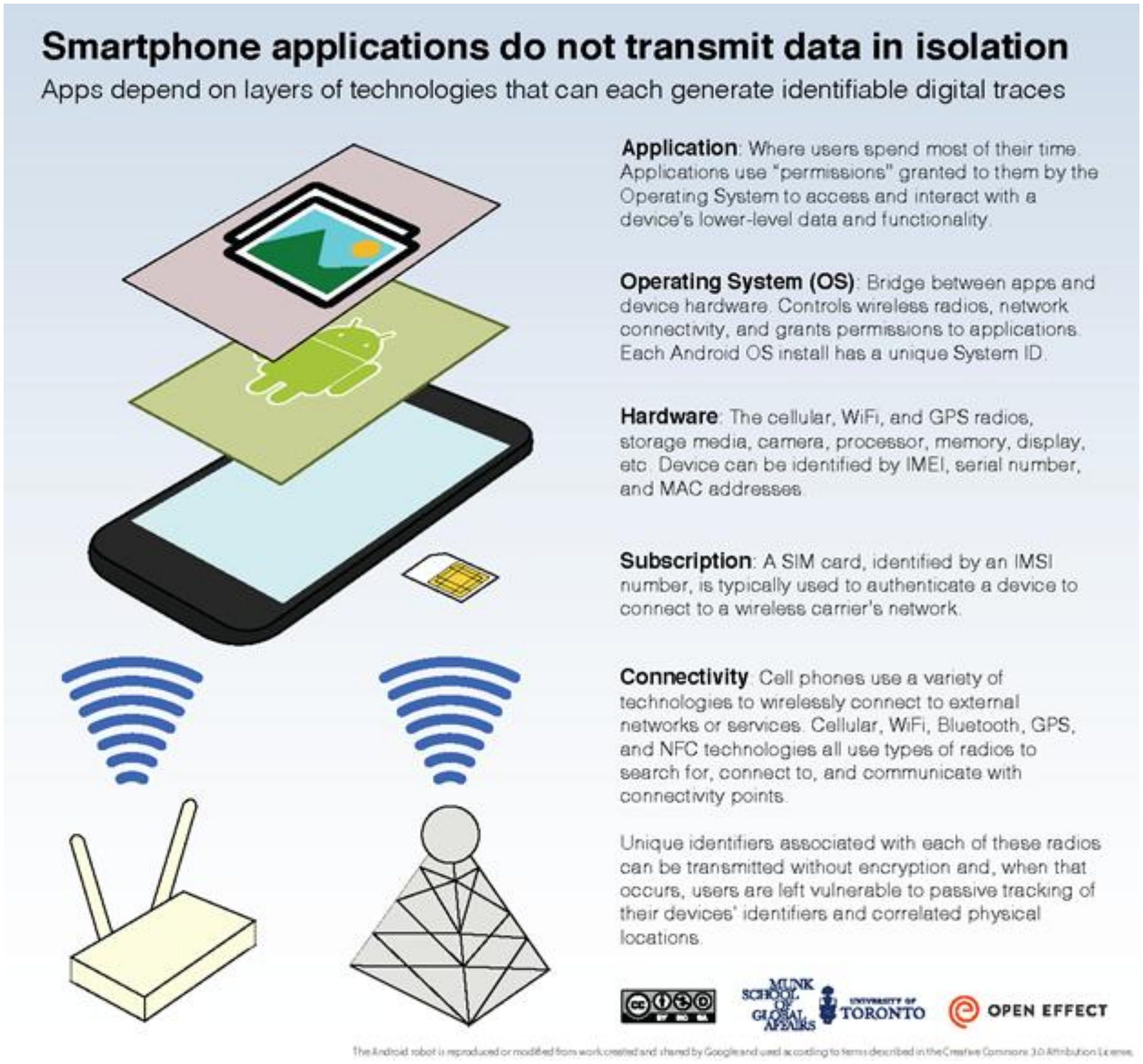
- Location and user data, including IMSI, IMEI, and data about nearby cellular towers and Wi-Fi access points, are sent with easily circumvented encryption by AMAP, an Alibaba mapping tool, in the Chinese language version.
- Reason for Concern: UC Browser's transmission of personally identifiable subscriber data, mobile device identifiers, and user geolocation data without effective encryption presents a security and privacy risk for users.

Private user data is retained on the device even after clearing the application's cache:

- In the Chinese language version, when users attempt to delete their private data by clearing the application's cache their DNS lookups are not deleted.
- Reason for concern: The cached record of DNS lookup data would allow for a third party with access to the device to identify the websites that a user visited.

This report is a continuation of our prior work examining the security and privacy of popular mobile applications in Asia. Our previous research includes investigations of censorship practices of search engines offered by Google, Microsoft, and Yahoo! in the Chinese market along with domestic Chinese search engine Baidu. In addition, we have analyzed keyword censorship and surveillance in TOM-Skype (the Chinese version of Skype at the time) and keyword censorship in Sina UC, another Chinese instant messaging platform. We are currently conducting comparative analysis of mobile chat applications used in Asia including WeChat, LINE, and KakaoTalk.

For those who may be interested in more context on mobile security and privacy, please see our general introduction to mobile communications. Click on the image below to read "The Many Identifiers in our Pockets."



Smartphone applications do not transmit data in isolation

Apps depend on layers of technologies that can each generate identifiable digital traces

Application: Where users spend most of their time. Applications use "permissions" granted to them by the Operating System to access and interact with a device's lower-level data and functionality.

Operating System (OS): Bridge between apps and device hardware. Controls wireless radios, network connectivity, and grants permissions to applications. Each Android OS install has a unique System ID.

Hardware: The cellular, WiFi, and GPS radios, storage media, camera, processor, memory, display, etc. Device can be identified by IMEI, serial number, and MAC addresses.

Subscription: A SIM card, identified by an IMSI number, is typically used to authenticate a device to connect to a wireless carrier's network.

Connectivity: Cell phones use a variety of technologies to wirelessly connect to external networks or services. Cellular, WiFi, Bluetooth, GPS, and NFC technologies all use types of radios to search for, connect to, and communicate with connectivity points.

Unique identifiers associated with each of these radios can be transmitted without encryption and, when that occurs, users are left vulnerable to passive tracking of their devices' identifiers and correlated physical locations.

The Android robot is reproduced or modified from work created and shared by Google and used according to terms described in the Creative Commons 3.0 Attribution License.

Figure 1: Overview of mobile device data transmission.

NOTIFICATION

We disclosed our findings to Alibaba and UCWeb on April 15, 2015, and informed them that we would publish this report on or after April 29, 2015. Alibaba responded to our notification on April 19, 2015, indicating that their security engineers were investigating the issue. We followed up on April 23, 2015 to reiterate our intention to publish this report on or after April 29, 2015. As of May 19, 2015 we have not received further communication from Alibaba or UCWeb. After repeated efforts to contact them resulted in no further replies, we have decided to publish our findings.

SECTION 2 - UC BROWSER: QUICK BACKGROUND

UC Browser is a mobile web browser for Android, IOS, Windows Phone, and other platforms. A Windows version was released in April 2015. The application is the flagship product of UCWeb Inc., a Guangzhou, China-based company founded in 2004. After an initial investment by e-commerce giant Alibaba, the two companies launched the joint mobile search service Shenma. Shenma reportedly has more than 100 million users per month. In June 2014, Alibaba purchased the remaining stake in UCWeb in the biggest ever merger of Chinese Internet firms.

UC Browser is among the most popular mobile apps in the Chinese Internet space. UC Browser claims to have more than 500 million registered users, and is reported to be the most popular mobile browser in China and India. Overall, the application is the fourth most popular mobile browser globally, and is behind only pre-installed Chrome, Android, and Safari browsers.

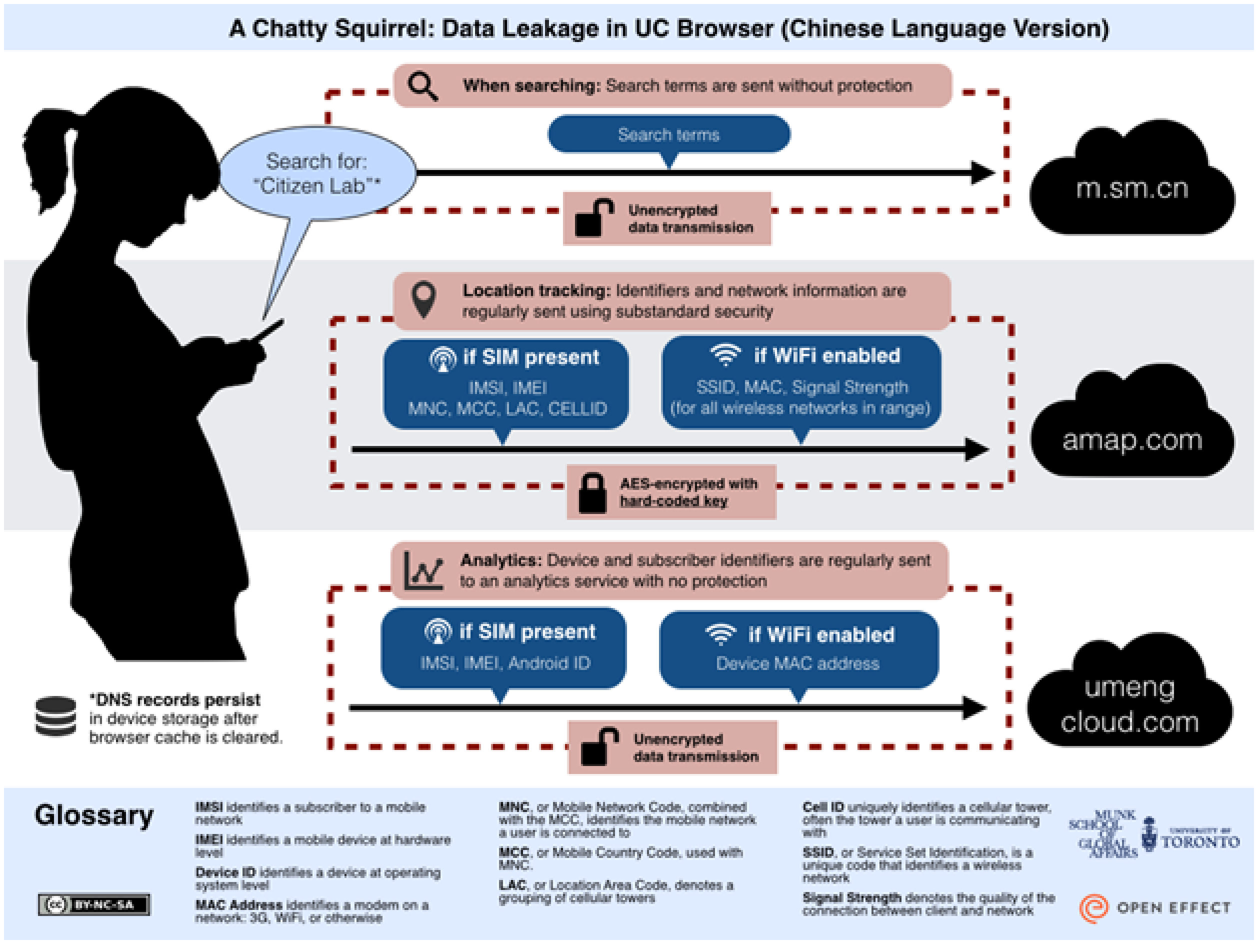
UCWeb Inc. claims the app has 100 million daily active users, while parent company Alibaba's 2014 prospectus reported the number of active users at 264 million in June 2014. UC Browser was ranked as the second most popular app by usage in China in January 2013. The company has also increased its global push, and claims it has at least 10 percent market share in 10 different countries.

UC Browser offers a custom default homepage with links to search engines and social media integration, as well as news, weather, and shopping services. A set of features are aimed at reducing bandwidth usage on mobile clients. "Cloud download," for example, allows users to send downloads directly to UDisk (a UC cloud offering) in order to save on bandwidth costs. In addition to this feature, UC Browser can act as an optional proxy and compress web sites it fetches to reduce bandwidth consumption.

SECTION 3 - METHODOLOGY AND TECHNICAL ANALYSIS

This section describes the methods we used to analyze UC Browser, and presents detailed findings from our analysis.

We isolated specific versions of the Chinese- and English-language builds of UC Browser for Android and analyzed their mobile (cellular network) data and Wi-Fi traffic. We also analyzed the application's data retention and deletion practices. Our analyses revealed major privacy and security issues with all of the tested versions of UC Browser. Figure 2 highlights the major findings for the Chinese language version of UC Browser.


A Chatty Squirrel: Data Leakage in UC Browser (Chinese Language Version)
When searching: Search terms are sent without protection
Search terms
Search for: "Citizen Lab"*
Unencrypted data transmission
m.sm.cn
Location tracking: Identifiers and network information are regularly sent using substandard security
if SIM present
IMSI, IMEI
MNC, MCC, LAC, CELLID
if WiFi enabled
SSID, MAC, Signal Strength (for all wireless networks in range)
AES-encrypted with hard-coded key
amap.com
Analytics: Device and subscriber identifiers are regularly sent to an analytics service with no protection
if SIM present
IMSI, IMEI, Android ID
if WiFi enabled
Device MAC address
Unencrypted data transmission
umeng cloud.com
*DNS records persist in device storage after browser cache is cleared.
Glossary
BY-NC-SA
MUNK SCHOOL OF GLOBAL AFFAIRS
UNIVERSITY OF TORONTO
OPEN EFFECT

IMSI identifies a subscriber to a mobile network

IMEI identifies a mobile device at hardware level

Device ID identifies a device at operating system level

MAC Address identifies a modem on a network: 3G, WiFi, or otherwise

MNC, or Mobile Network Code, combined with the MCC, identifies the mobile network a user is connected to

MCC, or Mobile Country Code, used with MNC.

LAC, or Location Area Code, denotes a grouping of cellular towers

Cell ID uniquely identifies a cellular tower, often the tower a user is communicating with

SSID, or Service Set Identification, is a unique code that identifies a wireless network

Signal Strength denotes the quality of the connection between client and network

Figure 2: A visual summary of privacy and security issues presented by UC Browser.

TEST SETUP

We isolated specific versions of the Chinese- and English-language builds in order to examine UC Browser's security and privacy features. Specifically, we monitored the data that was transmitted between the application and external servers. We were specifically interested in what, if any, personally identifiable information was transmitted by UC Browser, and whether encryption was used to secure those transmissions. We analyzed the state of the application, both in-idle state (soon after the app was opened) as well as during use of the app's features, such as searching. Lastly, we examined the data UC Browser stored on the device, and whether that data was protected with encryption.

Tests were conducted within an Android emulator and on an Android handset. All traffic sent to and from the device was collected and analyzed using the packet-capture utility WireShark. We decompiled the downloaded APKS with APKtool and then analyzed the code for functionality related to the transmission of user data.

VERSIONS ANALYZED

We analyzed two versions of UC Browser for Android. We downloaded the two versions from different app stores: the Chinese-language version of UC Browser (UC浏览器) was downloaded in March 2015 from the Xiaomi mobile app store. Henceforth we will refer to this app as **UC Browser (Chinese)** to distinguish between versions. We downloaded the English version, henceforth **UC Browser (English)**, in April 2015 from the UCWeb website. These two versions have differences beyond language: by default, the Chinese version uses Shenma (sm.cn) for search, while the English version uses Yahoo! India and Google; the Chinese version has links to China-based services such as Baidu, Sina Weibo, and Youku, while the English version uses services such as Google, Facebook, and Twitter.

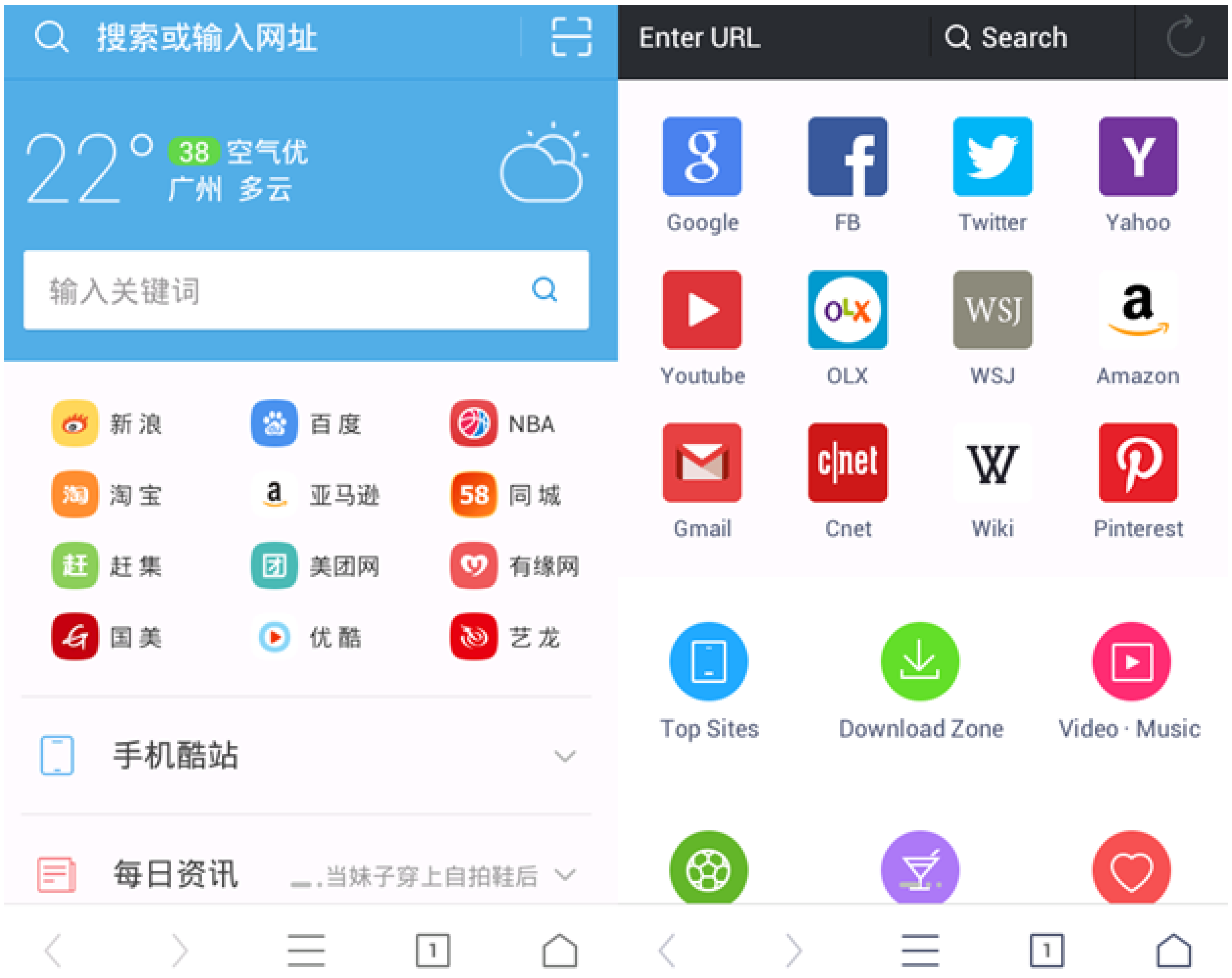
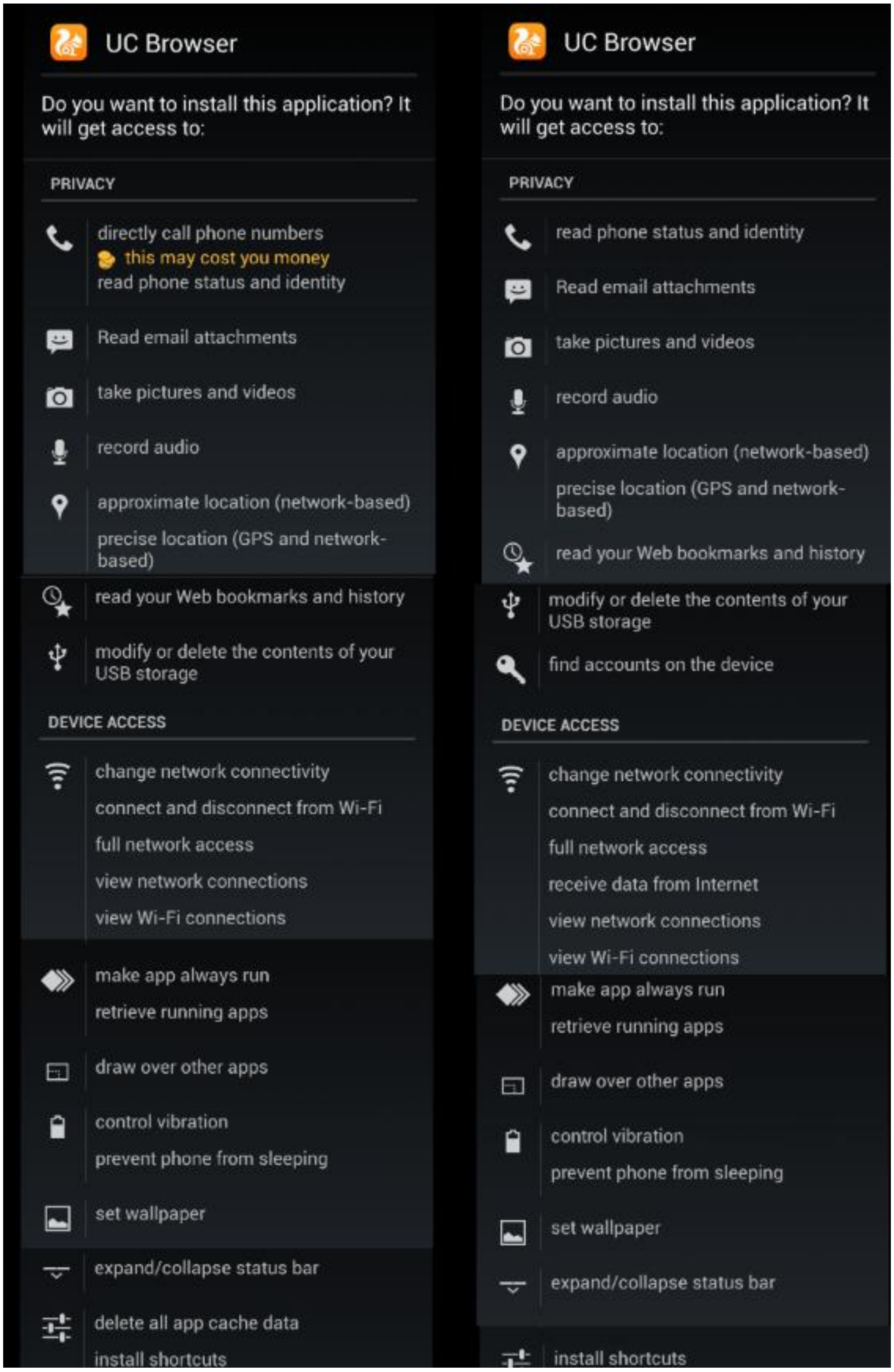


Figure 3: Side-by-side comparison of UC Browser (Chinese) and UC Browser (English).



UC Browser

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

PRIVACY

directly call phone numbers
this may cost you money
read phone status and identity

Read email attachments

take pictures and videos

record audio

approximate location (network-based)
precise location (GPS and network-based)

read your Web bookmarks and history

modify or delete the contents of your USB storage

DEVICE ACCESS

change network connectivity
connect and disconnect from Wi-Fi
full network access
view network connections
view Wi-Fi connections

make app always run
retrieve running apps

draw over other apps

control vibration
prevent phone from sleeping

set wallpaper

expand/collapse status bar

delete all app cache data
install shortcuts

UC Browser

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

PRIVACY

read phone status and identity

Read email attachments

take pictures and videos

record audio

approximate location (network-based)
precise location (GPS and network-based)

read your Web bookmarks and history

modify or delete the contents of your USB storage

find accounts on the device

DEVICE ACCESS

change network connectivity
connect and disconnect from Wi-Fi
full network access
receive data from Internet
view network connections
view Wi-Fi connections

make app always run
retrieve running apps

draw over other apps

control vibration
prevent phone from sleeping

set wallpaper

expand/collapse status bar

install shortcuts

Figure 4: Side-by-side comparison of permissions requested during installation of UC Browser (Chinese) (left) and UC Browser (English) (right) versions of UC Browser.

Figure 4 shows the permissions that the Chinese and English versions of UC Browser requires when being installed on an Android device. For the purposes of this report, it is notable that the application requests access to phone status and identity, geolocation information, the ability to read Web bookmarks and history, as well as to extensive networking information. As discussed further in the report, UC Browser (Chinese) accesses phone identity and geolocation information and transmits it insecurely.

Table 1 summarizes the two versions of UC Browser we analyzed:

Platform	**Version (Language)**	**Version**	**Date Downloaded**	**Source**
Android	UC Browser (Chinese) (UC浏览器)	10.2.1_161	March 12, 2015	Xiaomi App Storehttp://app.mi.com/detail/1363
Android	UC Browser (English)	10.4.1.565	April 10, 2015	Direct from UC websitehttp://www.ucweb.com/ucbrowser/download/android.html

3.1 UC BROWSER (UC浏览器) CHINESE LANGUAGE VERSION 10.2.1_161

This section describes our findings from analyzing the Chinese language version of UC Browser (UC浏览器), downloaded from the Xiaomi mobile app store. Our test results are summarized in Table 2:

Table 2: Summary of test results for UC Browser (Chinese).

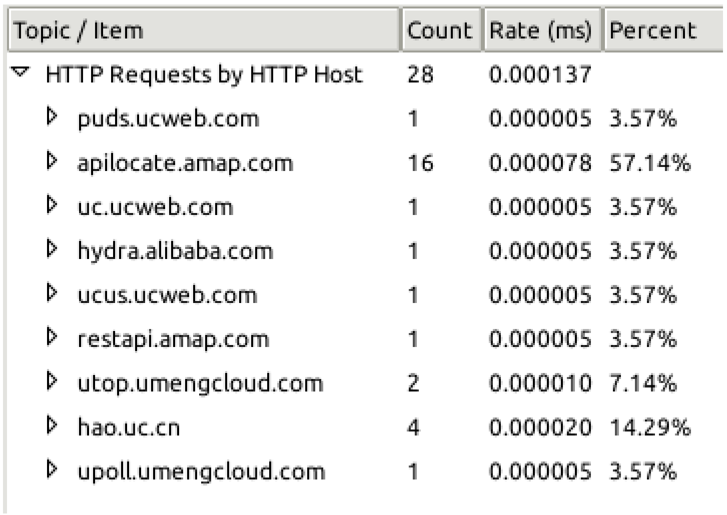
Test	**Mode**	**Results**
Idle test	Cell only	• The AMAP component of UC Browser (Chinese) contacts apilocate.amap.com, and sends user and device identifiers (IMSI, IMEI) and location data (cell tower data) with easily circumvented encryption. • The Umeng component of UC Browser (Chinese) sends device identifiers (including IMSI, IMEI, Android ID) without encryption.
Idle test	Cell + Wi-Fi	• The AMAP component sends, with easily circumvented encryption, the same data as noted in the row above (“Cell only” mode). In addition, it sends device Wi-Fi MAC address, SSID, and MAC address of the Wi-Fi access point to which the user is connected, and nearby wireless access points. • The Umeng component sends the same device identifiers as noted in the row above (“Cell only” mode), with the addition of the device Wi-Fi MAC address, without encryption.
Search	Cell only & Cell + Wi-Fi	• Search queries using the search bar are sent to Alibaba’s Shenma search engine without encryption.
Data storage	Cell only & Cell + Wi-Fi	• After clearing the application’s private data, the cached record of DNS lookup data remains.

Idle Test

In our first test, we launched the application, let UC Browser (Chinese) idle for 270 seconds, and collected all network traffic sent to and from the device. We then analyzed that traffic to determine what data was being sent and its destination. We performed the test first by connecting the mobile device to the Internet using a cellular connection, and, second, by connecting the device to the Internet using a Wi-Fi network.

Testing UC Browser (Chinese)'S Cellular-Only Communication

Upon starting the application, the application communicated with the following hosts over HTTP:



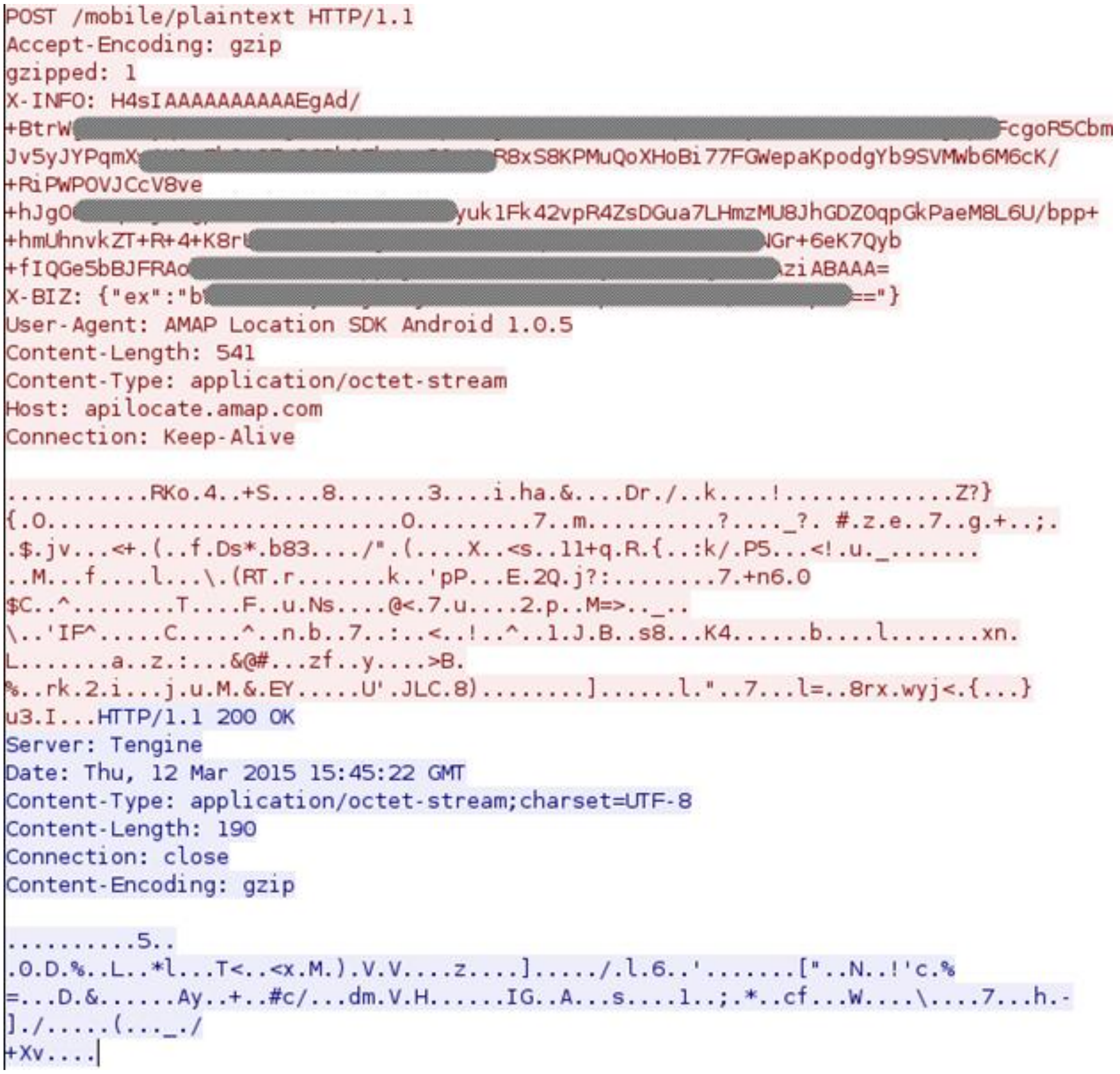
Topic / Item	Count	Rate (ms)	Percent
HTTP Requests by HTTP Host	28	0.000137	
puds.ucweb.com	1	0.000005	3.57%
apilocate.amap.com	16	0.000078	57.14%
uc.ucweb.com	1	0.000005	3.57%
hydra.alibaba.com	1	0.000005	3.57%
ucus.ucweb.com	1	0.000005	3.57%
restapi.amap.com	1	0.000005	3.57%
utop.umengcloud.com	2	0.000010	7.14%
hao.uc.cn	4	0.000020	14.29%
upoll.umengcloud.com	1	0.000005	3.57%

Figure 5: Hosts communicated with by UC Browser (Chinese) within 270 seconds of launching the app

Easily Decrypted Data Sent To AMAP

As seen in Figure 5, the majority of communications (57% of HTTP packets in our 270 second sample) are between the application and apilocate.amap.com. AMAP is a mobile mapping application that was originally developed by Autonavi, a company acquired by Alibaba in April 2014. AMAP is estimated to have more than 100 million users.

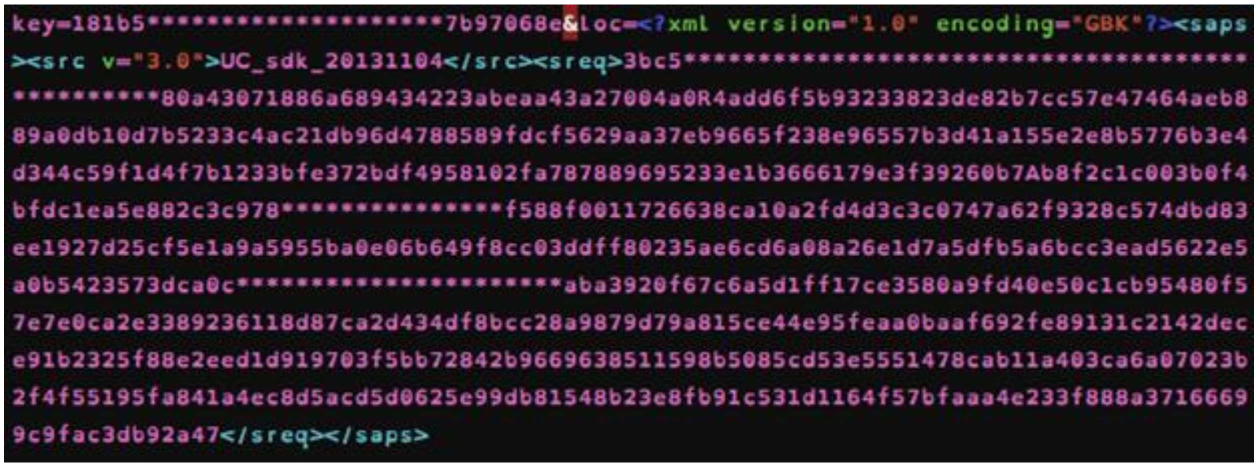
Investigating these communications further, we saw that a typical exchange with apilocate.amap.com looked as follows:



```
POST /mobile/plaintext HTTP/1.1
Accept-Encoding: gzip
gzipped: 1
X-INFO: H4sIAAAAAAAAAAEgAd/
+BtrW[redacted]FcgoR5Cbm
Jv5yJYPqmX[redacted]R8xS8KPMuQoXHoBi77FGWepaKpodgYb9SVMWb6M6cK/
+RiPWPOVJCcV8ve
+hJgO[redacted]yuk1Fk42vpR4ZsDGua7LHmzMU8JhGDZ0qpGkPaeM8L6U/bpp+
+hmUhnvkZT+R+4+K8r[redacted]Gr+6eK7Qyb
+fIQGe5bBJFRAo[redacted]ziABAAA=
X-BIZ: {"ex":"b[redacted]=="}
User-Agent: AMAP Location SDK Android 1.0.5
Content-Length: 541
Content-Type: application/octet-stream
Host: apilocate.amap.com
Connection: Keep-Alive

...........RKo.4..+S....8.......3....i.ha.&....Dr./..k....!.............Z?}
{.O..........................O.........7..m..........?...._?. #.z.e..7..g.+..;.
.$.jv...<+.(..f.Ds*.b83..../".(....X..<s..1l+q.R.{..:k/.P5...<!.u._.......
..M...f....l...\.(RT.r.......k..'pP...E.2Q.j?:........7.+n6.0
$C..^........T....F..u.Ns....@<.7.u....2.p..M=>.._..
\..'IF^.....C.....^..n.b..7..:..<..!..^..1.J.B..s8...K4......b....l.......xn.
L.......a..z.:...&@#...zf..y....>B.
%..rk.2.i...j.u.M.&.EY.....U'.JLC.8)........]......l."..7...l=..8rx.wyj<.{...}
u3.I...HTTP/1.1 200 OK
Server: Tengine
Date: Thu, 12 Mar 2015 15:45:22 GMT
Content-Type: application/octet-stream;charset=UTF-8
Content-Length: 190
Connection: close
Content-Encoding: gzip

..........5..
.O.D.%..L..*l...T<..<x.M.).V.V...z....]...../.l.6..'.......["..N..!'c.%
=...D.&......Ay..+..#c/...dm.V.H......IG..A...s....1..;.*..cf...W....\....7...h.-
]./.....(..._./
+Xv....
```

Figure 6: Sample communication between UC Browser (Chinese) and apilocate.amap.com. The user-agent string indicates this communication is from "AMAP Location SDK Android 1.0.5." We have redacted personally identifiable data.

As we were interested in determining what, if any, data the application transfers from the device, the block of data sent to apilocate.amap.com was of interest. We used a freely available tool, pyhttpextract, to decipher the contents of this encoded data block. After using this tool, we saw that UC Browser (Chinese) sent the following data to apilocate.amap.com:

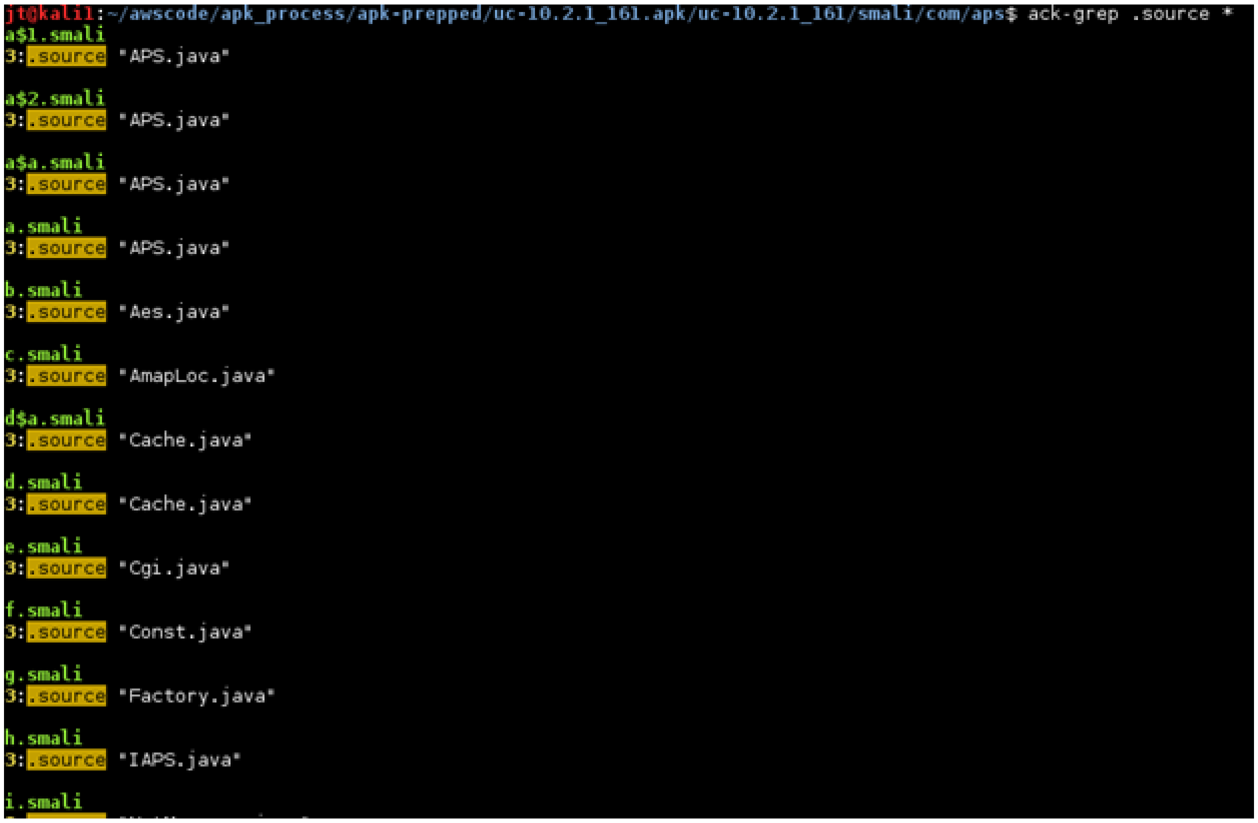


```
key=181b5*******************7b97068e&Loc=<?xml version="1.0" encoding="GBK"?><saps
><src v="3.0">UC_sdk_20131104</src><sreq>3bc5****************************************
**********80a43071886a689434223abeaa43a27004a0R4add6f5b93233823de82b7cc57e47464aeb8
89a0db10d7b5233c4ac21db96d4788589fdcf5629aa37eb9665f238e96557b3d41a155e2e8b5776b3e4
d344c59f1d4f7b1233bfe372bdf4958102fa787889695233e1b3666179e3f39260b7Ab8f2c1c003b0f4
bfdc1ea5e882c3c978***************f588f0011726638ca10a2fd4d3c3c0747a62f9328c574dbd83
ee1927d25cf5e1a9a5955ba0e06b649f8cc03ddff80235ae6cd6a08a26e1d7a5dfb5a6bcc3ead5622e5
a0b5423573dca0c*********************aba3920f67c6a5d1ff17ce3580a9fd40e50c1cb95480f5
7e7e0ca2e3389236118d87ca2d434df8bcc28a9879d79a815ce44e95feaa0baaf692fe89131c2142dec
e91b2325f88e2eed1d919703f5bb72842b9669638511598b5085cd53e5551478cab11a403ca6a07023b
2f4f55195fa841a4ec8d5acd5d0625e99db81548b23e8fb91c531d1164f57bfaaa4e233f888a3716669
9c9fac3db92a47</sreq></saps>
```

Figure 7: Deciphered communication between UC Browser (Chinese) and apilocate.amap.com. Personally identifiable data redacted with asterisks.

The encoded data sent within the ‘’ structure intrigued us because it was a relatively large block of data. Its size suggested that it might contain user data. To confirm if user data was present we first analyzed how the data was encrypted. Using apktool we decompiled UC Browser (Chinese) to see how the application created and serialized the ‘’ structure. After decompiling, we looked for the string ‘sreq’ in the outputted code and found this in a directory associated with the com/aps class (Android programs often incorporate a mix of components, called classes, sometimes from different developers).

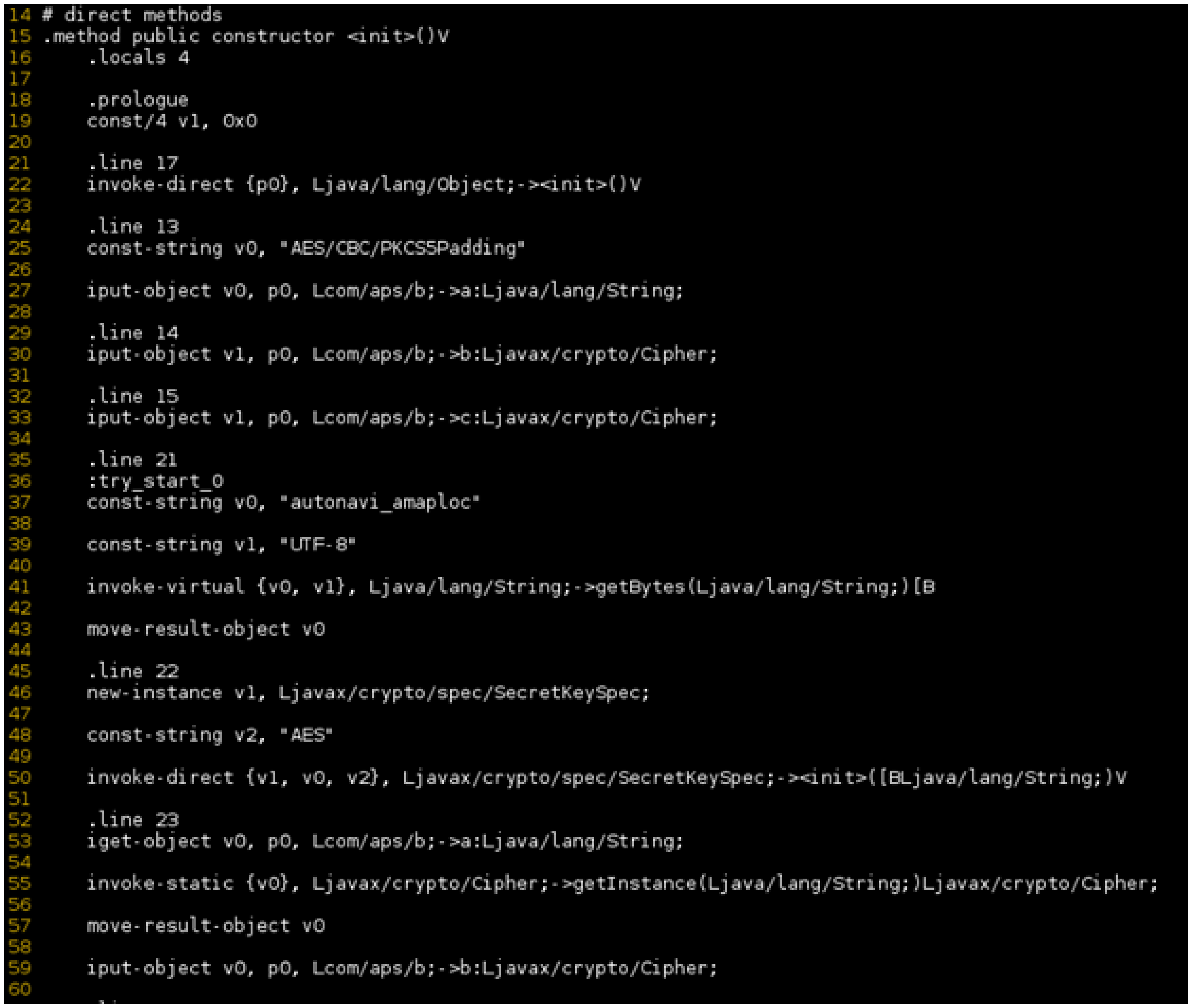
Since the com/aps/* directory serializes the ‘sreq’ structure in which we were interested, we next examined this directory to see which .smali files (code format used by Android) translated to which .java filenames:



```
jt@kali:~/awscode/apk_process/apk-prepped/uc-10.2.1_161.apk/uc-10.2.1_161/smali/com/aps$ ack-grep .source *
a$1.smali
3:.source "APS.java"

a$2.smali
3:.source "APS.java"

a$a.smali
3:.source "APS.java"

a.smali
3:.source "APS.java"

b.smali
3:.source "Aes.java"

c.smali
3:.source "AmapLoc.java"

d$a.smali
3:.source "Cache.java"

d.smali
3:.source "Cache.java"

e.smali
3:.source "Cgi.java"

f.smali
3:.source "Const.java"

g.smali
3:.source "Factory.java"

h.smali
3:.source "IAPS.java"

i.smali
```

Figure 8: Mapping of smali files in com/aps directory to java source code.

We examined the file Aes.java in searching for the component of the application that encrypted data in the 'sreq' structure (AES is a widely used form of encryption). The file showed that encryption was performed using symmetric AES/CBC encryption that used the hard-coded key ‘autonavi_amaploc’. The encryption process is shown in Figure 9:

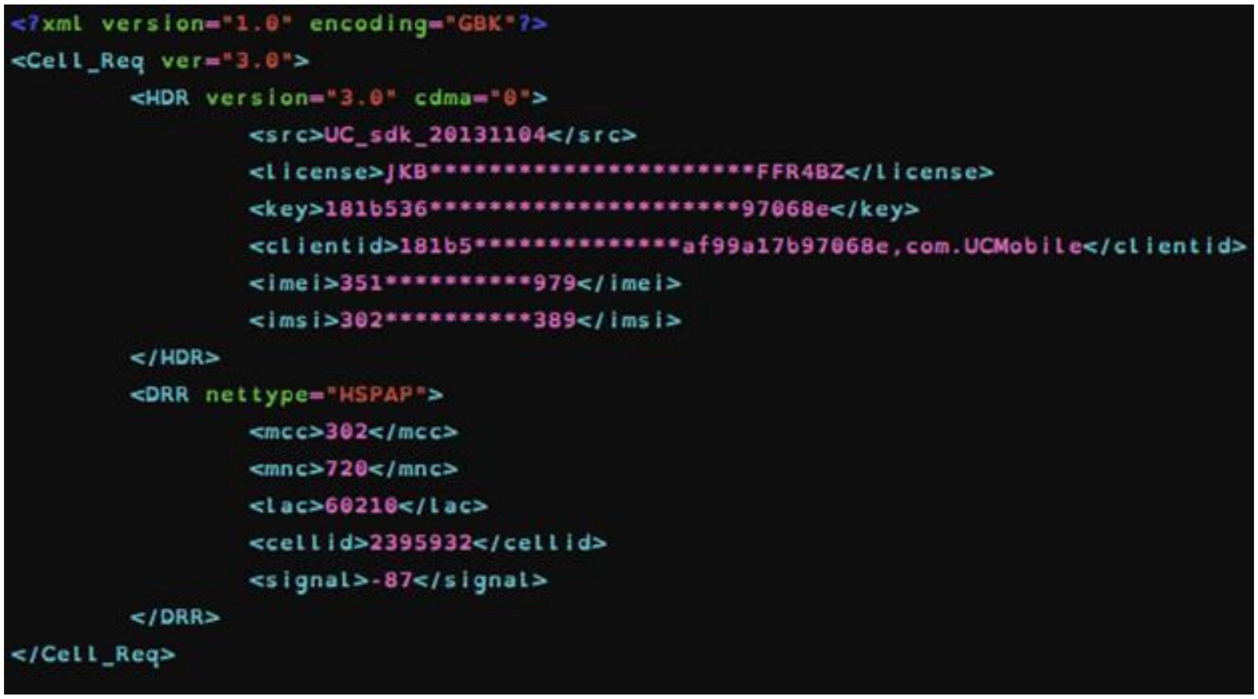


```
# direct methods
.method public constructor <init>()V
    .locals 4

    .prologue
    const/4 v1, 0x0

    .line 17
    invoke-direct {p0}, Ljava/lang/Object;-><init>()V

    .line 13
    const-string v0, "AES/CBC/PKCS5Padding"

    iput-object v0, p0, Lcom/aps/b;->a:Ljava/lang/String;

    .line 14
    iput-object v1, p0, Lcom/aps/b;->b:Ljavax/crypto/Cipher;

    .line 15
    iput-object v1, p0, Lcom/aps/b;->c:Ljavax/crypto/Cipher;

    .line 21
    :try_start_0
    const-string v0, "autonavi_amaploc"

    const-string v1, "UTF-8"

    invoke-virtual {v0, v1}, Ljava/lang/String;->getBytes(Ljava/lang/String;)[B

    move-result-object v0

    .line 22
    new-instance v1, Ljavax/crypto/spec/SecretKeySpec;

    const-string v2, "AES"

    invoke-direct {v1, v0, v2}, Ljavax/crypto/spec/SecretKeySpec;-><init>([BLjava/lang/String;)V

    .line 23
    iget-object v0, p0, Lcom/aps/b;->a:Ljava/lang/String;

    invoke-static {v0}, Ljavax/crypto/Cipher;->getInstance(Ljava/lang/String;)Ljavax/crypto/Cipher;

    move-result-object v0

    iput-object v0, p0, Lcom/aps/b;->b:Ljavax/crypto/Cipher;

```

Figure 9: Aes.java smali code showing AES/CBC encryption with a hard-coded key.

The use of symmetric encryption with a hard-coded key means that anyone who knows the key can decrypt UC Browser (Chinese) traffic in transit. Moreover, key holders can also retroactively decrypt any historical data that they have collected or obtained.

We used a standard AES decryption tool to decrypt the ‘sreq’ data structure in order to demonstrate that retroactive decryption was possible. After formatting the structure for readability, the data sent looked as follows:

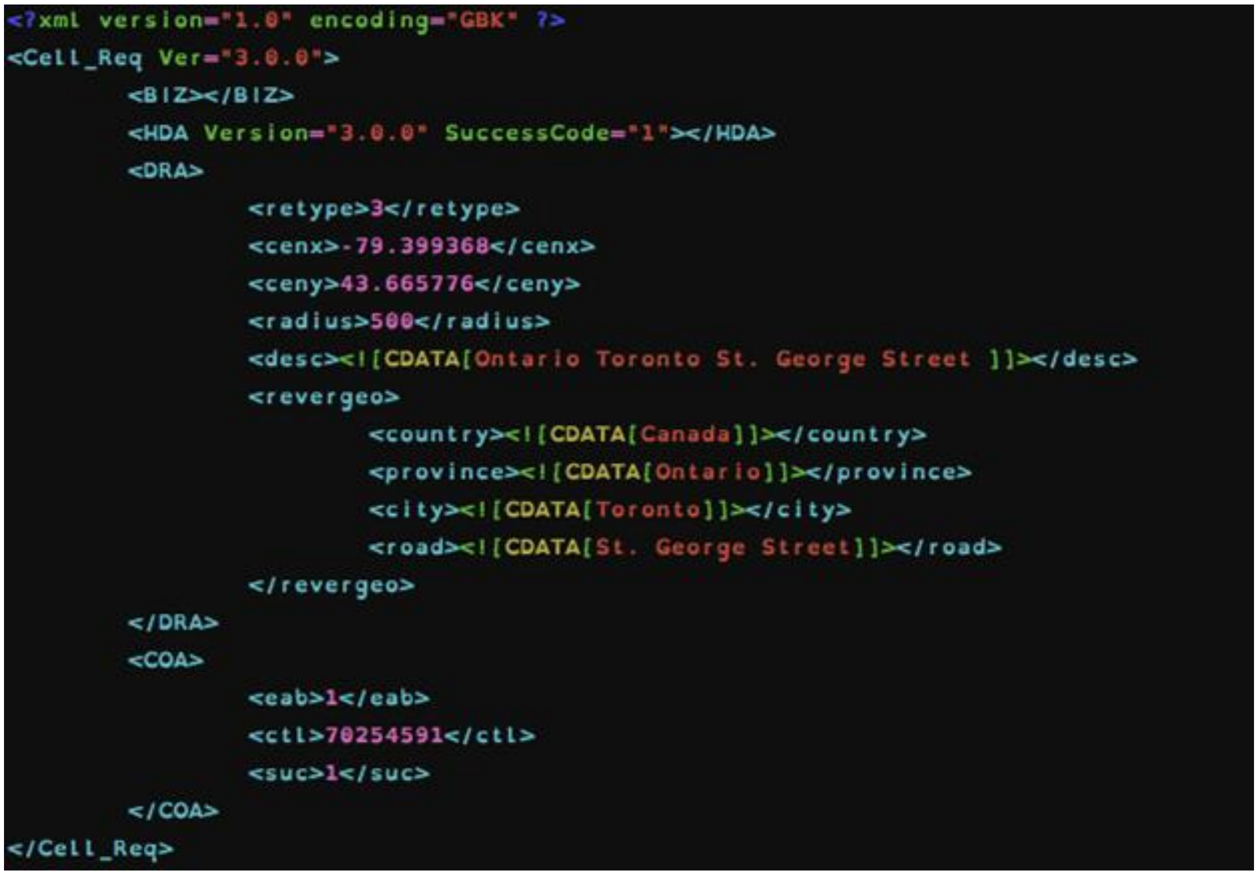


```
<?xml version="1.0" encoding="GBK"?>
<Cell_Req ver="3.0">
        <HDR version="3.0" cdma="0">
                <src>UC_sdk_20131104</src>
                <license>JKB**********************FFR4BZ</license>
                <key>181b536*********************97068e</key>
                <clientid>181b5**************af99a17b97068e,com.UCMobile</clientid>
                <imei>351**********979</imei>
                <imsi>302**********389</imsi>
        </HDR>
        <DRR nettype="HSPAP">
                <mcc>302</mcc>
                <mnc>720</mnc>
                <lac>60210</lac>
                <cellid>2395932</cellid>
                <signal>-87</signal>
        </DRR>
</Cell_Req>
```

Figure 10: Decrypted data sent via AMAP service. Personally identifiable data redacted with asterisks.

The data sent to apilocate.amap.com included a number of unique identifiers. The first set referred to the mobile device itself: the IMSI, IMEI, and unique user data related to this installation of UC Browser (Chinese). The second set referred to details of the cell tower to which the device is connected: the Mobile Country Code (MCC), Mobile Network Code (MNC), Location Area Code (LAC), Cell Tower ID, and the cell tower signal strength. In aggregate, these sets of information can be used to identify the cellular subscriber, the physical handset, and the physical location of the device.

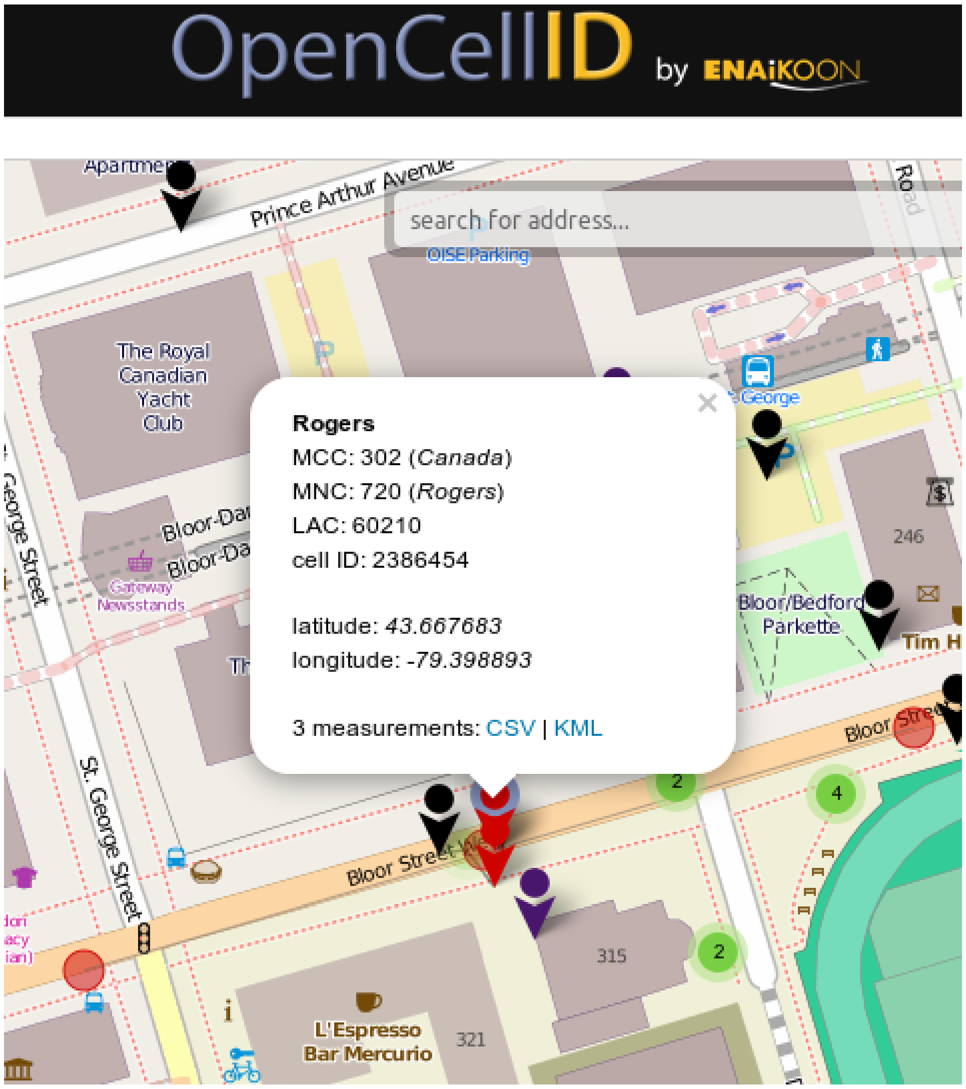
After transmitting this location information, the application received an unencrypted response that included the longitude/latitude of the user (the ‘cenx’ and ‘ceny’ values seen below), as well as the specific street name where the user was located, as shown below in Figure 11:



```
<?xml version="1.0" encoding="GBK" ?>
<Cell_Req Ver="3.0.0">
        <BIZ></BIZ>
        <HDA Version="3.0.0" SuccessCode="1"></HDA>
        <DRA>
                <retype>3</retype>
                <cenx>-79.399368</cenx>
                <ceny>43.665776</ceny>
                <radius>500</radius>
                <desc><![CDATA[Ontario Toronto St. George Street ]]></desc>
                <revergeo>
                        <country><![CDATA[Canada]]></country>
                        <province><![CDATA[Ontario]]></province>
                        <city><![CDATA[Toronto]]></city>
                        <road><![CDATA[St. George Street]]></road>
                </revergeo>
        </DRA>
        <COA>
                <eab>1</eab>
                <ctl>70254591</ctl>
                <suc>1</suc>
        </COA>
</Cell_Req>
```

Figure 11: Unencrypted response received by UC Browser (Chinese).

Such identifications are problematic because, using the information and the rate at which the application transmits it, we can determine a cell tower location with considerable accuracy and thus geolocate the person using the application. As an example, we were able to pinpoint the location of our lab where we conducted the testing, as shown in Figure 12:



Rogers
MCC: 302 (*Canada*)
MNC: 720 (*Rogers*)
LAC: 60210
cell ID: 2386454

latitude: *43.667683*
longitude: *-79.398893*

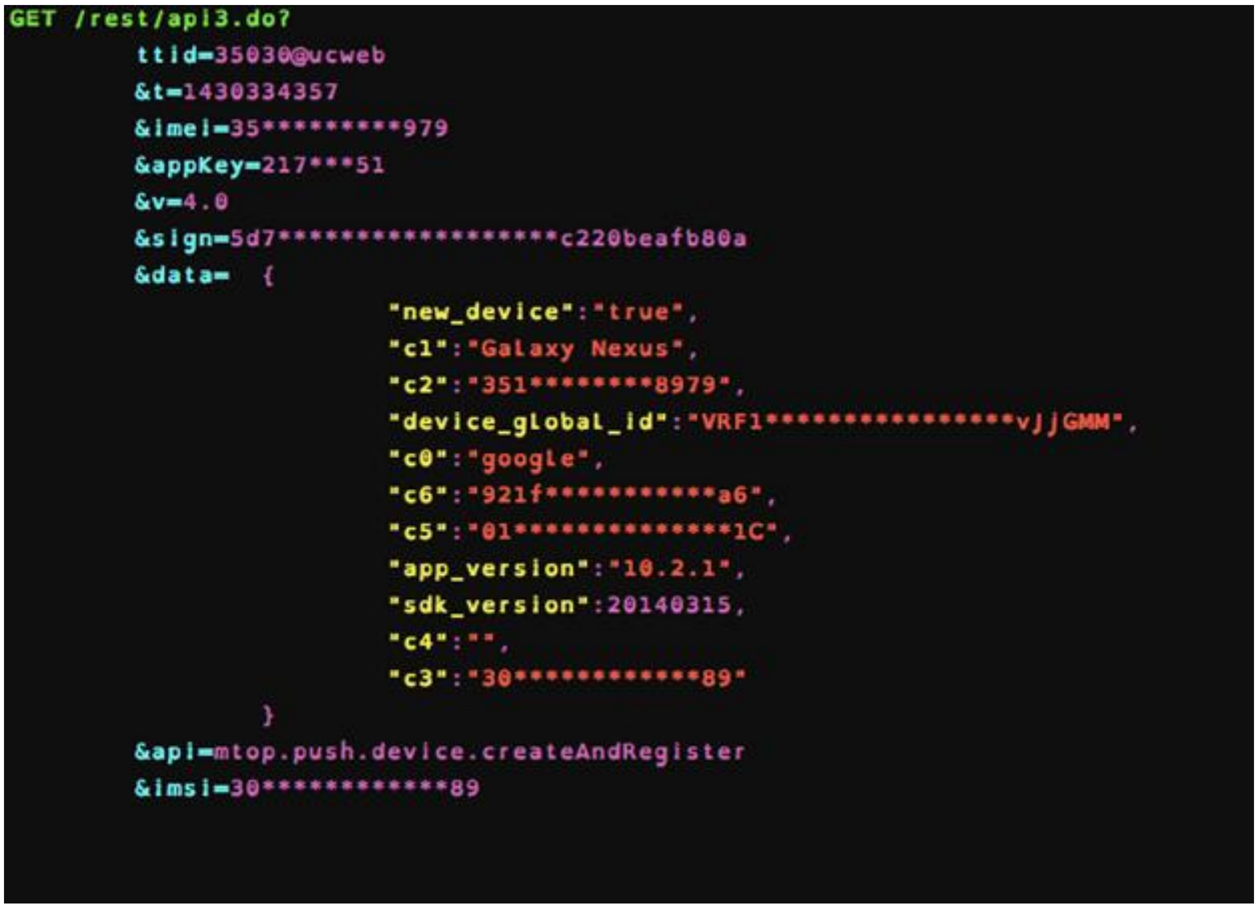
3 measurements: CSV | KML

Figure 12: Example of location identified from MCC, MNC, LAC, and CellID.

In summary, we strongly suspect that AMAP is the component of UC Browser (Chinese) that is responsible for transmitting the geolocation information. Our belief is based on the fact that the user-agent string ("AMAP Location SDK Android 1.0.5"), the location where the data is sent (apilocate.amap.com), and the text of the hard-coded key (''autonavi_amaploc") all reference AMAP. Given the apparent integration between AMAP and UCWeb we believe that it is likely that AMAP was incorporated into UC Browser (Chinese) to provide mapping and geolocation functionality.

Unencrypted Data Transfer To Umeng

As seen in Figure 5, UC Browser (Chinese) also periodically contacted utop.umengcloud.com and upoll.umengcloud.com when idle. Umeng is a mobile analytics service that is reportedly used by over 180,000 applications. It was purchased by Alibaba in 2013. The data structure sent, unencrypted, to utop.umengcloud.com hosts looked as follows:



```
GET /rest/api3.do?
        ttid=35030@ucweb
        &t=1430334357
        &imei=35*********979
        &appKey=217***51
        &v=4.0
        &sign=5d7******************c220beafb80a
        &data=  {
                        "new_device":"true",
                        "c1":"Galaxy Nexus",
                        "c2":"351*******8979",
                        "device_global_id":"VRF1****************vJjGMM",
                        "c0":"google",
                        "c6":"921f***********a6",
                        "c5":"01**************1C",
                        "app_version":"10.2.1",
                        "sdk_version":20140315,
                        "c4":"",
                        "c3":"30************89"
                }
        &api=mtop.push.device.createAndRegister
        &imsi=30************89
```

Figure 13: GET request made to utop.umeng.cloud with a cellular Internet connection showing that application sends IMSI, IMEI, Android ID, and build serial number without encryption. Personally identifiable data redacted with asterisks.

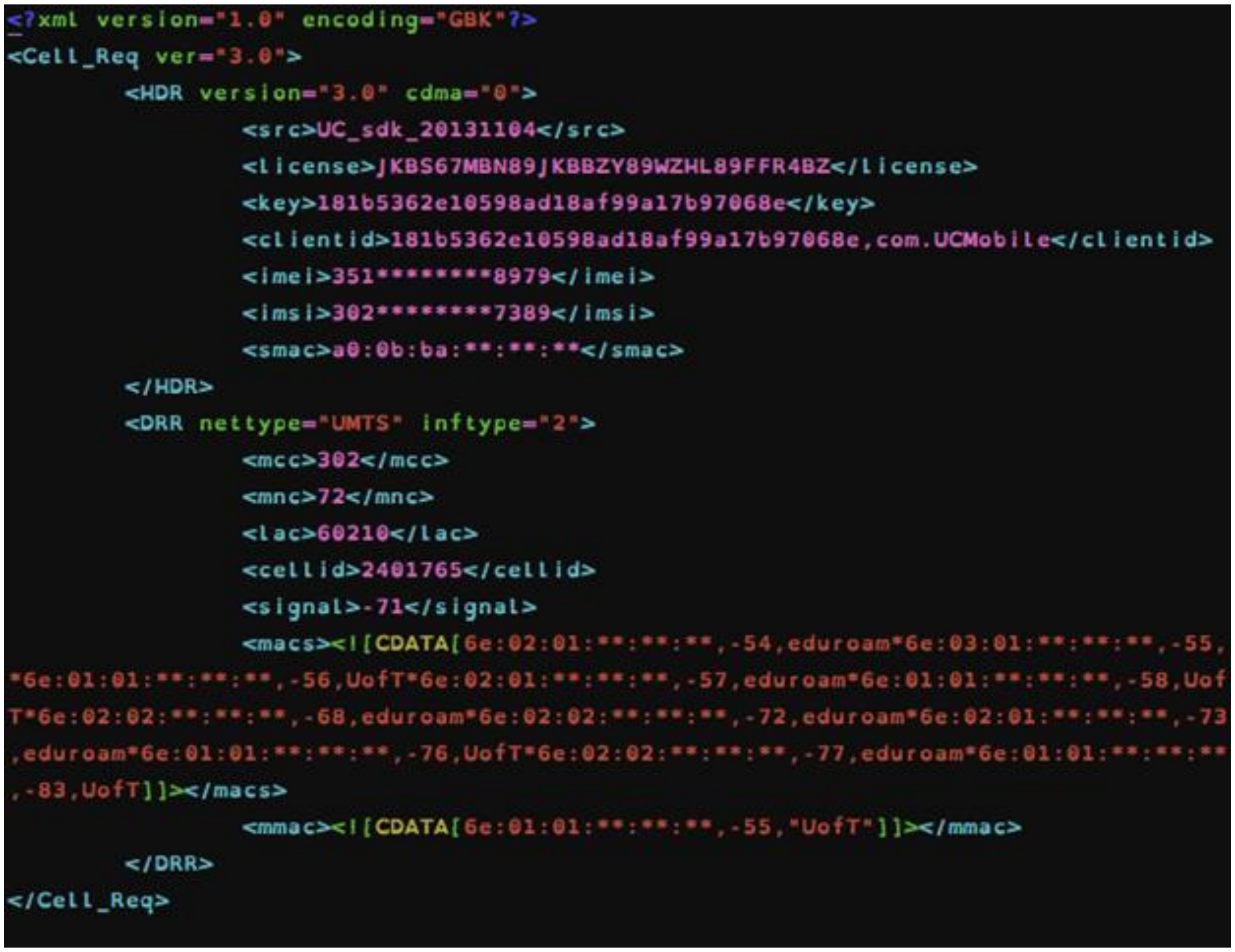
A number of unique personal identifiers are included in this data structure. This included the IMEI, IMSI, device Android ID (‘c6’), and the build serial number (‘c5’). Like AMAP, we believe it is likely that Umeng was incorporated into UC Browser (Chinese) to provide in-app analytics.

TESTING UC BROWSER’S WI-FI COMMUNICATION

Easily Decrypted Data Sent To AMAP

Upon starting the application and letting it idle for 270 seconds while connected to a Wi-Fi network, we found that UC Browser (Chinese) sent the same easily-decrypted user data seen in Cell only communication. However, in addition the application sent data about nearby Wi-Fi access points, including their MAC address.

These data elements are identified in Figure 14:

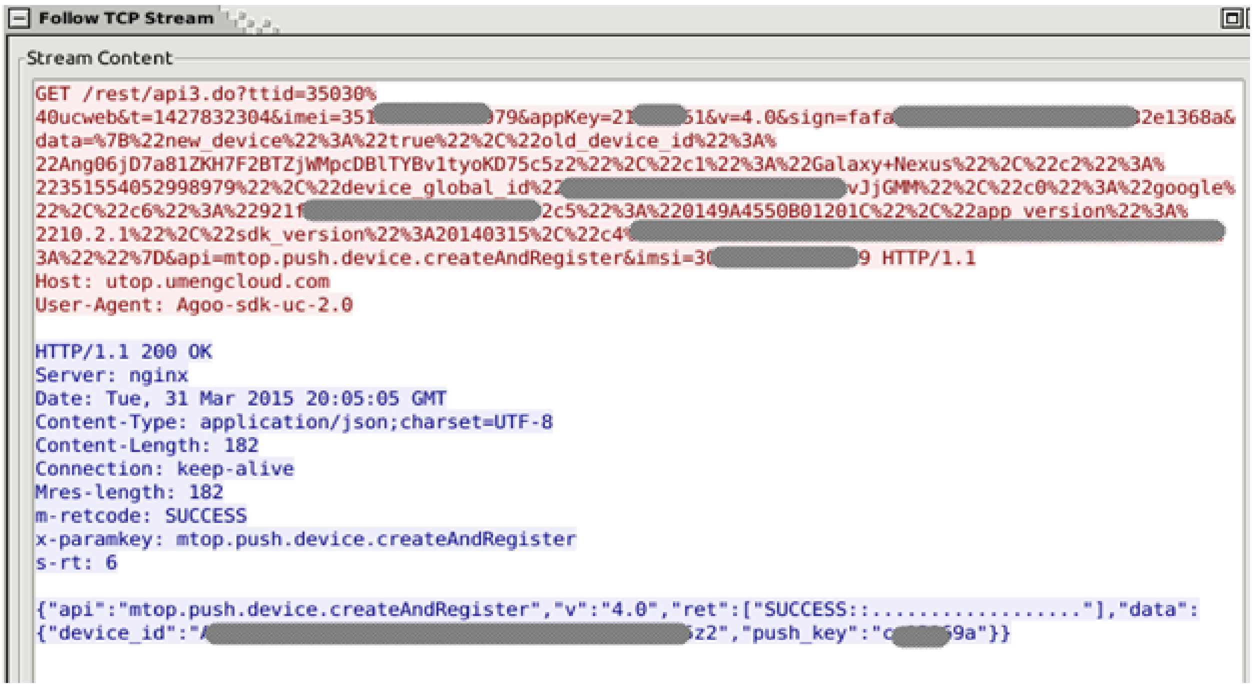


```
<?xml version="1.0" encoding="GBK"?>
<Cell_Req ver="3.0">
        <HDR version="3.0" cdma="0">
                <src>UC_sdk_20131104</src>
                <license>JKBS67MBN89JKBBZY89WZHL89FFR4BZ</license>
                <key>181b5362e10598ad18af99a17b97068e</key>
                <clientid>181b5362e10598ad18af99a17b97068e,com.UCMobile</clientid>
                <imei>351********8979</imei>
                <imsi>302********7389</imsi>
                <smac>a0:0b:ba:**:**:**</smac>
        </HDR>
        <DRR nettype="UMTS" inftype="2">
                <mcc>302</mcc>
                <mnc>72</mnc>
                <lac>60210</lac>
                <cellid>2401765</cellid>
                <signal>-71</signal>
                <macs><![CDATA[6e:02:01:**:**:**,-54,eduroam*6e:03:01:**:**:**,-55,
*6e:01:01:**:**:**,-56,UofT*6e:02:01:**:**:**,-57,eduroam*6e:01:01:**:**:**,-58,Uof
T*6e:02:02:**:**:**,-68,eduroam*6e:02:02:**:**:**,-72,eduroam*6e:02:01:**:**:**,-73
,eduroam*6e:01:01:**:**:**,-76,UofT*6e:02:02:**:**:**,-77,eduroam*6e:01:01:**:**:**
,-83,UofT]]></macs>
                <mmac><![CDATA[6e:01:01:**:**:**,-55,"UofT"]]></mmac>
        </DRR>
</Cell_Req>
```

Figure 14: Data sent by UC Browser (Chinese) to AMAP when Wi-Fi is enabled. Personally identifiable data redacted with asterisks.

Unencrypted Data Transfer To Umeng

When connected to a Wi-Fi network, personal user data is also sent unencrypted to Umeng. In addition to the data sent while connected to a cell network, the application also sent the device’s Wi-Fi MAC address. Figure 15 shows a sample of traffic sent to utop.umengcloud.com while connected to a Wi-Fi network:



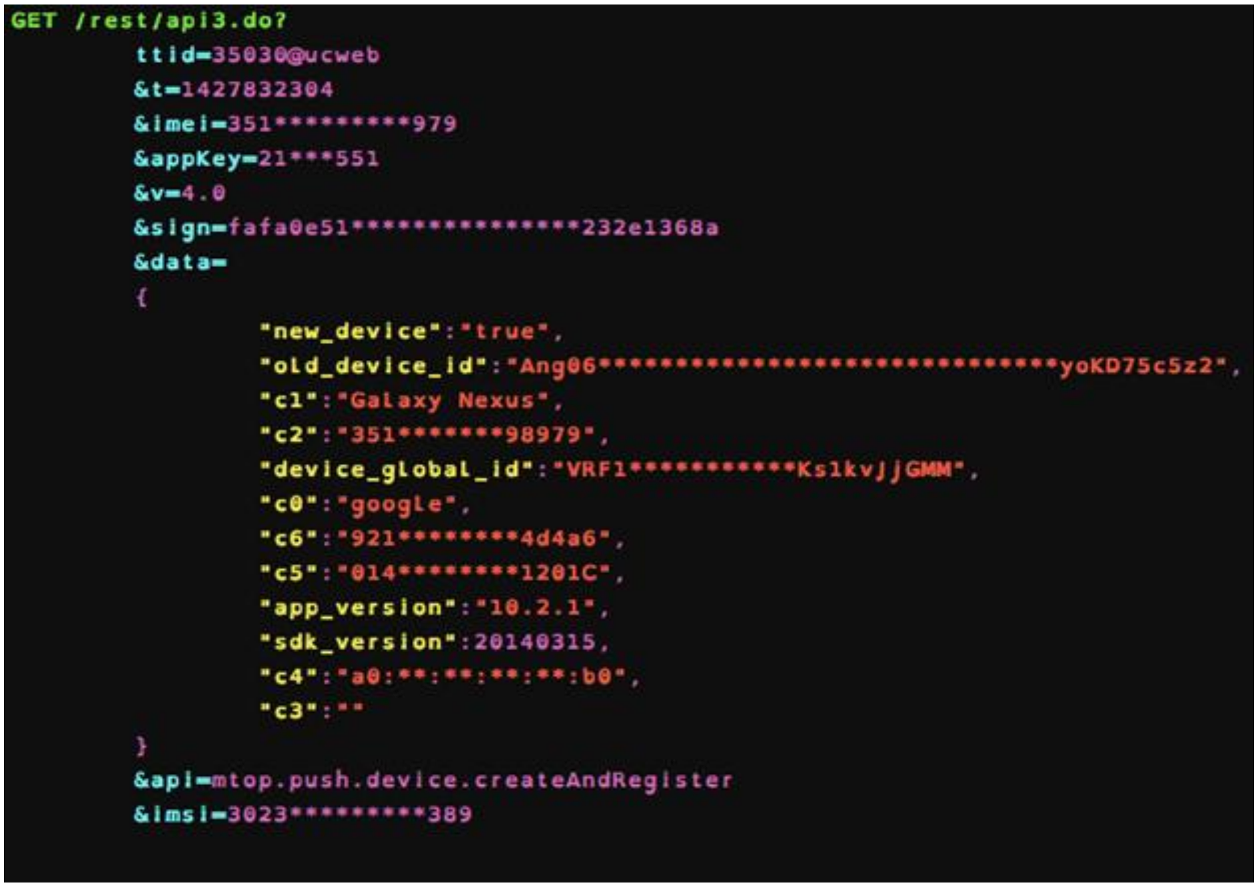
Follow TCP Stream

Stream Content

```
GET /rest/api3.do?ttid=35030%
40ucweb&t=1427832304&imei=351[redacted]79&appKey=21[redacted]51&v=4.0&sign=fafa[redacted]2e1368a&
data=%7B%22new_device%22%3A%22true%22%2C%22old_device_id%22%3A%
22Ang06jD7a81ZKH7F2BTZjWMpcDBlTYBv1tyoKD75c5z2%22%2C%22c1%22%3A%22Galaxy+Nexus%22%2C%22c2%22%3A%
22351554052998979%22%2C%22device_global_id%22[redacted]vJjGMM%22%2C%22c0%22%3A%22google%
22%2C%22c6%22%3A%22921f[redacted]2c5%22%3A%220149A4550B01201C%22%2C%22app_version%22%3A%
2210.2.1%22%2C%22sdk_version%22%3A20140315%2C%22c4%[redacted]
3A%22%22%7D&api=mtop.push.device.createAndRegister&imsi=30[redacted]9 HTTP/1.1
Host: utop.umengcloud.com
User-Agent: Agoo-sdk-uc-2.0

HTTP/1.1 200 OK
Server: nginx
Date: Tue, 31 Mar 2015 20:05:05 GMT
Content-Type: application/json;charset=UTF-8
Content-Length: 182
Connection: keep-alive
Mres-length: 182
m-retcode: SUCCESS
x-paramkey: mtop.push.device.createAndRegister
s-rt: 6

{"api":"mtop.push.device.createAndRegister","v":"4.0","ret":["SUCCESS::..................."],"data":
{"device_id":"A[redacted]5z2","push_key":"c[redacted]9a"}}
```

Figure 15: Traffic sent to utop.umengcloud.com during idle. Personally identifiable data redacted.

Re-formatting the above traffic for readability shows the following GET request to utop.umengcloud.com:



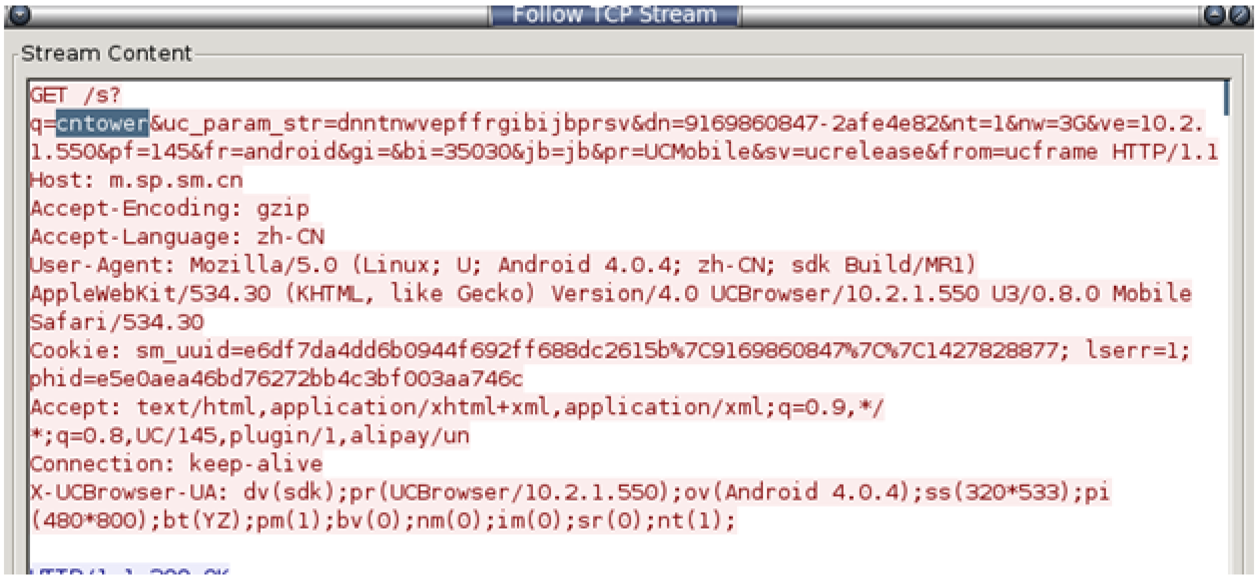
```
GET /rest/api3.do?
        ttid=35030@ucweb
        &t=1427832304
        &imei=351*********979
        &appKey=21***551
        &v=4.0
        &sign=fafa0e51***************232e1368a
        &data=
        {
                "new_device":"true",
                "old_device_id":"Ang06******************************yoKD75c5z2",
                "c1":"Galaxy Nexus",
                "c2":"351*******98979",
                "device_global_id":"VRF1***********KslkvJjGMM",
                "c0":"google",
                "c6":"921********4d4a6",
                "c5":"014********1201C",
                "app_version":"10.2.1",
                "sdk_version":20140315,
                "c4":"a0:**:**:**:**:b0",
                "c3":""
        }
        &api=mtop.push.device.createAndRegister
        &imsi=3023*********389
```

Figure 16: GET request made to utop.umengcloud.com with Wi-Fi enabled showing that IMSI, IMEI, Android ID, and device Wi-Fi MAC address are sent unencrypted. Personally identifiable data redacted with asterisks.

In summary, when connected to a Wi-Fi network, the IMSI, IMEI, Android ID, and Wi-Fi MAC address are sent unencrypted by UC Browser (Chinese) to umengcloud.com.

EXAMINING THE SEARCH FUNCTIONALITY OF UC BROWSER (CHINESE)

The Chinese version of UC Browser uses the mobile search engine Shenma, a joint venture between UCWeb and Alibaba. Search queries that are entered into the search bar were sent without encryption to http://m.sm.cn, as seen in Figure 17:

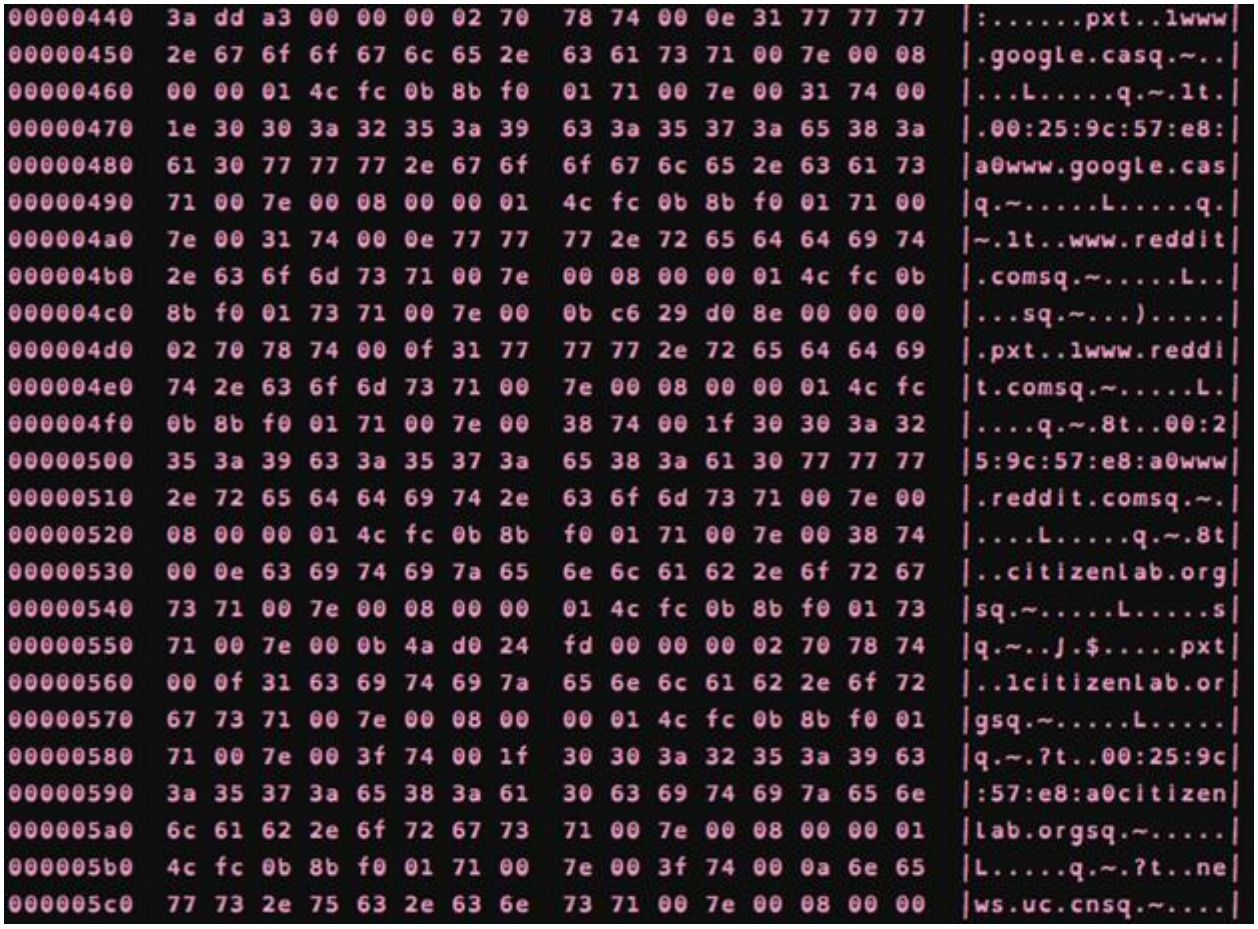


```
GET /s?
q=cntower&uc_param_str=dnntnwvepffrgibijbprsv&dn=9169860847-2afe4e82&nt=1&nw=3G&ve=10.2.
1.550&pf=145&fr=android&gi=&bi=35030&jb=jb&pr=UCMobile&sv=ucrelease&from=ucframe HTTP/1.1
Host: m.sp.sm.cn
Accept-Encoding: gzip
Accept-Language: zh-CN
User-Agent: Mozilla/5.0 (Linux; U; Android 4.0.4; zh-CN; sdk Build/MR1)
AppleWebKit/534.30 (KHTML, like Gecko) Version/4.0 UCBrowser/10.2.1.550 U3/0.8.0 Mobile
Safari/534.30
Cookie: sm_uuid=e6df7da4dd6b0944f692ff688dc2615b%7C9169860847%7C%7C1427828877; lserr=1;
phid=e5e0aea46bd76272bb4c3bf003aa746c
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/
*;q=0.8,UC/145,plugin/1,alipay/un
Connection: keep-alive
X-UCBrowser-UA: dv(sdk);pr(UCBrowser/10.2.1.550);ov(Android 4.0.4);ss(320*533);pi
(480*800);bt(YZ);pm(1);bv(0);nm(0);im(0);sr(0);nt(1);
```

Figure 17: Search for "cntower" using Shenma mobile search engine via UC Browser (Chinese) search bar.

INSECURE DATA DELETION IN UC BROWSER (CHINESE)

As part of our research we conducted tests to determine what, if any, personal user data was stored on the device. We began by selecting the option in the application allowing users to delete private information from the device, such as cookies and browser history. After selecting this option, we examined whether user data remained on the device by checking the cache directory for the application. While most user data was deleted, a record of the application's DNS lookups remained, as shown in Figure 18:



```
00000440  3a dd a3 00 00 00 02 70  78 74 00 0e 31 77 77 77  |:......pxt..1www|
00000450  2e 67 6f 6f 67 6c 65 2e  63 61 73 71 00 7e 00 08  |.google.casq.~..|
00000460  00 00 01 4c fc 0b 8b f0  01 71 00 7e 00 31 74 00  |...L.....q.~.1t.|
00000470  1e 30 30 3a 32 35 3a 39  63 3a 35 37 3a 65 38 3a  |.00:25:9c:57:e8:|
00000480  61 30 77 77 77 2e 67 6f  6f 67 6c 65 2e 63 61 73  |a0www.google.cas|
00000490  71 00 7e 00 08 00 00 01  4c fc 0b 8b f0 01 71 00  |q.~.....L.....q.|
000004a0  7e 00 31 74 00 0e 77 77  77 2e 72 65 64 64 69 74  |~.1t..www.reddit|
000004b0  2e 63 6f 6d 73 71 00 7e  00 08 00 00 01 4c fc 0b  |.comsq.~.....L..|
000004c0  8b f0 01 73 71 00 7e 00  0b c6 29 d0 8e 00 00 00  |...sq.~...).....|
000004d0  02 70 78 74 00 0f 31 77  77 77 2e 72 65 64 64 69  |.pxt..1www.reddi|
000004e0  74 2e 63 6f 6d 73 71 00  7e 00 08 00 00 01 4c fc  |t.comsq.~.....L.|
000004f0  0b 8b f0 01 71 00 7e 00  38 74 00 1f 30 30 3a 32  |....q.~.8t..00:2|
00000500  35 3a 39 63 3a 35 37 3a  65 38 3a 61 30 77 77 77  |5:9c:57:e8:a0www|
00000510  2e 72 65 64 64 69 74 2e  63 6f 6d 73 71 00 7e 00  |.reddit.comsq.~.|
00000520  08 00 00 01 4c fc 0b 8b  f0 01 71 00 7e 00 38 74  |....L.....q.~.8t|
00000530  00 0e 63 69 74 69 7a 65  6e 6c 61 62 2e 6f 72 67  |..citizenlab.org|
00000540  73 71 00 7e 00 08 00 00  01 4c fc 0b 8b f0 01 73  |sq.~.....L.....s|
00000550  71 00 7e 00 0b 4a d0 24  fd 00 00 00 02 70 78 74  |q.~..J.$.....pxt|
00000560  00 0f 31 63 69 74 69 7a  65 6e 6c 61 62 2e 6f 72  |..1citizenlab.or|
00000570  67 73 71 00 7e 00 08 00  00 01 4c fc 0b 8b f0 01  |gsq.~.....L.....|
00000580  71 00 7e 00 3f 74 00 1f  30 30 3a 32 35 3a 39 63  |q.~.?t..00:25:9c|
00000590  3a 35 37 3a 65 38 3a 61  30 63 69 74 69 7a 65 6e  |:57:e8:a0citizen|
000005a0  6c 61 62 2e 6f 72 67 73  71 00 7e 00 08 00 00 01  |lab.orgsq.~.....|
000005b0  4c fc 0b 8b f0 01 71 00  7e 00 3f 74 00 0a 6e 65  |L.....q.~.?t..ne|
000005c0  77 73 2e 75 63 2e 63 6e  73 71 00 7e 00 08 00 00  |ws.uc.cnsq.~....|
```

Figure 18: DNS records found in cache after private information was deleted.

This DNS data was stored in the cache as a serialized LinkedHashMap, and persisted even after all other data in the cache had been cleared using the application's feature to clear private browsing data. There was sufficient plaintext remaining that the records could be read using a simple text editor. In other words, even if a user attempts to clear browsing records their personal data remains available for scrutiny and can be trivially accessed.

3.2 UC BROWSER (ENGLISH) LANGUAGE VERSION 10.4.1.565

We next conducted tests of the English language edition of UC Browser version 10.4.1.565. This version was downloaded as an APK directly from the UCWeb English-language website. Our test results are summarized in Table 3:

Table 3: Summary of test results for UC Browser (English).

Test	Both	Results
Idle test	Cell only & Cell + Wi-Fi	• No issues identified
Search	Cell only & Cell + Wi-Fi	• Search queries sent through the search bar are sent unencrypted to Yahoo! India • Search queries sent through the address bar are sent unencrypted to Google
Data	Cell only & Cell +	• No issues identified

storage	Wi-Fi	

Idle Test

We performed the same idle test described previously with the English language version: the application was launched, left idle for 270 seconds, and all traffic sent to and from the device was collected. We first performed the test by connecting the mobile device to the Internet using a cellular connection and, second, by connecting the device to the Internet using a Wi-Fi network.

Our analysis showed that UC Browser (English) did not send and receive traffic through the AMAP or Umeng component as in the Chinese language version. We were not able to identify any easily decrypted traffic sent in the English language version.

Search

UC Browser (English) has two methods for performing web searches. The first method is by tapping the "Search" button shown in the upper-right hand corner of the application. The second method is by entering a search term in the address bar to the left of the search button. Both of these methods can be seen in the following screenshot:

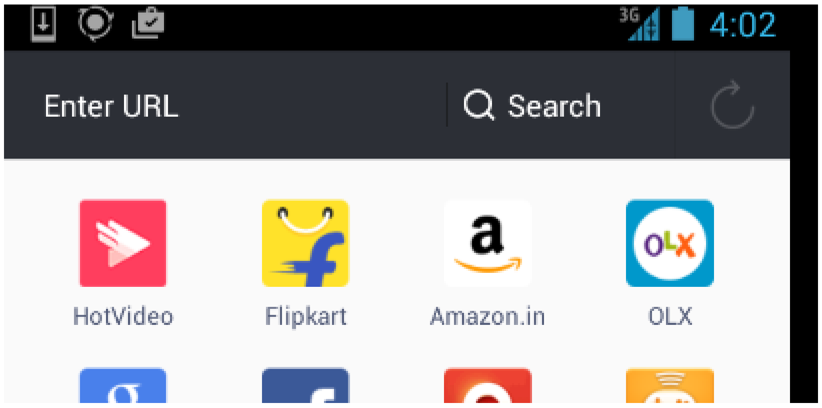
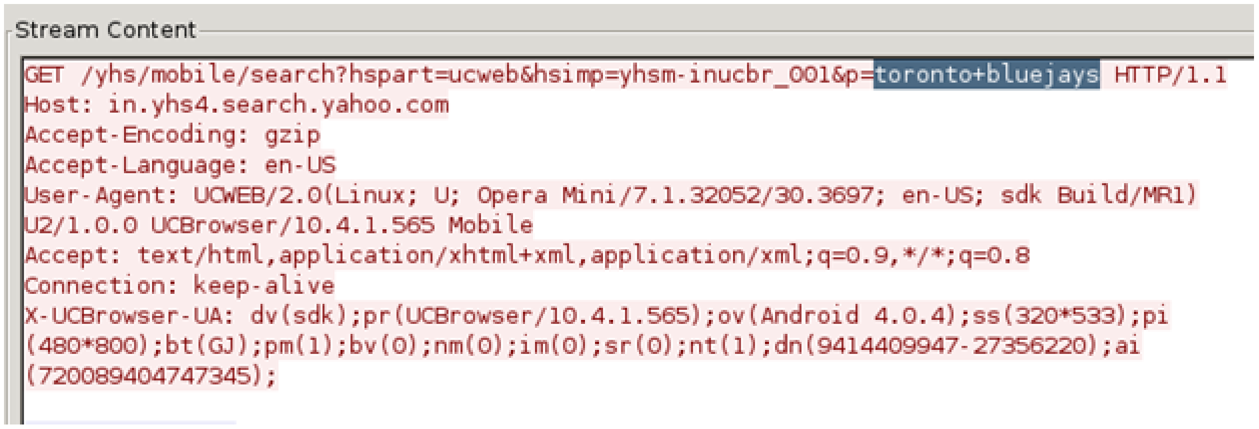


Figure 19: Screenshot of search features of UC Browser (English). Both the "Enter URL" and "Search" fields can be used to perform a web search.

We performed searches through both methods and observed the traffic sent and received by UC Browser (English). Performing the search through the search bar sent data unencrypted to Yahoo! India search, as shown in this packet capture:



```
GET /yhs/mobile/search?hspart=ucweb&hsimp=yhsm-inucbr_001&p=toronto+bluejays HTTP/1.1
Host: in.yhs4.search.yahoo.com
Accept-Encoding: gzip
Accept-Language: en-US
User-Agent: UCWEB/2.0(Linux; U; Opera Mini/7.1.32052/30.3697; en-US; sdk Build/MR1)
U2/1.0.0 UCBrowser/10.4.1.565 Mobile
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Connection: keep-alive
X-UCBrowser-UA: dv(sdk);pr(UCBrowser/10.4.1.565);ov(Android 4.0.4);ss(320*533);pi
(480*800);bt(GJ);pm(1);bv(0);nm(0);im(0);sr(0);nt(1);dn(9414409947-27356220);ai
(720089404747345);
```

Figure 20: Packet capture of search for 'toronto bluejays' (highlighted in blue) performed using search bar in UC Browser (English). Search query is sent unencrypted to Yahoo! India.

In addition, the results of such a search are displayed with a green checkmark at the far left of the search bar:

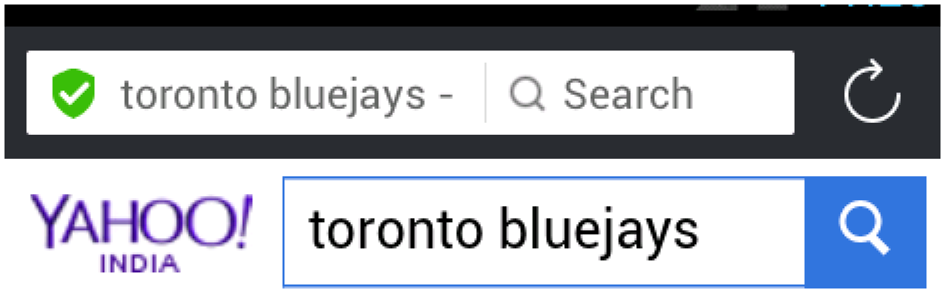
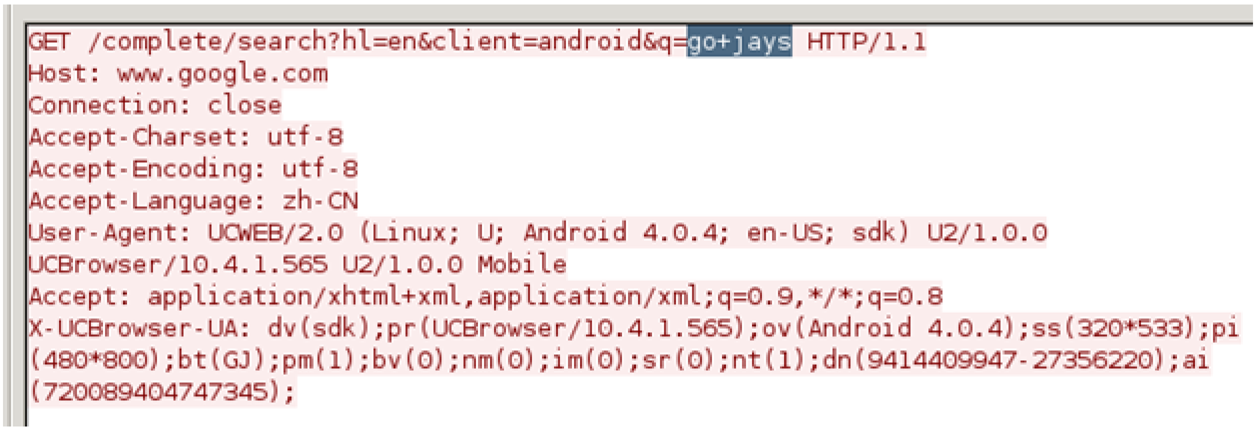


Figure 21: Search results for an unencrypted search to Yahoo! India with a green checkmark.

The use of this green checkmark could lead to some confusion with users, as most web browsers use a green icon to the left of the search bar to reflect an encrypted connection. UC Browser (English) displays a green padlock to the left of the browser bar if the HTTPS-encrypted version of a site is opened manually in a browser.

Next, we performed a search by entering a search term in the address bar of UC Browser (English). This search was sent unencrypted to Google search:



```
GET /complete/search?hl=en&client=android&q=go+jays HTTP/1.1
Host: www.google.com
Connection: close
Accept-Charset: utf-8
Accept-Encoding: utf-8
Accept-Language: zh-CN
User-Agent: UCWEB/2.0 (Linux; U; Android 4.0.4; en-US; sdk) U2/1.0.0
UCBrowser/10.4.1.565 U2/1.0.0 Mobile
Accept: application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
X-UCBrowser-UA: dv(sdk);pr(UCBrowser/10.4.1.565);ov(Android 4.0.4);ss(320*533);pi
(480*800);bt(GJ);pm(1);bv(0);nm(0);im(0);sr(0);nt(1);dn(9414409947-27356220);ai
(720089404747345);
```

Figure 22: Packet capture of search for "go jays" (highlighted in blue) performed using the address bar of UC Browser (English).

In summary, both methods of performing a search in UC Browser (English) sent the search query unencrypted to either Yahoo! India or Google. The standard web version of each of these search engines uses HTTPS encryption by default.

DATA STORAGE

Finally, we analyzed how UC Browser (English) stored personal user data on the device. The English language version, unlike the Chinese language version, did not store DNS lookup data as part of the private browsing data. Further, using the option within the application to delete private browsing data did delete all such data.

SECTION 4 - UC BROWSER LEAKS SENSITIVE USER DATA

Our analysis shows that both versions of UC Browser leak information to third parties, but that privacy and security concerns for the Chinese language UC Browser are much greater. The Chinese UC Browser version we tested ("UC Browser (Chinese)" in this report) leaks a significant amount of personally identifiable information, raising major security and privacy concerns. The leakage of the IMSI, IMEI, and geolocational information can identify a cellular subscriber, the device that they are using, and their specific location. As a result of the weak encryption used by UC Browser, any party with access to the data traffic -- either real-time or historical -- can link specific devices to specific places at specific times. And if the decrypting party has a large volume of data they can track subscribers vis-a-vis their mobile devices as they move around the world. Just by installing and opening UC Browser (Chinese), users unwittingly expose a significant number of personal identifiers and location information to numerous third parties. Although users must agree to grant the application permission to access personal identifiers and location data, it is not made clear to the user how this data will be shared. This exposed information includes:

- Device info sent unencrypted: IMSI, IMEI, Android ID, and Wi-Fi MAC address
- Search queries sent unencrypted
- Location data received unencrypted: longitude/latitude and street name
- Device and location sent with breakable encryption: IMSI, IMEI, MCC, MNC, LAC, CellId, nearby cellular towers and Wi-Fi access points

In many political jurisdictions (including China and India) it is common for authorities to require telecommunications companies, cellular providers, and Internet cafes to share the data they collect with security agencies as a condition of obtaining an operating license. By leaking a large volume of fine-grained data points to multiple network operators, the UC Browser app is increasing the risks to its users that such data may be used against them by authorities, criminals, or other third parties.

The data leakages we outline are particularly problematic for individuals who use their devices to engage in sensitive communications or for whom disclosing their physical location could place them at increased risk. Similarly, individuals concerned with protecting sensitive activities related to their work while traveling or communicating should be concerned about the potential for industrial espionage.

While we concluded that UC Browser (English) leaks considerably less identifying information, users might be surprised to realize that, despite the presence of an icon suggesting security in one of the search bars, their search terms are transmitted without encryption to Google and Yahoo! India servers.

ON THE ISSUE OF MOBILE SECURITY AND PRIVACY

Ultimately, the concerns identified here with respect to UC Browser demonstrate the larger challenges of ensuring user security and privacy within the burgeoning market for mobile applications. The [TK link to primer]mobile ecosystem[/link] is complex and multi-layered, involving large volumes of personally identifiable information that are transmitted across networks, devices, operating systems, and applications owned and operated by numerous private companies across many political and regulatory jurisdictions. Such a complex system underscores the importance of systematically evaluating the privacy and security of mobile communications as they become integral to the everyday lives of individuals and communities worldwide.

WOULD ENCRYPTION SOLVE UC BROWSER’S PROBLEMS?

We have highlighted the lack of encryption for personally identifiable data as a key reason for concern over UC Browser. Encrypting data that is this sensitive certainly represents an industry best practice, and it is unclear why only the English version seems to implement encryption consistently. Modifying the Chinese version to match the encryption used in its English counterpart could be an important step in increasing user security, as would encrypting queries to Google and Yahoo! India in the English version.

However, even if all data were strongly encrypted, this step would simply simply make it more difficult for unauthorized parties to read the contents of data transmissions. Encrypting sensitive user data can limit the number of actors who can access the data but does not prevent the inappropriate collection, retention, and analysis of the data by application developers and their commercial partners. Put bluntly: increases in transport security do not necessarily improve corporate data handling practices.

The core advantage of better encrypting data traffic is to engage, and hopefully make more transparent, the processes that government agencies and other third parties must engage in to access information that is collected, retained, and processed by application developers. In many jurisdictions, authorities will first have to obtain a court order before lawfully accessing the application developers' data. Nevertheless, when data is held in jurisdictions where this process is not enshrined, or where there are strong incentives to share the data, encryption alone does not solve the problem.

ACKNOWLEDGMENTS

This project was supported by the John D. and Catherine T. MacArthur Foundation.