Chinese Police & CloudPets

DeepSec
November 28-29, 2019 – Vienna, Austria
Presented by : Abraham Aranguren

> admin@7asecurity.com

> @7asecurity

> @7a_

> @owtfp [OWASP OWTF - owtf.org]

+ 7asecurity.com

SECURITY

Who am I?

- ★ Director at **7ASecurity**, public reports, presentations, etc. here: 7asecurity.com/publications
- ★ Former Team Lead & Penetration Tester at **Cure53** and **Version 1**
- ★ Co-Author of hands-on 7ASecurity courses:
  - ○ **Pwn & Fix JS apps, shells, injections and fun!** a Node.js & Electron course
  - ○ **Hacking Android, iOS and IoT** a Mobile App Security course
- ★ Author of **Practical Web Defense**, a hands-on attack & defense course: www.elearnsecurity.com/PWD
- ★ Founder and leader of **OWASP OWTF**, and **OWASP flagship project**: owtf.org
- ★ Some presentations: www.slideshare.net/abrahamaranguren/presentations
- ★ Some **sec certs**: CISSP, OSCP, GWEB, OSWP, CPTS, CEH, MCSE: Security, MCSA: Security, Security+
- ★ Some **dev certs**: ZCE PHP 5, ZCE PHP 4, Oracle PL/SQL Developer Certified Associate, MySQL 5 CMDev, MCTS SQL Server 2005

Public Mobile Pentest Reports - I

Smart Sheriff mobile app mandated by the South Korean government:

Public Pentest Reports:

→ Smart Sheriff: Round #1 - https://7asecurity.com/reports/pentest-report_smartsheriff.pdf
→ Smart Sheriff: Round #2 - https://7asecurity.com/reports/pentest-report_smartsheriff-2.pdf

Presentation:“Smart Sheriff, Dumb Idea, the wild west of government assisted parenting”
Slides:https://www.slideshare.net/abrahamaranguren/smart-sheriff-dumb-idea-the-wild-west-of-government-assisted-parenting
Video: https://www.youtube.com/watch?v=AbGX67CuVBQ

Chinese Police Apps Pentest Reports:

→ “Study the Great Nation” 09.2019 https://7asecurity.com/reports/analysis-report_sgn.pdf
→ "BXAQ" (OTF) 03.2019 - https://7asecurity.com/reports/analysis-report_bxaq.pdf
→ "IJOP" (HRW) 12.2018 - https://7asecurity.com/reports/analysis-report_ijop.pdf

Public Mobile Pentest Reports - II

Other reports:

→ Exodus iOS Mobile App - https://7asecurity.com/reports/pentest-report_exodus.pdf

→ imToken Wallet - https://7asecurity.com/reports/pentest-report_imtoken.pdf

→ Whistler Apps - https://7asecurity.com/reports/pentest-report_whistler.pdf

→ Psiphon - https://7asecurity.com/reports/pentest-report_psiphon.pdf

→ Briar - https://7asecurity.com/reports/pentest-report_briar.pdf

→ Padlock - https://7asecurity.com/reports/pentest-report_padlock.pdf

→ Peerio - https://7asecurity.com/reports/pentest-report_peerio.pdf

→ OpenKeyChain - https://7asecurity.com/reports/pentest-report_openkeychain.pdf

→ F-Droid / Baazar - https://7asecurity.com/reports/pentest-report_fdroid.pdf

→ Onion Browser - https://7asecurity.com/reports/pentest-report_onion-browser.pdf

More here:

https://7asecurity.com/publications

Agenda

3 different security audits with interesting backgrounds:

1. **CloudPets:**
   - Preliminary work & epic track record
   - What we found
   - What happened afterwards
2. “**IJOP**” Chinese Police app:
   - Police enter data manually, fill out forms
3. “**BXAQ**” Chinese Police app:
   - Police install an app that grabs data from a phone

"**BXAQ**" and "**IJOP**" are related to surveillance of ethnic minorities, but in different ways.

PART 1: CloudPets

What are CloudPets?

https://www.youtube.com/watch?v=11gvtRg3_V8

How do CloudPets work?

https://www.youtube.com/watch?v=kgyRvO0sgcE

CloudPets Summary - I

Intended usage:

→ **Parent** (far from home) **sends messages** to children using a **mobile app**

→ **Children** receive these messages on the **Soft Toy**

→ **Children** can send messages via the **Soft Toy**

→ **Parent** receives messages on the **mobile app**

The Toys:

→ Use Bluetooth LE → To communicate with the mobile app

→ Have a Microphone

→ Have a speaker

CloudPets Summary - II

Mobile app on **parent phone = Away** from the **toy**

→ **Sends/Receives messages** to/from:

CloudPets servers and Amazon S3

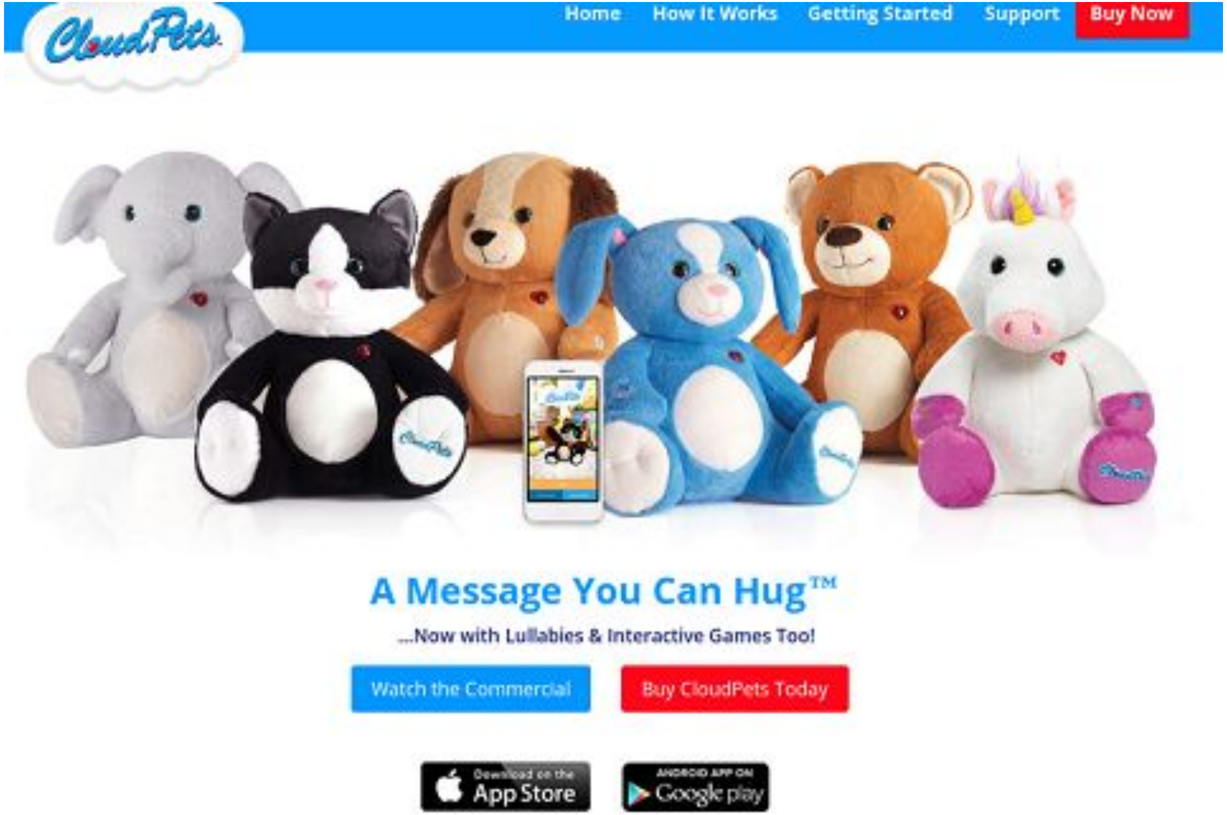
Mobile app on **children device = Close** to the **toy**

→ **Sends/Receives messages** to/from:

CloudPets servers and Amazon S3

→ **Uploads/Downloads messages** to/from Toy via: **Bluetooth LE**

What could possibly go wrong? Any ideas?



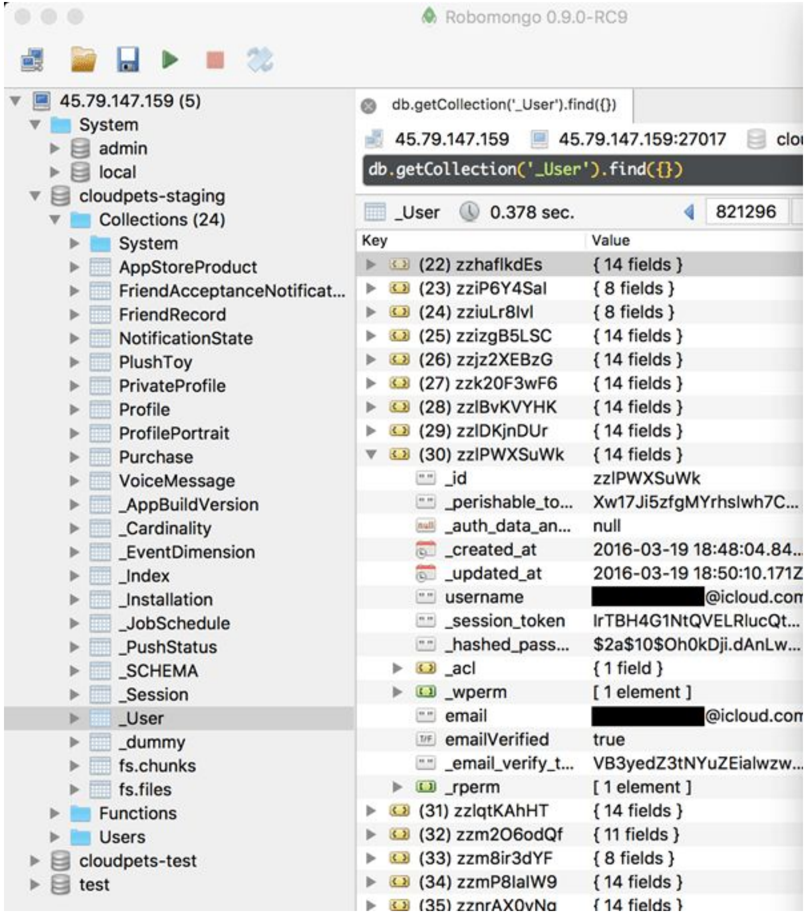
Previous Work: #1 - Mongo DB without auth

Full access to **all messages** ever sent between parents and children!

Summary:

→ **Mongo DB exposed to the internet without authentication**
→ **Unauthorized** parties downloaded **the database**
→ **3 Ransom requests**
→ **Indexed by Shodan**
→ **821k** user **records** at risk.
→ Spiral Toys (CloudPets's company) claimed to **never** have found evidence of any breach…..

Previous Work: #1 - Mongo DB without auth



45.79.147.159 (5)
- System
  - admin
  - local
- cloudpets-staging
  - Collections (24)
    - System
    - AppStoreProduct
    - FriendAcceptanceNotificat...
    - FriendRecord
    - NotificationState
    - PlushToy
    - PrivateProfile
    - Profile
    - ProfilePortrait
    - Purchase
    - VoiceMessage
    - _AppBuildVersion
    - _Cardinality
    - _EventDimension
    - _Index
    - _Installation
    - _JobSchedule
    - _PushStatus
    - _SCHEMA
    - _Session
    - _User
    - _dummy
    - fs.chunks
    - fs.files
  - Functions
  - Users
- cloudpets-test
- test

db.getCollection('_User').find({})

45.79.147.159 45.79.147.159:27017 clo

```
db.getCollection('_User').find({})
```

_User 0.378 sec. 821296

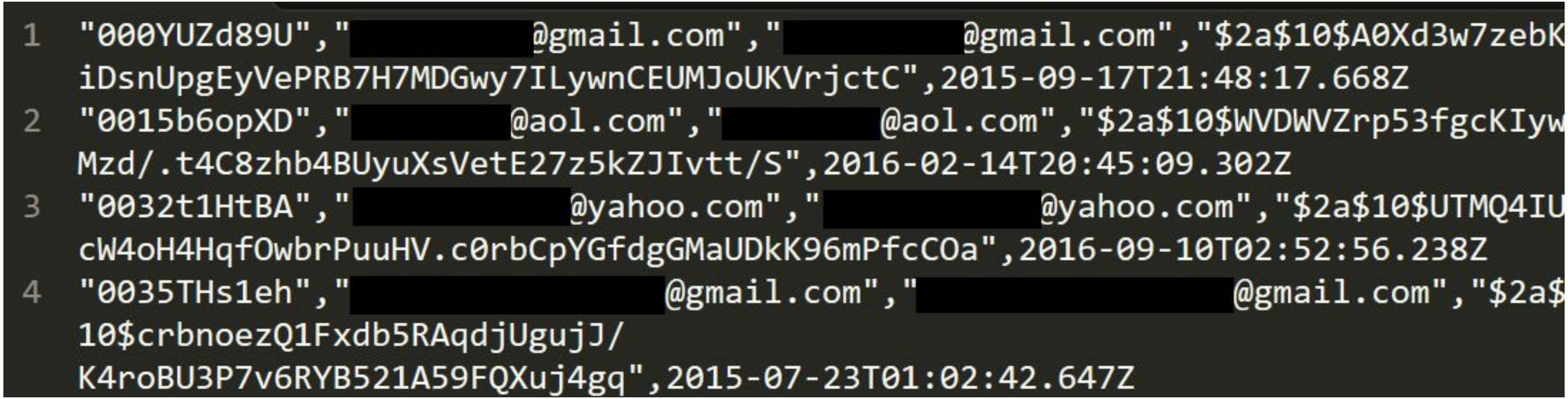
Key	Value
(22) zzhaflkdEs	{ 14 fields }
(23) zziP6Y4Sal	{ 8 fields }
(24) zziuLr8lvl	{ 8 fields }
(25) zzizgB5LSC	{ 14 fields }
(26) zzjz2XEBzG	{ 14 fields }
(27) zzk20F3wF6	{ 14 fields }
(28) zzlBvKVYHK	{ 14 fields }
(29) zzlDKjnDUr	{ 14 fields }
(30) zzIPWXSuWk	{ 14 fields }
_id	zzIPWXSuWk
_perishable_to...	Xw17Ji5zfgMYrhslwh7C...
_auth_data_an...	null
_created_at	2016-03-19 18:48:04.84...
_updated_at	2016-03-19 18:50:10.171Z
username	[redacted]@icloud.com
_session_token	IrTBH4G1NtQVELRlucQt...
_hashed_pass...	$2a$10$Oh0kDji.dAnLw...
_acl	{ 1 field }
_wperm	[1 element]
email	[redacted]@icloud.com
emailVerified	true
_email_verify_t...	VB3yedZ3tNYuZEialwzw...
_rperm	[1 element]
(31) zzlqtKAhHT	{ 14 fields }
(32) zzm2O6odQf	{ 11 fields }
(33) zzm8ir3dYF	{ 8 fields }
(34) zzmP8lalW9	{ 14 fields }
(35) zznrAXOvNg	{ 14 fields }

+ 7asecurity.com

SECURITY

Previous Work: #1 - Mongo DB without auth

Password hashes, emails, links to all voice recordings from children and parents, etc.



```
"000YUZd89U","[redacted]@gmail.com","[redacted]@gmail.com","$2a$10$A0Xd3w7zebK
iDsnUpgEyVePRB7H7MDGwy7ILywnCEUMJoUKVrjctC",2015-09-17T21:48:17.668Z
"0015b6opXD","[redacted]@aol.com","[redacted]@aol.com","$2a$10$WVDWVZrp53fgcKIyw
Mzd/.t4C8zhb4BUyuXsVetE27z5kZJIvtt/S",2016-02-14T20:45:09.302Z
"0032t1HtBA","[redacted]@yahoo.com","[redacted]@yahoo.com","$2a$10$UTMQ4IU
cW4oH4HqfOwbrPuuHV.c0rbCpYGfdgGMaUDkK96mPfcCOa",2016-09-10T02:52:56.238Z
"0035THs1eh","[redacted]@gmail.com","[redacted]@gmail.com","$2a$
10$crbnoezQ1Fxdb5RAqdjUgujJ/
K4roBU3P7v6RYB521A59FQXuj4gq",2015-07-23T01:02:42.647Z
```

https://www.troyhunt.com/data-from-connected-cloudpets-teddy-bears-leaked-and-ransomed-exposing-kids-voice-messages/

Previous Work: #2 - First Ransom

*“**You DB is backed up on our servers**, send 1 BTC to 1J5ADzFv1gx3fsUPUY1AWktuJ6DF9P6hiF then send your ip address to email:kraken0@india.com”*

https://twitter.com/nmerrigan/status/817289743817998337/photo/1

https://pastebin.com/BgJADkqW

Previous Work: #3 - Initial Timeline

2016.12.30 - 2017.01.04:

Multiple security researchers alert to CloudPets via multiple means

2017.01.07:

Ransom #1: Original databases **deleted** + **ransom demand** left on the system via "PLEASE_READ" message

2017.01.08:

Ransom #2: Demand left for "README_MISSING_DATABASES"

Ransom #3: Demand left for "PWNED_SECURE_YOUR_STUFF_SILLY"

2017.01.13:

No databases were found to still be publicly accessible

https://www.troyhunt.com/data-from-connected-cloudpets-teddy-bears-leaked-and-ransomed-exposing-kids-voice-messages/

Previous Work #4: Toy Security

Paul Stone's research: https://www.contextis.com/en/blog/hacking-unicorns-web-bluetooth

The Toy has:

→ **No built-in Bluetooth** security features.

→ **No authentication** for **bonding/pairing** between the **device** and **phone**.

→ Anyone can **connect** to the toy as long as it is **switched on**. (!)

→ **Unencrypted firmware** upgrades only **validation** is a **CRC16 checksum.**

→ Possible to **remotely modify** the toy's **firmware**.

Previous Work #4: Paul Stone’s demo

https://youtu.be/5pQt6Aa3AVs

Previous Work #5: Vendor Response

→ Write-ups on **lack of the security** of the toy and **lack** of **use** of **built-in security features** published.

→ All **attempts** to **warn** Spiral Toys **fail**.

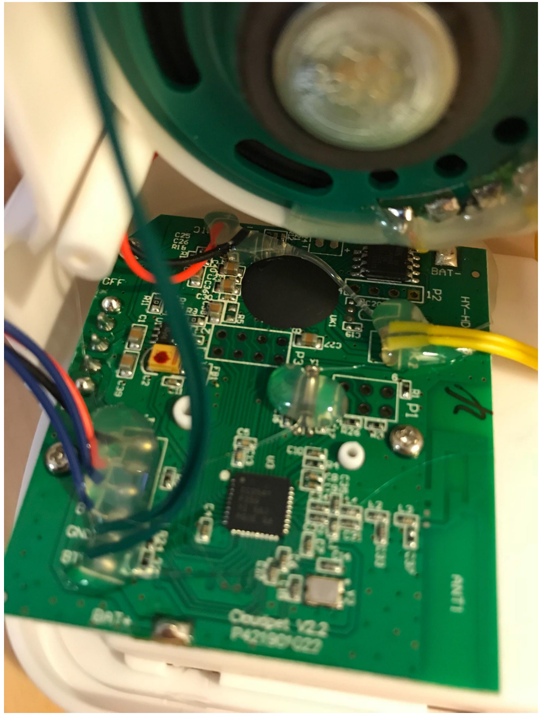
→ Spiral Toys confirms that they **did not reply** to the **data breach emails**, and rather decided to fix them.

Question: What did they fix?

Mozilla asks: Are toys safe now?

Our Work: Viking Style

Unicorn Analysis:



What could possibly go wrong?

PET-01-001 Backend: *Tour domain is for sale and used over clear-text HTTP* (*High*)

CloudPets app directs users to http://mycloudpets.com/tour for tutorials and help.

→ **Domain** is **currently** on **sale**.

→ **Anybody** can **purchase** the **domain** and **influence** users.

→ i.e. **prompting** users for their **CloudPets credentials**.

→ i.e. prompt users to **download malicious apps**.

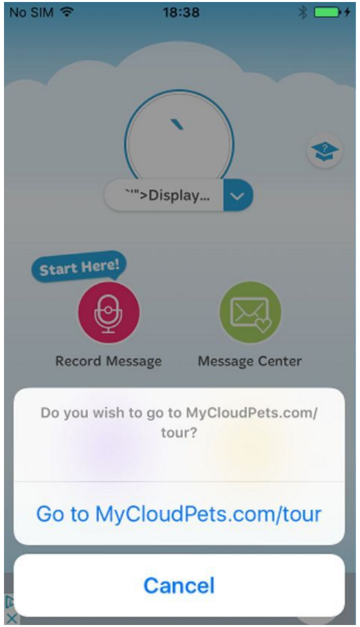
PET-01-001 Backend: *Tour domain is for sale and used over clear-text HTTP* (*High*)

Also:

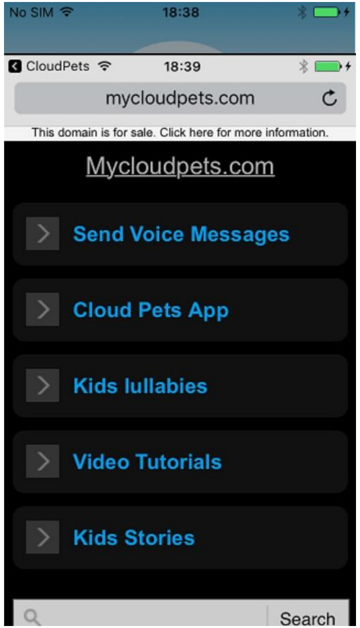
- → The page is **requested via clear-text HTTP.**
- → This makes it easier for a malicious attacker on the local network (i.e. Public WIFI) to trivially **modify the Tour page**.
- → Allows attackers to target users.
- → i.e. ask for user credentials.
- → i.e. prompt users to **download malicious apps**.

PET-01-001 Backend: *Tour domain is for sale and used over clear-text HTTP* (*High*)

Taps on the help icon:



Do you wish to go to MyCloudPets.com/ tour?



This domain is for sale. Click here for more information.

PET-01-001 Backend: *Tour domain is for sale and used over clear-text HTTP* (*High*)

Demo

PET-01-002 Toy: *Authless attacks via Bluetooth remain possible (Critical)*

Paul Stone’s **public PoC** remains working **without any changes**:

https://github.com/pdjstone/cloudpets-web-bluetooth

https://pdjstone.github.io/cloudpets-web-bluetooth/index.html

→ **Strangers** can still connect to the toys **without authentication.**

→ **Push audio & play it on the Toy:**
Anyone can interact with the child: i.e. “Open the door…”

→ **Download audio from the toy:**
Turns the toys into **spy devices**.

PET-01-003 Toy: No firmware protections is in place (*High*)

Lack of adequate firmware verification remains:

→ A **discovery** was made **during** the **initial setup** of the **device.**

→ **Firmware is installed** into the device from the app **via BLE**.

→ The **installation** process **still** has **no verification**:

- NO **signature** or **integrity** checks in place.

→ The only “**protection”** is a **CRC16 checksum**.

PET-01-004 Backend: CloudPets voice recordings world-reachable (*High*)

→ Audio recordings created from the device are still being saved at **cloudpet-prod.s3.amazonaws.com**.

→ When users upload a new avatar or message, the application will post the data through the **API** and carries out a **DNS** lookup to **cloudpet-prod.s3.amazonaws.com.**

→ The **S3 Bucket** has no authorization or authentication in place.

→ There are **no limitations** when it comes to **accessing** the **files** placed in the basket.

CloudPets Summary

→ CloudPets Toys, can store and replay **voice messages**

→ Exposed personal **information** of more than **800k customers**.

→ Effectively turned the toys into potential spy devices

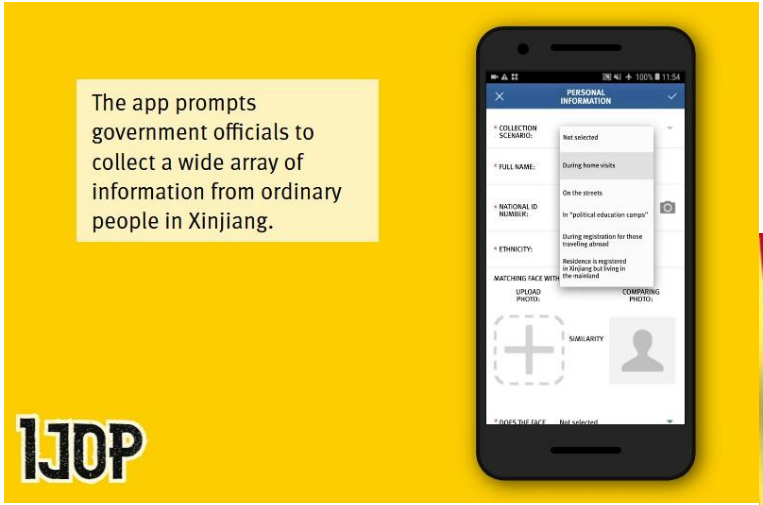
→ Mozilla asked for a retest of issues:

Spiral toys didn’t fix much, other than the Mongo DB without authentication

→ Toys now removed from Amazon, Walmart, etc.

https://www.mic.com/articles/189673/target-and-walmart-stop-selling-the-superhackable-kids-toy-cloudpets-after-pressure-from-mozilla

Part 2: Chinese Police Apps - IJOP & BXAQ



The app prompts government officials to collect a wide array of information from ordinary people in Xinjiang.

PERSONAL INFORMATION

* COLLECTION SCENARIO:

- Not selected
- During home visits
- On the streets
- In "political education camps"
- During registration for those traveling abroad
- Residence is registered in Xinjiang but living in the mainland

* FULL NAME:

* NATIONAL ID NUMBER:

* ETHNICITY:

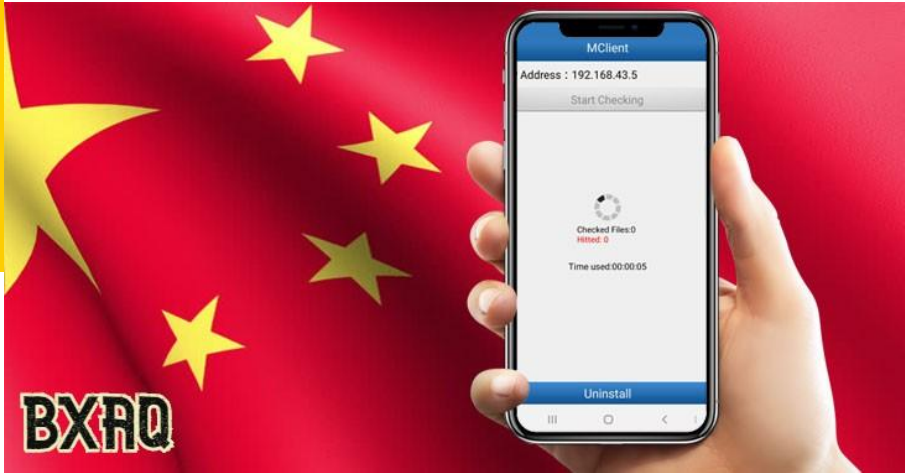
MATCHING FACE WITH

UPLOAD PHOTO:

SIMILARITY

COMPARING PHOTO:

* DOES THE FACE Not selected



MClient

Address : 192.168.43.5

Start Checking

Checked Files:0
Hitted: 0

Time used:00:00:05

Uninstall

Brief Background

The **Chinese government** uses IJOP and BXAQ to evaluate the “threat level” of their **minority Muslim population** in **Xinjiang**

https://www.hrw.org/report/2019/05/01/chinas-algorithms-repression/reverse-engineering-xinjiang-police-mass-surveillance

Human Rights Watch (HRW) requested a project funded by the **Open Technology Fund (OTF)** to report and document the findings of the apps:

To investigate the **potential violation** of **human rights**

Question:

Is “The system of systems” IJOP potentially compromising the human rights of a minority Chinese population?

What is the IJOP app?

How did the Chinese government use this?

→ The **Integrated Joint Operations Platform** or (IJOP) is a policing program

→ Based on **big data analysis**.

→ The program **aggregates data about people**, often without their knowledge.

→ **Flags data** it deems **potentially threatening** to officials.

Source: From Memo on IJOP provided by HRW

https://www.hrw.org/news/2018/02/26/china-big-data-fuels-crackdown-minority-region

What is the IJOP app?

WSJ
They would punch the ID number in
and then they would see everything about you.

https://www.youtube.com/watch?v=_Hy9eIjkmOM

Details on IJOP

- → Originally developed by **China Electronics Technology Group Corporation (CETC)**, a state-owned military contractor.

- → The company was set up in **2017** as a research centre at **UTS**, the University of Technology, Sydney.

- → They primarily focused on areas such as:
  - ○ **Artificial Intelligence**
  - ○ **Mapping**
  - ○ **Big data**

Preliminaries #1: Language barrier

App written for Chinese Police is in Chinese, obviously :)

Had to:

1. Decompile APK → apktool d ijop.apk
2. Translate strings.xml → Google Translate
3. Recompile APK → apktool + sign + install

<3 https://ibotpeaches.github.io/Apktool/

Result:

Broken-English app version :)

Preliminaries #2: Setting up Tor

Just in case :)

Goal: Refrain from any form of behavior that could be perceived as “noisy” and, thus, alert the maintainers of the application and/or server owners.

Had to:

1. Setup Tor in test environment
2. Setup Burp to use Tor as outbound proxy
3. MitM app to send traffic through Burp

Result:

We can inspect traffic, but without leaking our IPs

Preliminaries #3: What is after the login?

Problems:

1. App requires authentication to Chinese Police servers = **We cannot login**
2. We **don’t know how (politically) bad** the info on the screen is

Solution: **Show after-login screens to human right activists**

1. Export all app activities
2. Provide tweaked build to activists to access all activities
3. Provide list of ADB commands to activists to see activities
4. Activists can now run a script to review after-login screens quickly

What we Found in the "IJOP" app

XJ1-01-002 NewCollection activity tracks blood type and political views (Proven)

The **HRW's** question was as follows:

*"**Recording of Height and Blood Type:** We want to know to what extent the authorities can justify this by saying this is all for counter-terrorism. So far, I see only a few mentions of terrorism."*

XJ1-01-002 NewCollection activity tracks blood type and political views (Proven)

→ IJOP app indulges in data collection likely for **anomaly detection**

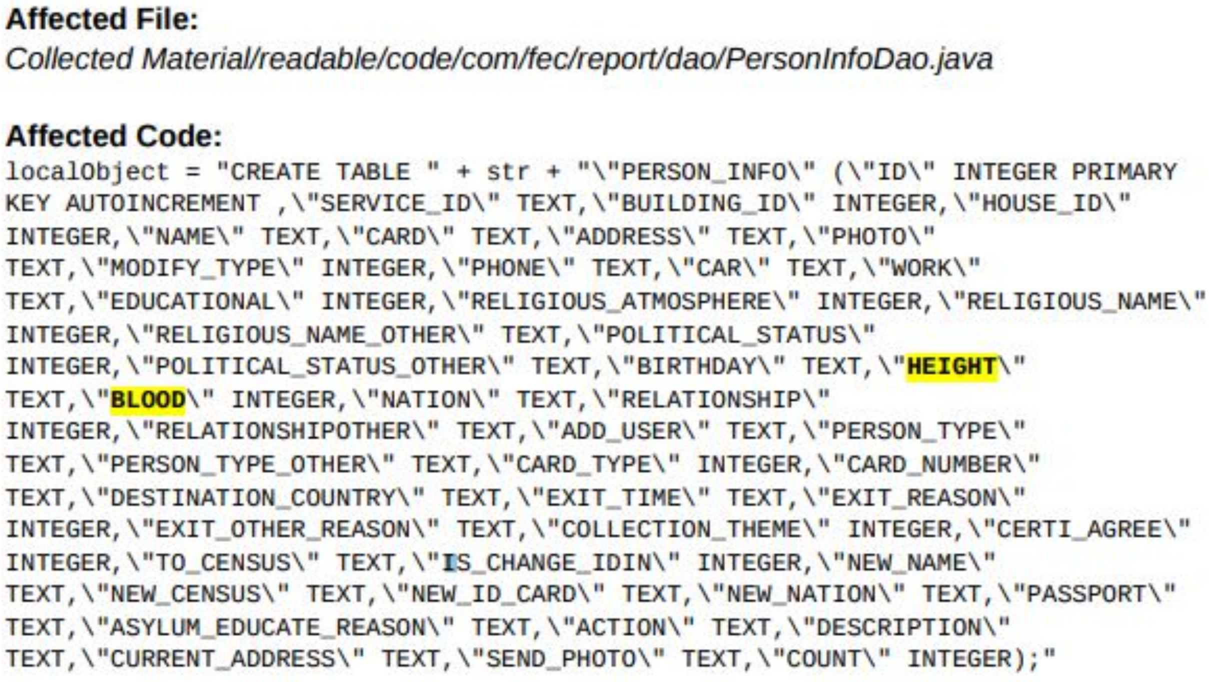
→ It is **assumed** that the **goal** of this **data collection** is to have more **reference data**.

→ Namely, to **mine** and **gather data** on **individuals**.

→ With increasingly **strong indicators**, **data that does not match** the **information** from the **HQ**, may reveal more **suspicious** and **problematic** subsets of **users/actual people** and **groups**.

XJ1-01-002 NewCollection activity tracks blood type and political views (Proven)

Gathered personal information around height and blood type:



Affected File:

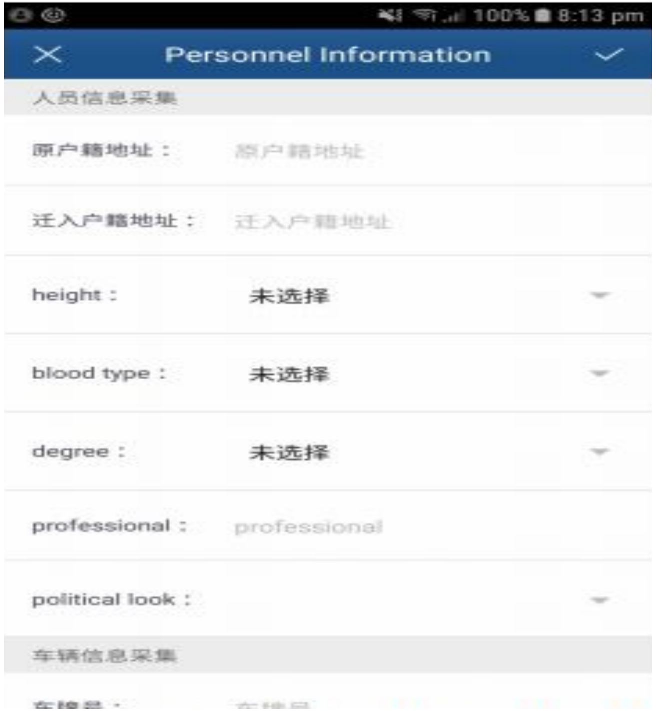
Collected Material/readable/code/com/fec/report/dao/PersonInfoDao.java

Affected Code:

```
localObject = "CREATE TABLE " + str + "\"PERSON_INFO\" (\"ID\" INTEGER PRIMARY
KEY AUTOINCREMENT ,\"SERVICE_ID\" TEXT,\"BUILDING_ID\" INTEGER,\"HOUSE_ID\"
INTEGER,\"NAME\" TEXT,\"CARD\" TEXT,\"ADDRESS\" TEXT,\"PHOTO\"
TEXT,\"MODIFY_TYPE\" INTEGER,\"PHONE\" TEXT,\"CAR\" TEXT,\"WORK\"
TEXT,\"EDUCATIONAL\" INTEGER,\"RELIGIOUS_ATMOSPHERE\" INTEGER,\"RELIGIOUS_NAME\"
INTEGER,\"RELIGIOUS_NAME_OTHER\" TEXT,\"POLITICAL_STATUS\"
INTEGER,\"POLITICAL_STATUS_OTHER\" TEXT,\"BIRTHDAY\" TEXT,\"HEIGHT\"
TEXT,\"BLOOD\" INTEGER,\"NATION\" TEXT,\"RELATIONSHIP\"
INTEGER,\"RELATIONSHIPOTHER\" TEXT,\"ADD_USER\" TEXT,\"PERSON_TYPE\"
TEXT,\"PERSON_TYPE_OTHER\" TEXT,\"CARD_TYPE\" INTEGER,\"CARD_NUMBER\"
TEXT,\"DESTINATION_COUNTRY\" TEXT,\"EXIT_TIME\" TEXT,\"EXIT_REASON\"
INTEGER,\"EXIT_OTHER_REASON\" TEXT,\"COLLECTION_THEME\" INTEGER,\"CERTI_AGREE\"
INTEGER,\"TO_CENSUS\" TEXT,\"IS_CHANGE_IDIN\" INTEGER,\"NEW_NAME\"
TEXT,\"NEW_CENSUS\" TEXT,\"NEW_ID_CARD\" TEXT,\"NEW_NATION\" TEXT,\"PASSPORT\"
TEXT,\"ASYLUM_EDUCATE_REASON\" TEXT,\"ACTION\" TEXT,\"DESCRIPTION\"
TEXT,\"CURRENT_ADDRESS\" TEXT,\"SEND_PHOTO\" TEXT,\"COUNT\" INTEGER);"
```

XJ1-01-002 NewCollection activity tracks blood type and political views (Proven)

NewCollection activity with the items of concern.



100% 8:13 pm

Personnel Information

人员信息采集

原户籍地址： 原户籍地址

迁入户籍地址： 迁入户籍地址

height： 未选择

blood type： 未选择

degree： 未选择

professional： professional

political look：

车辆信息采集

车牌号： 车牌号

XJ1-01-005 App receives electricity consumption data from HQ (Proven)

HRW wanted to know about application’s data on **electricity consumption**.

Authorities logging people’s electricity use, **how is it problematic**?

HRW’s take:

→ “ Is it problematic because the **authorities are logging everyone’s electricity use**. “

→ “ They are trying to see if there’s a **reason** for **“abnormal” level of electricity use**. “

→ The application **employs** a **database** which fetches various **utility data** about an **individual**.

XJ1-01-005 App receives electricity consumption data from HQ (Proven)

→ A police officer receives a **new task** from the **HQ**.

→ A **police officer** can **file** a **report** to **investigate** the occurrence of **unusual power consumption**.

→ They can do so on any **given date** and may mark **any reasons** for the same.

→ This prepares the **ground** for further **investigation** by the **public security agency**.

XJ1-01-005 App receives electricity consumption data from HQ (Proven)

→ In case of a **false positive**, the officer can **file** in the **actual electricity meter value**.

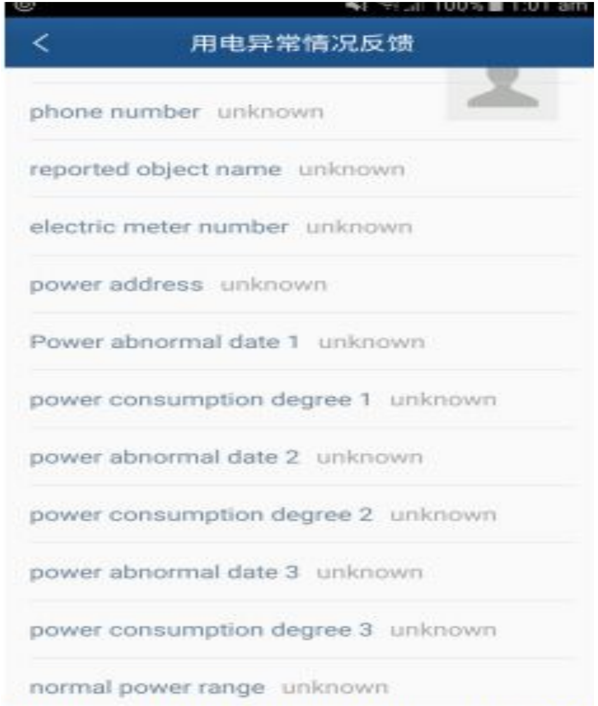
→ A **justification** by **law enforcement** might be to monitor the use of electricity for

- **Cryptocurrency mining**
- **Growing cannabis indoors.**

→ These **types** of **activities** lead to **increased consumption**.

XJ1-01-005 App receives electricity consumption data from HQ (Proven)

Rendered Activity for CheckElcInfoActivity



phone number unknown

reported object name unknown

electric meter number unknown

power address unknown

Power abnormal date 1 unknown

power consumption degree 1 unknown

power abnormal date 2 unknown

power consumption degree 2 unknown

power abnormal date 3 unknown

power consumption degree 3 unknown

normal power range unknown

XJ1-01-005 App receives electricity consumption data from HQ (Proven)

What do you think?

→ Should the government be able to monitor how much electricity everybody uses?

→ Is this a privacy intrusion?

→ Can this stop terrorist attacks?

XJ1-01-006 Reporting feature for problematic tools (Proven)

HRW had doubts about the tools. Specifically,

→ “Tools they use and possess: An option under “electronic” appears to refer to **problematic tools**.“

→ “These includes tools that may be implemented to make **explosives**.“

XJ1-01-006 Reporting feature for problematic tools (Proven)

Proven Facts

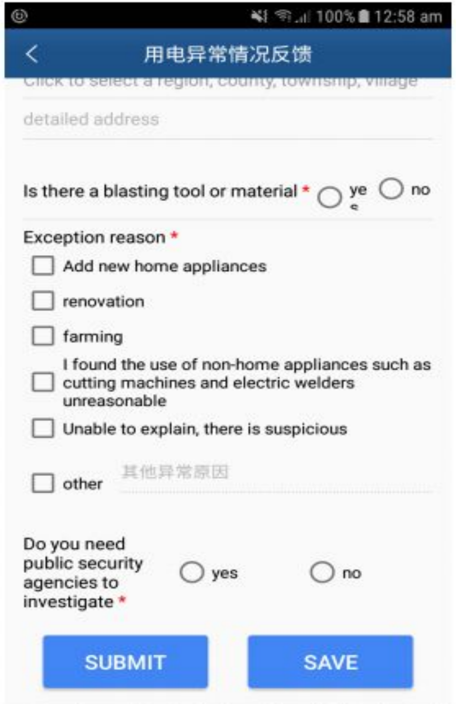
→ **Officers** can **submit** a **report** to the **HQ** about **explosive materials** and **tools**.

→ Furthermore, an **officer** can **ask** for an **investigation** of the matter.

→ This **investigation** can be further **handed over** and **carried on** by the **public security agency**.

XJ1-01-006 Reporting feature for problematic tools (Proven)

Rendered Activity for CheckElcInfoActivity



Click to select a region, county, township, village

detailed address

Is there a blasting tool or material * ○ yes ○ no

Exception reason *

□ Add new home appliances

□ renovation

□ farming

□ I found the use of non-home appliances such as cutting machines and electric welders unreasonable

□ Unable to explain, there is suspicious

□ other 其他异常原因

Do you need public security agencies to investigate * ○ yes ○ no

Fig.: Rendered Activity for CheckElcInfoActivity.

XJ1-01-014 PII collection via MissPhoneTrailFeedbackActivity (Assumed)

HRW was also curious about the MissPhoneTrailFeedbackActivity:

→ *“This refers to when someone’s gone off grid — **suddenly stopped using their phone.**”*

Summary answer:

Reviewing the activity and data processing from the decompiled source code provides evidence of tracking this information.

XJ1-01-014 PII collection via MissPhoneTrailFeedbackActivity (Assumed)

The following list summarizes the fields used by the app for this purpose:

addUser, addUserName, expression, expressionDesc, fkMptv, id, latitude, locationDescription, longitude, missTrailReason, note, otherReason, police, policeCheck, policeReason, relationship, telNumber, userOrgName, userOrganizationId

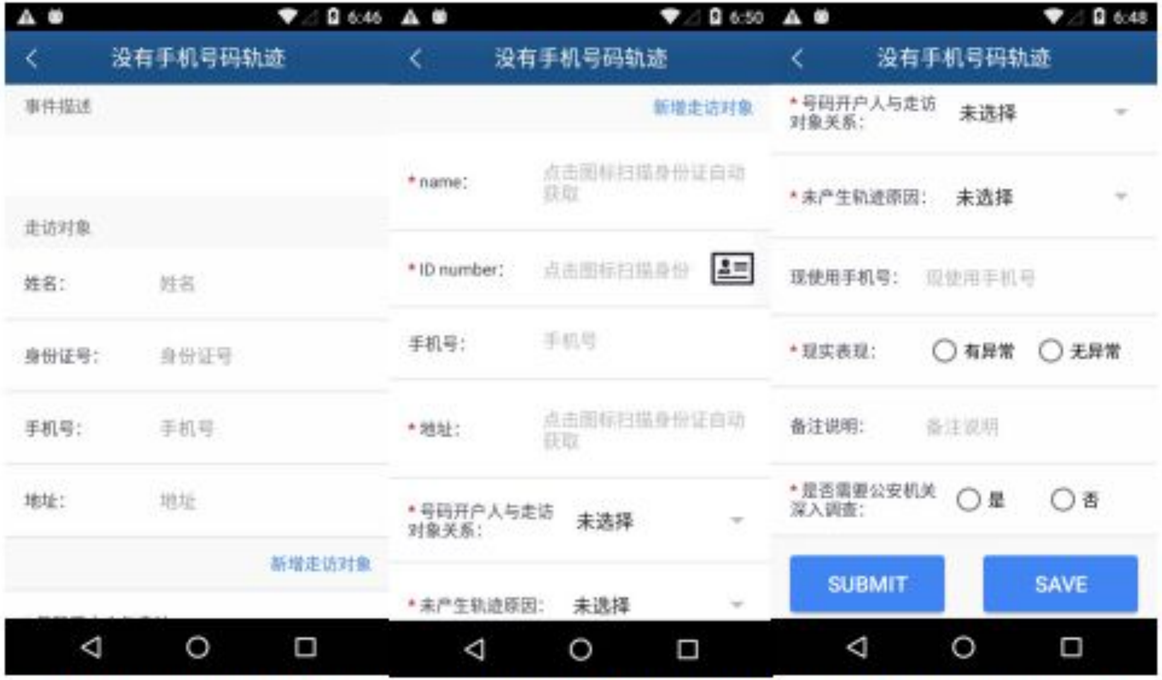
Investigated launching the activity as follows:

ADB Command:

```
adb shell am start -n
"com.hbfec.xjoneproject/com.fec.xjoneproject.ui.task.miss_phone_trail.activity.MissPhoneTrailFeedbackActivity"
```

XJ1-01-014 PII collection via MissPhoneTrailFeedbackActivity (Assumed)

This resulted in the following PII collection form being rendered:



没有手机号码轨迹

事件描述

走访对象

姓名： 姓名

身份证号： 身份证号

手机号： 手机号

地址： 地址

新增走访对象

没有手机号码轨迹

新增走访对象

*name： 点击图标扫描身份证自动获取

*ID number： 点击图标扫描身份

手机号： 手机号

*地址： 点击图标扫描身份证自动获取

*号码开户人与走访对象关系： 未选择

*未产生轨迹原因： 未选择

没有手机号码轨迹

*号码开户人与走访对象关系： 未选择

*未产生轨迹原因： 未选择

现使用手机号： 现使用手机号

*现实表现： ○ 有异常 ○ 无异常

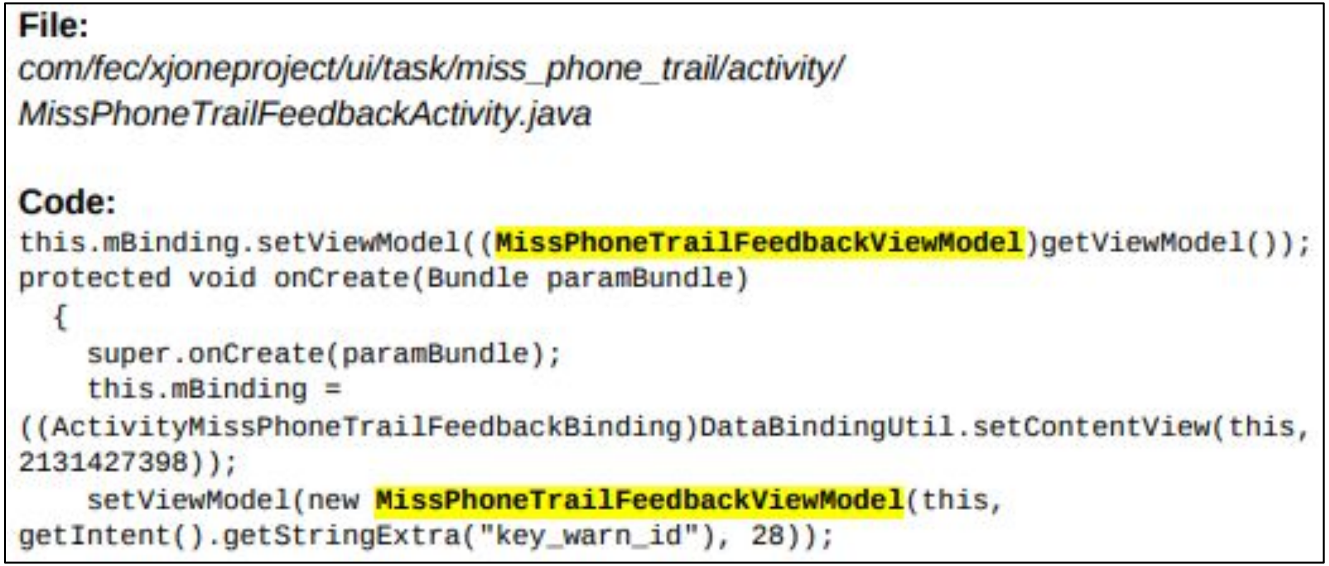
备注说明： 备注说明

*是否需要公安机关深入调查： ○ 是 ○ 否

Fig.: Data collection form rendered by MissPhoneTrailFeedbackActivity.

XJ1-01-014 PII collection via MissPhoneTrailFeedbackActivity (Assumed)

Upon closer inspection of the decompiled source code of the application, it was found that MissPhoneTrailFeedbackActivity **makes use of the MissPhoneTrailFeedbackViewModel**.



File:

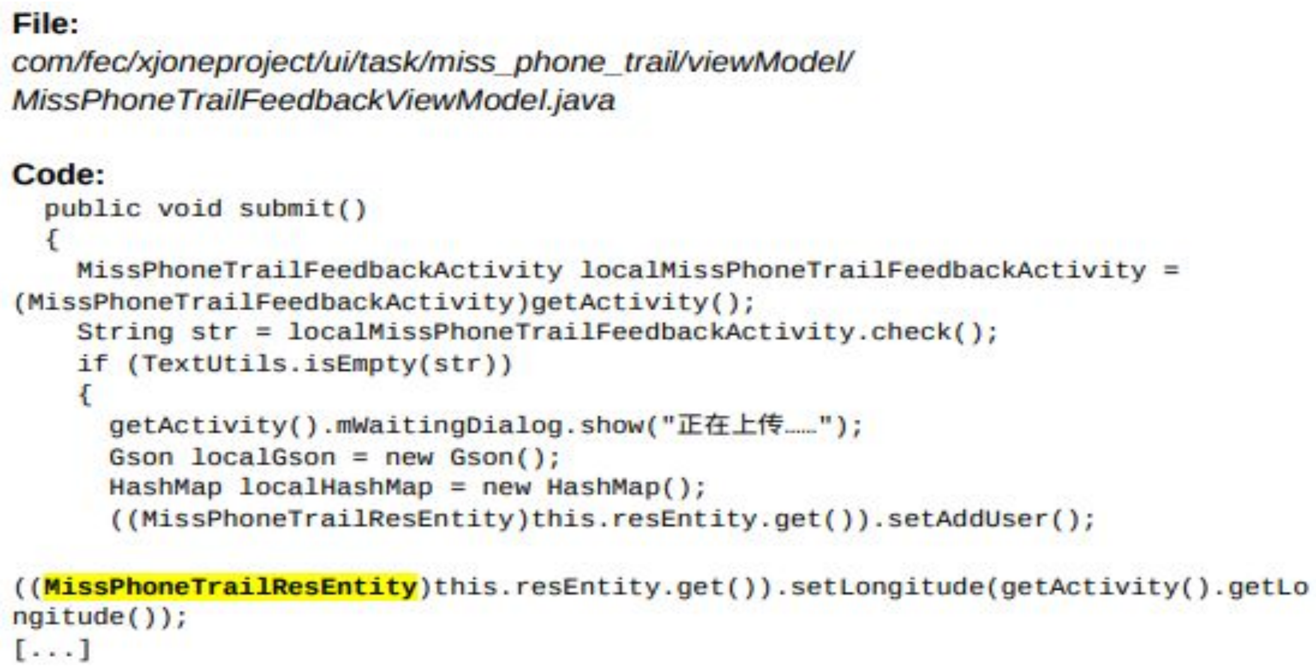
com/fec/xjoneproject/ui/task/miss_phone_trail/activity/MissPhoneTrailFeedbackActivity.java

Code:

```
this.mBinding.setViewModel((MissPhoneTrailFeedbackViewModel)getViewModel());
protected void onCreate(Bundle paramBundle)
  {
    super.onCreate(paramBundle);
    this.mBinding =
((ActivityMissPhoneTrailFeedbackBinding)DataBindingUtil.setContentView(this,
2131427398));
    setViewModel(new MissPhoneTrailFeedbackViewModel(this,
getIntent().getStringExtra("key_warn_id"), 28));
```

XJ1-01-014 PII collection via MissPhoneTrailFeedbackActivity (Assumed)

The sequence submits the information to the API in a fashion illustrated next and making use of a **MissPhoneTrailResEntity**.



File:
com/fec/xjoneproject/ui/task/miss_phone_trail/viewModel/MissPhoneTrailFeedbackViewModel.java

Code:

```
  public void submit()
  {
    MissPhoneTrailFeedbackActivity localMissPhoneTrailFeedbackActivity =
(MissPhoneTrailFeedbackActivity)getActivity();
    String str = localMissPhoneTrailFeedbackActivity.check();
    if (TextUtils.isEmpty(str))
    {
      getActivity().mWaitingDialog.show("正在上传......");
      Gson localGson = new Gson();
      HashMap localHashMap = new HashMap();
      ((MissPhoneTrailResEntity)this.resEntity.get()).setAddUser();

((MissPhoneTrailResEntity)this.resEntity.get()).setLongitude(getActivity().getLo
ngitude());
[...]
```

XJ1-01-014 PII collection via MissPhoneTrailFeedbackActivity (Assumed)

From the MissPhoneTrailResEntity.java file, a more readable list of fields was collected.

Command:

```
cat $(find . -name MissPhoneTrailResEntity.java) | grep -i private|cut -f5 -d" "|cut -f1 -d';' | sort -u | tr "\n" ","|sed 's|,|, |g'
```

Output:

```
addUser, addUserName, expression, expressionDesc, fkMptv, id, latitude,
locationDescription, longitude, missTrailReason, note, otherReason, police,
policeCheck, policeReason, relationship, telNumber, userOrgName,
userOrganizationId
```

XJ1-01-022 HQ communications for investigations and arrest (Assumed)

HRW was also wondering about the following:

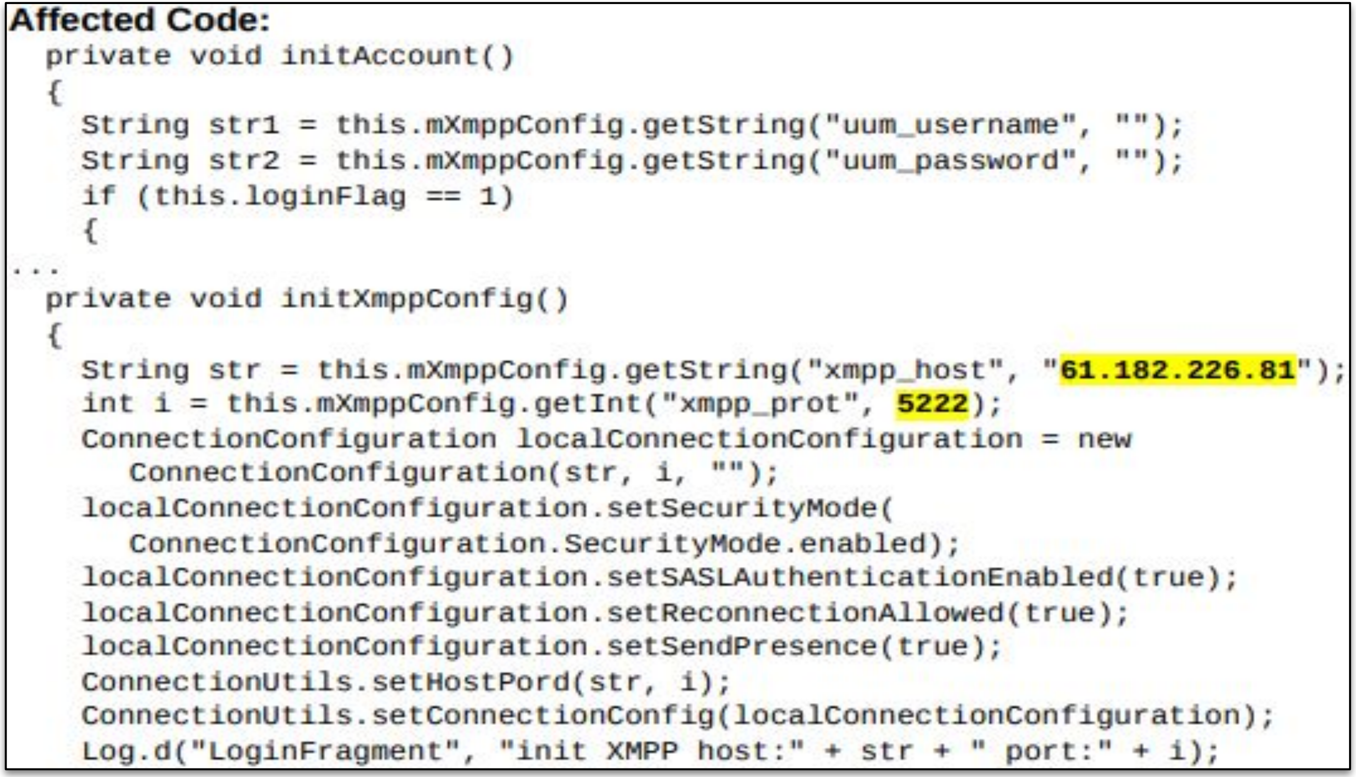
*"**How does the officer communicate with the HQ?**. I see there are terms like "**immediate arrest**" or "**detain for investigation**"—are these decisions made by officers or HQ makes."*

Most likely communication and some form of chat exchange takes place via the XMPP protocol. This can be assumed from the functions shown next.

Affected File:

com/fec/xjoneproject/ui/LoginFragment.java

XJ1-01-022 HQ communications for investigations and arrest (Assumed)



Affected Code:

```
  private void initAccount()
  {
    String str1 = this.mXmppConfig.getString("uum_username", "");
    String str2 = this.mXmppConfig.getString("uum_password", "");
    if (this.loginFlag == 1)
    {
...
  private void initXmppConfig()
  {
    String str = this.mXmppConfig.getString("xmpp_host", "61.182.226.81");
    int i = this.mXmppConfig.getInt("xmpp_prot", 5222);
    ConnectionConfiguration localConnectionConfiguration = new
       ConnectionConfiguration(str, i, "");
    localConnectionConfiguration.setSecurityMode(
       ConnectionConfiguration.SecurityMode.enabled);
    localConnectionConfiguration.setSASLAuthenticationEnabled(true);
    localConnectionConfiguration.setReconnectionAllowed(true);
    localConnectionConfiguration.setSendPresence(true);
    ConnectionUtils.setHostPord(str, i);
    ConnectionUtils.setConnectionConfig(localConnectionConfiguration);
    Log.d("LoginFragment", "init XMPP host:" + str + " port:" + i);
```

XJ1-01-022 HQ communications for investigations and arrest (Assumed)

→ **Unclear** how exactly these variables are used in the context of the app

→ There are **indicators that information pushed by the HQ**

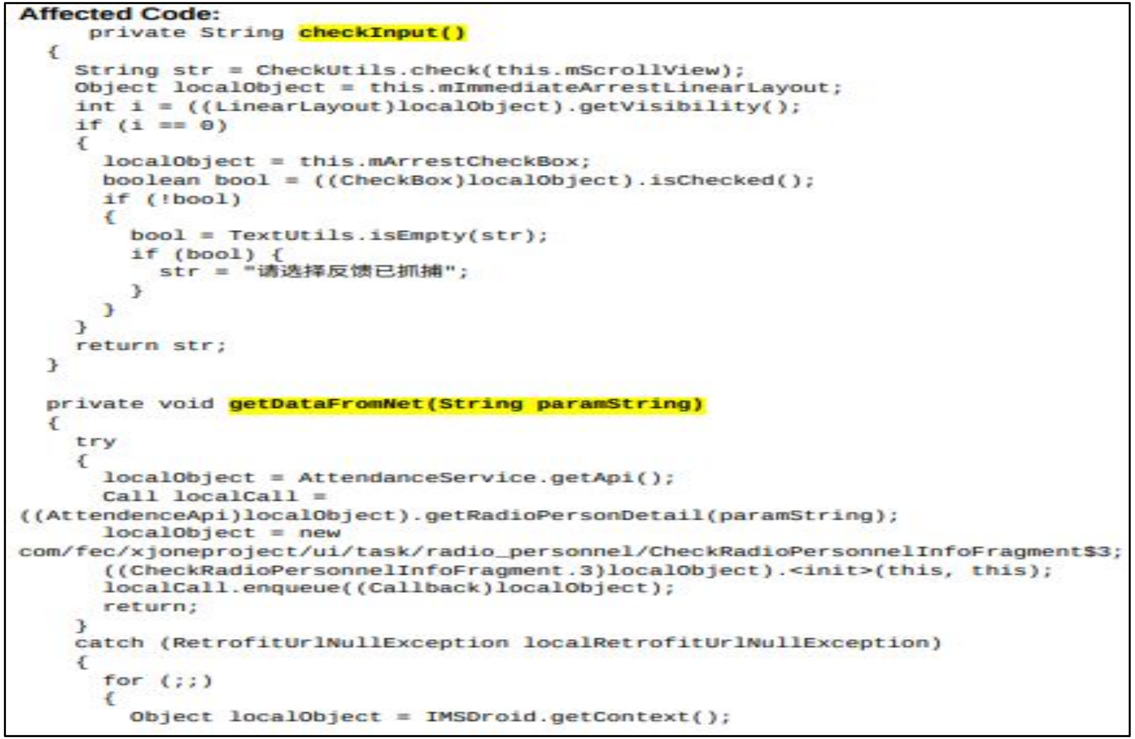
→ Setting the values for immediate investigations and immediate arrests

→ However, it seems **police officers might be able to edit this information as well.**

Affected File:

Collected Material/readable/code/com/fec/xjoneproject/ui/task/radio_personnel/CheckRadioPersonnelInfoFragment.java

XJ1-01-022 HQ communications for investigations and arrest (Assumed)



Affected Code:

```
  private String checkInput()
 {
  String str = CheckUtils.check(this.mScrollView);
  Object localObject = this.mImmediateArrestLinearLayout;
  int i = ((LinearLayout)localObject).getVisibility();
  if (i == 0)
  {
   localObject = this.mArrestCheckBox;
   boolean bool = ((CheckBox)localObject).isChecked();
   if (!bool)
   {
    bool = TextUtils.isEmpty(str);
    if (bool) {
     str = "请选择反馈已抓捕";
    }
   }
  }
  return str;
 }

 private void getDataFromNet(String paramString)
 {
  try
  {
   localObject = AttendanceService.getApi();
   Call localCall =
((AttendenceApi)localObject).getRadioPersonDetail(paramString);
   localObject = new
com/fec/xjoneproject/ui/task/radio_personnel/CheckRadioPersonnelInfoFragment$3;
   ((CheckRadioPersonnelInfoFragment.3)localObject).<init>(this, this);
   localCall.enqueue((Callback)localObject);
   return;
  }
  catch (RetrofitUrlNullException localRetrofitUrlNullException)
  {
   for (;;)
   {
    Object localObject = IMSDroid.getContext();
```

Sources of IJOP

The **multiple sources** or “**sensors**” for **intel** through IJOP are as follows:

→ **CCTV cameras** with **facial recognition** and **night vision**.

→ **WiFi sniffers** that **track** computers, smartphones, and other networked devices.

→ Other sources are **licence plate numbers**, **citizen ID card numbers**.

→ Vehicle ownership, health insurance, family planning, banking, and legal records.

https://www.hrw.org/report/2019/05/01/chinas-algorithms-repression/reverse-engineering-xinjiang-police-mass-surveillance

Current Status of IJOP

→ There are few checks on police surveillance powers.

→ **No effective privacy** against government intrusions in China.

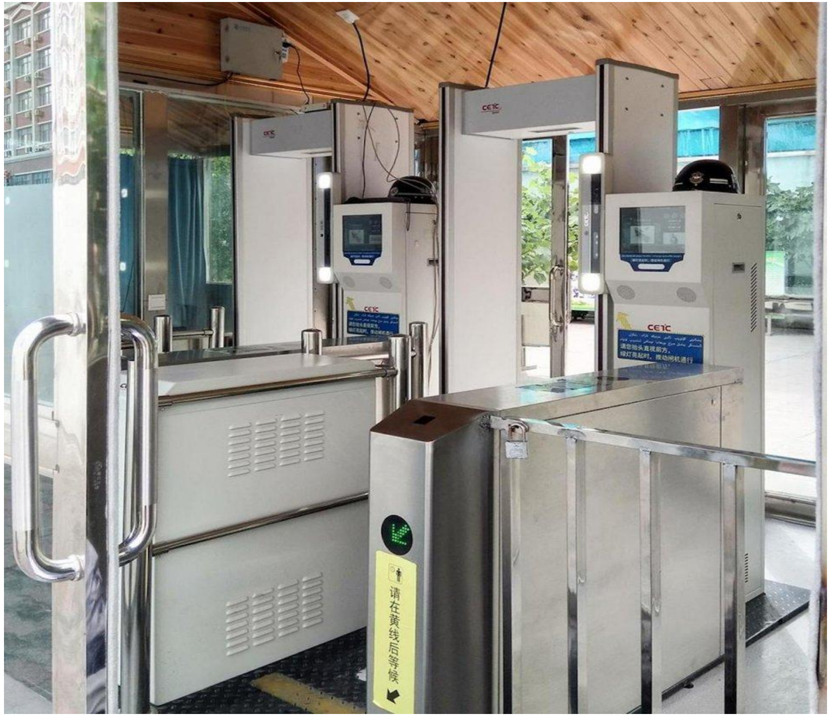
→ **China has no unified privacy or data protection law** to protect **personally identifying information** from **misuse**.

→ There is very **little information** available about how **securely**, the **data collected** by **IJOP** is **stored**.

→ There is **no formal system** for people to find out what **information** is held about them in **IJOP**.

https://www.hrw.org/report/2019/05/01/chinas-algorithms-repression/reverse-engineering-xinjiang-police-mass-surveillance

Data Collecting Machines/ Data Doors



https://www.scmp.com/tech/big-tech/article/3029333/chinas-data-doors-scoop-information-straight-your-phone

Conclusion

→ It is **unclear** how the **IJOP** and **Police Cloud** are **related.**

→ It is **unclear** if, and how, IJOP **connects** to other **databases** on **people** the **police manage** or have **access to**

→ Capable of collecting and managing vast amounts of **very specific data**

→ **Data** has the potential to become a basis for further action concerning a specific group

→ The application has a **tracking** of power over **energy consumption**, the recording of **political views** and the **religious atmosphere**

Conclusion

→ **Review** of the **IJOP** mobile app was carried out in close **collaboration** with the **Human Rights Watch team**

→ At the same time, it should be noted that **we operated** as a purely **technically-driven** team and an **unbiased investigating entity**

→ We worked from a premise of **technical evidence**, which is **based** on **reverse-engineering** operations

A police officer checks a Uighur man's ID documents in Kashgar, Xinjiang, in March 2017

(Credits: Thomas Peters/Reuters)

What is the "BXAQ" app?

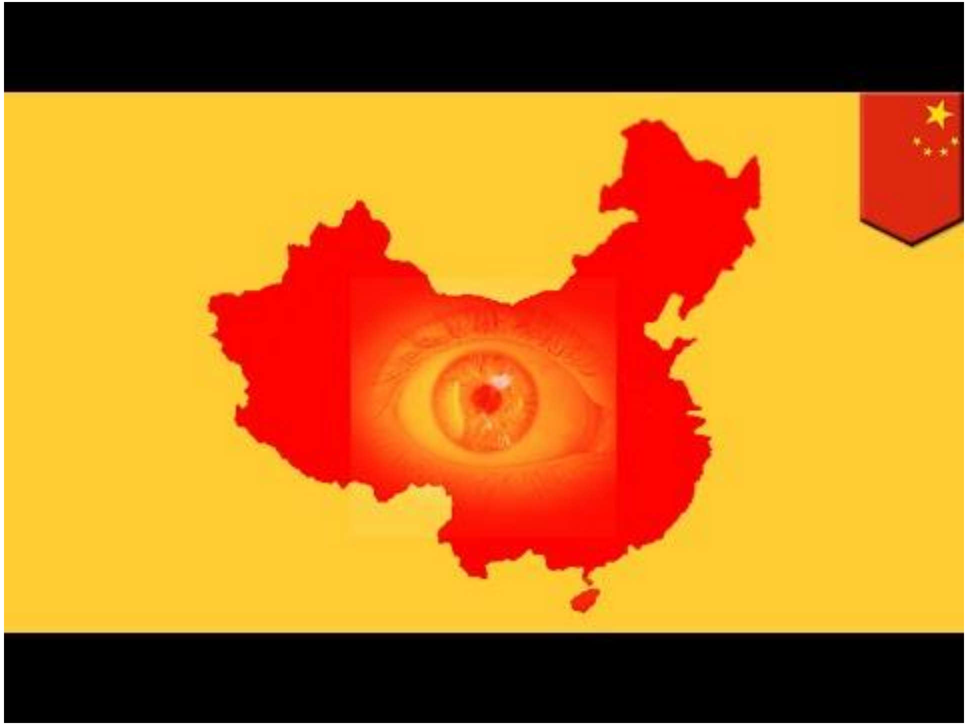
→ Widely known by the name of **Feng Cai**, BXAQ is an app used to scan the device in which it is installed, even on tourist phones:

https://www.vice.com/en_us/article/7xgame/at-chinese-border-tourists-forced-to-install-a-text-stealing-piece-of-malware

→ It is supposedly used in **specific regions** of China by **Law Enforcement personnel**.

→ Has the potential to **Gather & Manage** massive amount of data about **specific** group.

What is the "BXAQ" app?



https://www.youtube.com/watch?v=4St1E05umNQ

BXAQ Project

→ Access to App **APK** was provided by **OTF**

→ Assessment focused on the mobile application for **Android** phones

→ The project followed the so-called **white-box methodology**.

→ The assessment tackled both the **source code** and the **running application**.

BXAQ Project

→ **Open Technology Fund** (OTF) **requested** and **sponsored** the assessment of the **BXAQ** mobile application used by the **Chinese government**.

→ The app **isn't available** in the **Google Play Store** for **download** and needs to be **installed** by "**side-loading**" or **requesting** certain **permissions**.

→ The **research** was carried about as **delicately** as possible.

→ **Loud display** would **raise** an **element** of **suspicion** and **alert** the **server owners** and **maintainers** of the app.

BXA-01-001 - Information gathering (Proven)

The information that was **collected** through the app **included**:

→ Calendar entries, phone contacts, country codes and dialed numbers
→ (IMEI, IMSI, PhoneSN, WifiMac, BluetoothMac, and if the device is rooted
→ Android Model (CPU_ABI, BOARD, HARDWARE)
→ Texts Messages: Sent and Received
→ Information about the current base-station
→ Used hardware (Mac Addresses)

BXA-01-001 - Information gathering (Proven)

→ Information accessible for various installed apps + an MD5 hash of the app

→ Extracted information searched by the **GetVirAccount** application which searches the /sdcard for specific data of specific China-related apps. This information contains phone numbers and email addresses

→ Every call and message details are extracted and could be used to arrive at a conclusion about visits to **foreign countries**.

BXA-01-001 - Information gathering (Proven)

→ GetVirAccount is one of the applications shipped by the police-app

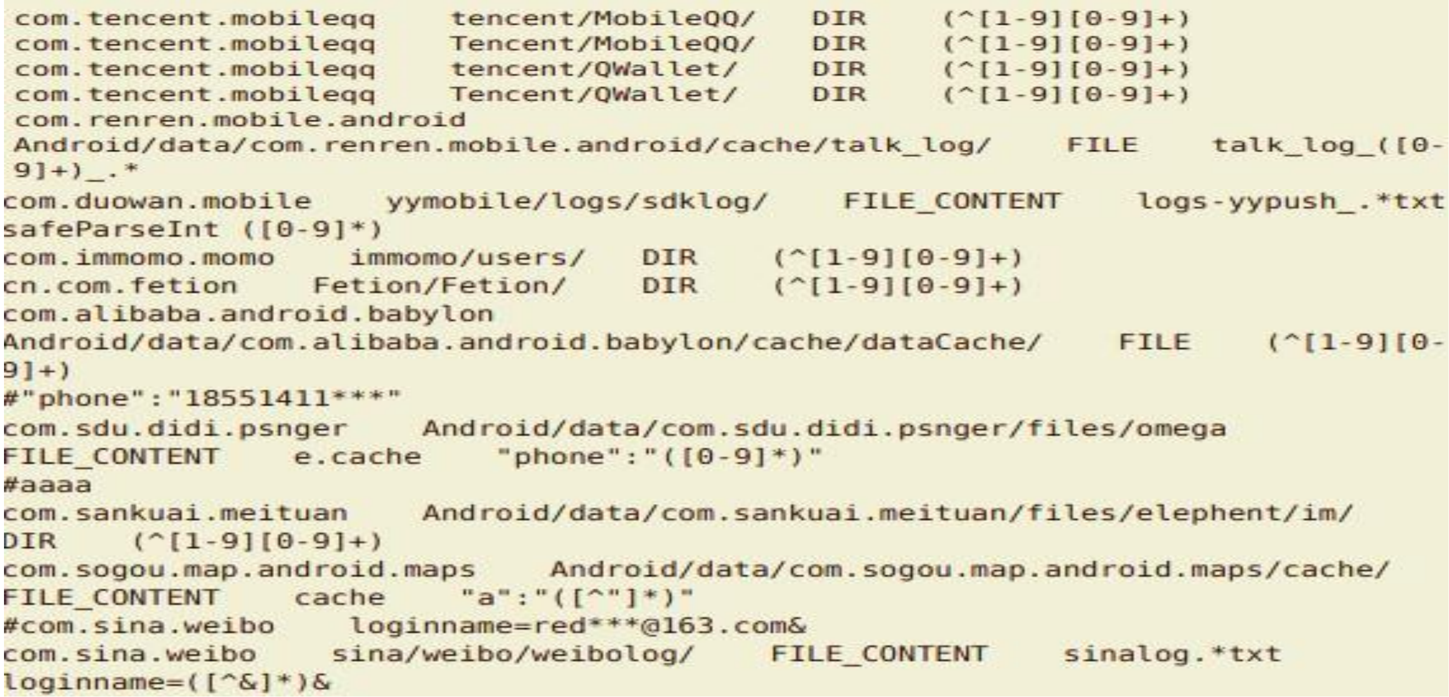
→ Extracted and stored in the apps data storage during installation process.

→ GetVirAccount gets executed by the app

→ GetVirAccount parses the id.conf file, which is shipped by the app as well

BXA-01-001 - Information gathering (Proven)

The id.conf file contains the following data:



```
 com.tencent.mobileqq     tencent/MobileQQ/    DIR     (^[1-9][0-9]+)
 com.tencent.mobileqq     Tencent/MobileQQ/    DIR     (^[1-9][0-9]+)
 com.tencent.mobileqq     tencent/QWallet/     DIR     (^[1-9][0-9]+)
 com.tencent.mobileqq     Tencent/QWallet/     DIR     (^[1-9][0-9]+)
 com.renren.mobile.android
 Android/data/com.renren.mobile.android/cache/talk_log/     FILE     talk_log_([0-
 9]+)_.*
com.duowan.mobile     yymobile/logs/sdklog/     FILE_CONTENT     logs-yypush_.*txt
safeParseInt ([0-9]*)
com.immomo.momo     immomo/users/    DIR     (^[1-9][0-9]+)
cn.com.fetion     Fetion/Fetion/     DIR     (^[1-9][0-9]+)
com.alibaba.android.babylon
Android/data/com.alibaba.android.babylon/cache/dataCache/     FILE     (^[1-9][0-
9]+)
#"phone":"18551411***"
com.sdu.didi.psnger     Android/data/com.sdu.didi.psnger/files/omega
FILE_CONTENT     e.cache     "phone":"([0-9]*)"
#aaaa
com.sankuai.meituan     Android/data/com.sankuai.meituan/files/elephent/im/
DIR     (^[1-9][0-9]+)
com.sogou.map.android.maps     Android/data/com.sogou.map.android.maps/cache/
FILE_CONTENT     cache     "a":"([^"]*)"
#com.sina.weibo     loginname=red***@163.com&
com.sina.weibo     sina/weibo/weibolog/     FILE_CONTENT     sinalog.*txt
loginname=([^&]*)&
```

BXA-01-001 - Information gathering (Proven)

→ All the **referenced file paths** are looked up inside the /**sdcard**/ folder of the phone

→ App is searching for **phone numbers** and **login names** of the defined apps

→ App uses its **SMS permissions** to dump all stored text messages and includes them in the report

→ **Wifiscan** application does check all files stored on the SD card via hash comparison

→ All extracted information is bundled as a **ZIP file**, without applying **Password protection**

BXA-01-002: File transmission and protection (Proven)

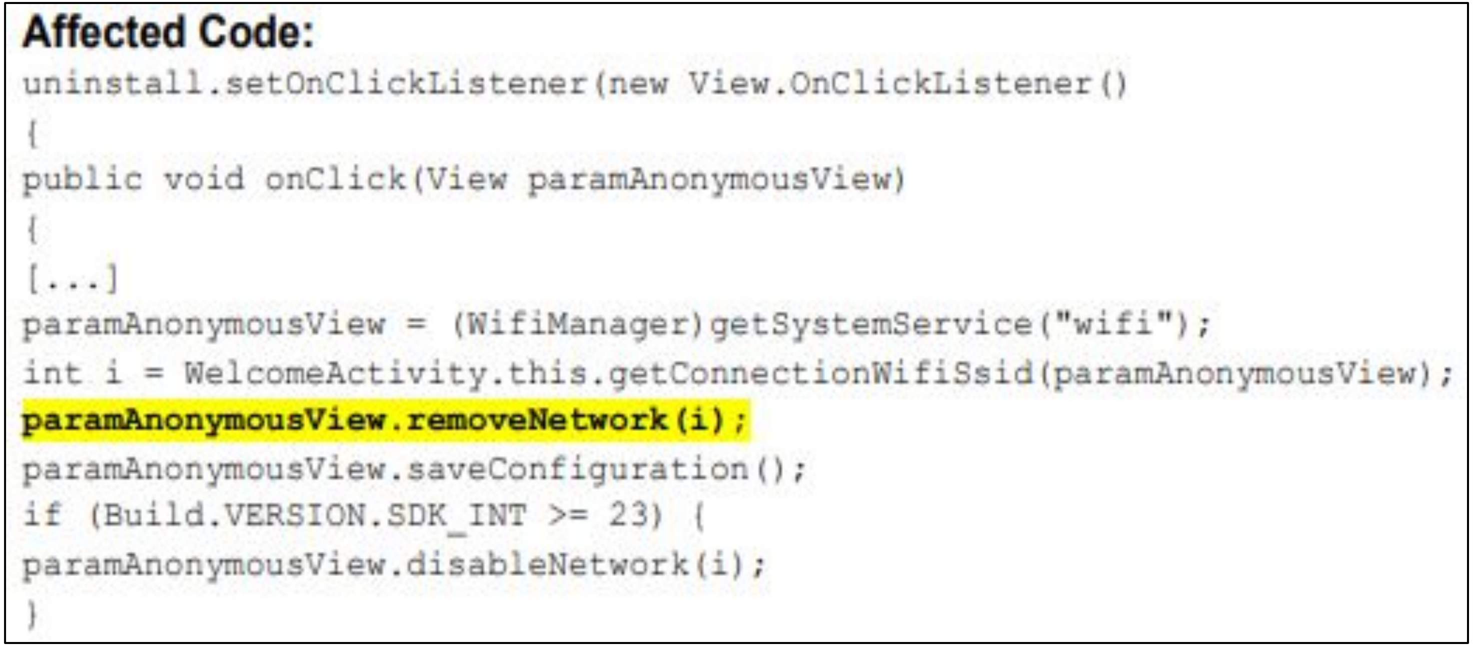
→ There was an indication of a **WiFi server**, but the code **doesn't** contain a **feature** which **supports** an **open hotspot**.

→ Alternatively, the **code** gives **indication** of getting **connected** to a specific **WLAN connection**.

→ The **authentication** details of the **connection** are **dismissed** once the **app** is **uninstalled**.

BXA-01-002: File transmission and protection (Proven)

File:
app/src/main/java/com/fenghuo/qzj/WelcomeActivity.java



Affected Code:

```
uninstall.setOnClickListener(new View.OnClickListener()
{
public void onClick(View paramAnonymousView)
{
[...]
paramAnonymousView = (WifiManager)getSystemService("wifi");
int i = WelcomeActivity.this.getConnectionWifiSsid(paramAnonymousView);
paramAnonymousView.removeNetwork(i);
paramAnonymousView.saveConfiguration();
if (Build.VERSION.SDK_INT >= 23) {
paramAnonymousView.disableNetwork(i);
}
```

BXA-01-002: File transmission and protection (Proven)

However, the app uses the Android AllowAllHostnameVerifier hostname-verifier which could lead to **Man-in-the-Middle** issues.

File:
/com/fenghuo/http/TrustAllSSLSocketFactory.java

Affected Code:

```
setHostnameVerifier(new AllowAllHostnameVerifier());
```

BXA-01-002: File transmission and protection (Proven)

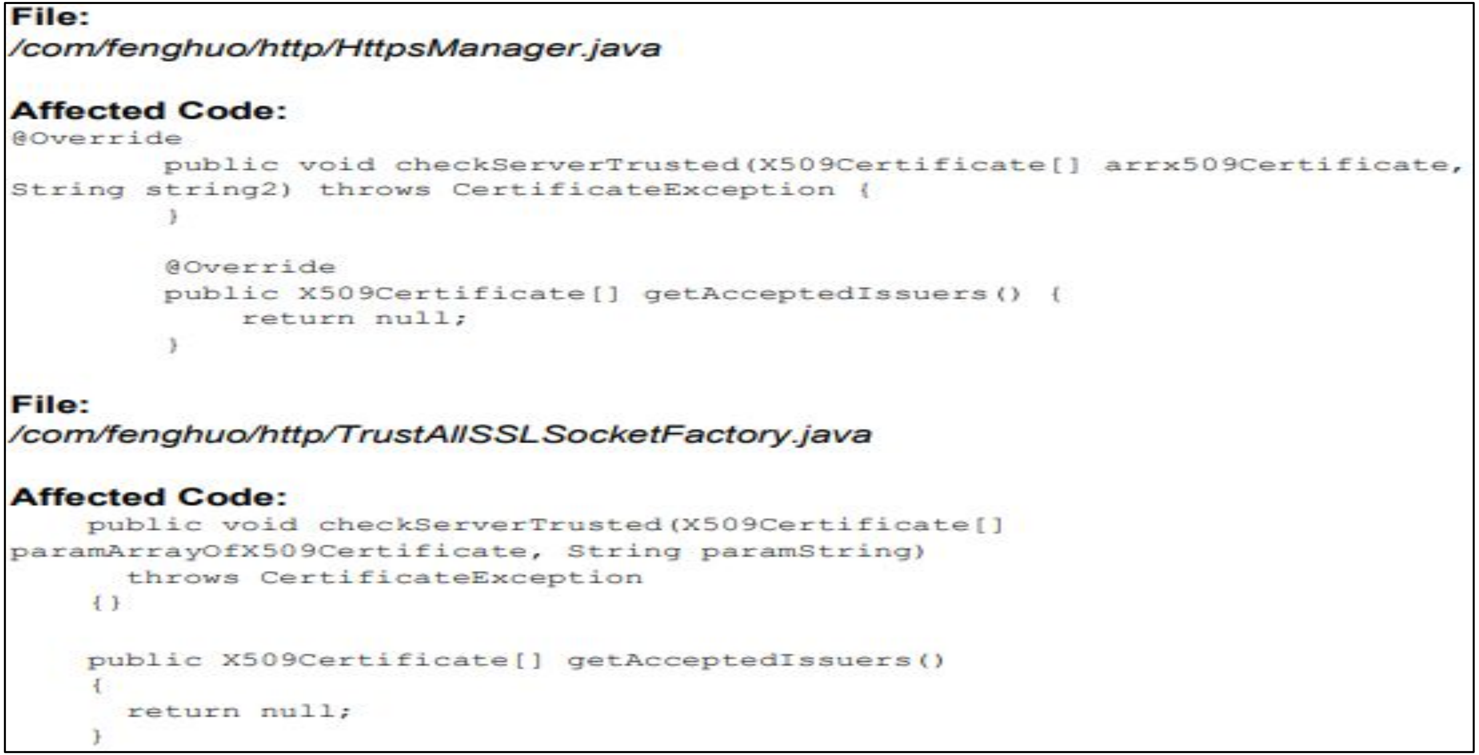
Official Android documentation about the AllowAllHostnameVerifier:
(https://developer.android.com/reference/org/apache/http/conn/ssl/AllowAllHostnameVerifier)

The ALLOW_ALL HostnameVerifier essentially turns hostname verification off.

This implementation is a no-op, and never throws the SSLException.

Furthermore, the app has an empty implementation of **checkServerTrusted** which could also cause **Man-in-the-Middle** problems.

BXA-01-002: File transmission and protection (Proven)



File:
/com/fenghuo/http/HttpsManager.java

Affected Code:

```
@Override
        public void checkServerTrusted(X509Certificate[] arrx509Certificate,
String string2) throws CertificateException {
        }

        @Override
        public X509Certificate[] getAcceptedIssuers() {
            return null;
        }
```

File:
/com/fenghuo/http/TrustAllSSLSocketFactory.java

Affected Code:

```
    public void checkServerTrusted(X509Certificate[]
paramArrayOfX509Certificate, String paramString)
      throws CertificateException
    {}

    public X509Certificate[] getAcceptedIssuers()
    {
      return null;
    }
```

BXA-01-002: File transmission and protection (Proven)

“Is data dumped in the SD Card from where it could be retrieved later without even entering the PIN to unlock the device?”

→ The application stores all scan-related data in its own data directory

→ The only file stored on the SD card is the cjlog.txt file

→ The file is stored in an encrypted form

→ Contains information about when the last scan took place

BXA-01-005 Similarities & differences between BXAQ & IJOP (Assumed)

The following figure shows the icons of BXAQ, JingWang and IJOP in the mentioned order:

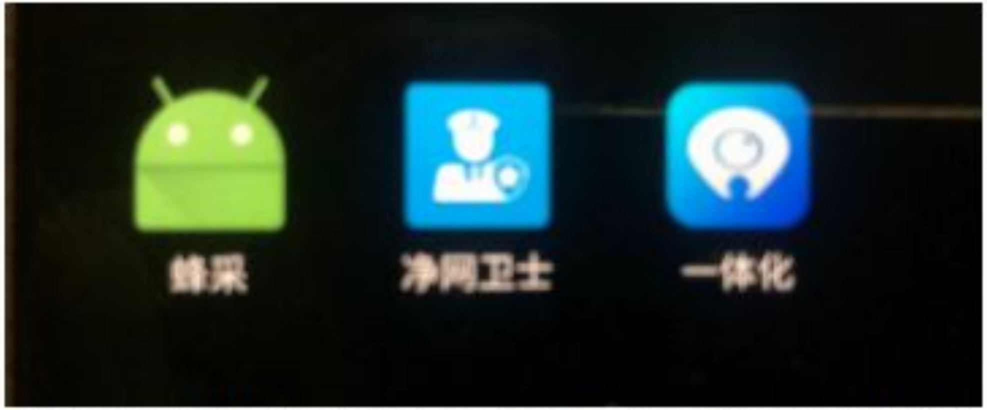
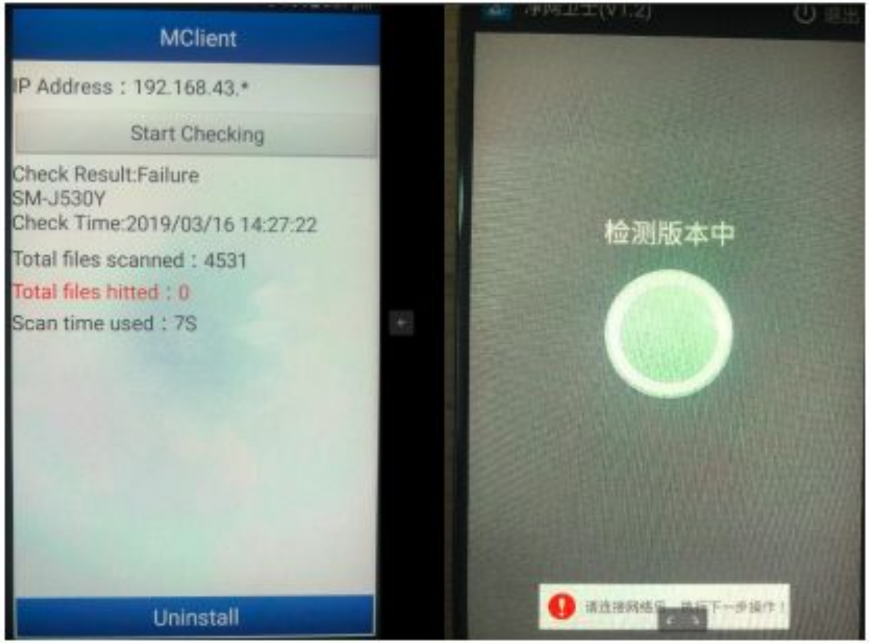


Fig.: The BXAQ app relies on the default Android icon while JingWang and IJOP have official icons.

BXA-01-005 Similarities & differences between BXAQ & IJOP (Assumed)

Different UI designs between **BXAQ** and **JingWang**



IP Address : 192.168.43.*

Check Result:Failure
SM-J530Y
Check Time:2019/03/16 14:27:22

Total files scanned : 4531
Total files hitted : 0
Scan time used : 7S

检测版本中

请连接网络后，执行下一步操作！

BXA-01-005 Similarities & differences between BXAQ & IJOP (Assumed)

The **BXAQ** App serves only one activity with three features:

1. **Scan**
2. **Upload**
3. **Uninstall**

IJOP makes a more official impression as it contains official icons and tries to establish a connection to a **public server**.

Local server contacted by **BXAQ**:

192.168.43.1:8080

Public server contacted by **IJOP**:

47.93.5.238:8081

Analysis

- The **main aspects** that should be **highlighted** among the findings with differently evaluated severities pertain to:

  → Plethora of information-gathering executed by the app (see **BXA-01-001**).

  → Transmission of a lot of data to a local file server (see **BXA-01-002**).

Conclusion

→ The **items** discovered in all of the apps, namely, **IJOP** and **BXAQ** Chinese police apps, were considered to be **potential violation** to the **Human Rights**.

→ Activists along with the HRW **raised concern**

→ The **reports** produced **reflected** highly on the **loopholes** in all of the **systems**.

→ The **team** conducting the **research** and **creation** of the **report** was **unbiased** and completely **technically-driven**.

Want to hack these and more cool apps?

- Global AppSec Amsterdam, EU: 23-25 September 2019
- c0c0n, Kochi, India: 25-26 September 2019
- LASCON, Austin, TX, USA: 22-23 October 2019

Cannot make it?
ping admin@7asecurity.com for training portal access.

7A SECURITY

Q & A

Any questions? :)

> admin@7asecurity.com
>
> @7asecurity
>
> @7a_
>
> @owtfp [OWASP OWTF - owtf.org]

+ 7asecurity.com