Deactivate the Rootkit

Anibal L. Sacco

Alfredo A. Ortega

CORE SECURITY TECHNOLOGIES

History:

2004: The BIOS size of 60% of all notebooks suffered an increase of 25Kb

- Fast forward 5 years, 2009:
  - We were trying to install our own BIOS rootkit (Persistent BIOS Infection Talk, CanSecWest / Syscan)
  - Here is a very quick look of that research:

Persistent BIOS Infection:

We presented a generic technique to modify the BIOS of most common chipsets to insert malicious code in it.

- This technique is applicable to any computer that supports installation of BIOS updates that are not digitally signed using cryptographically strong methods.

In the news:

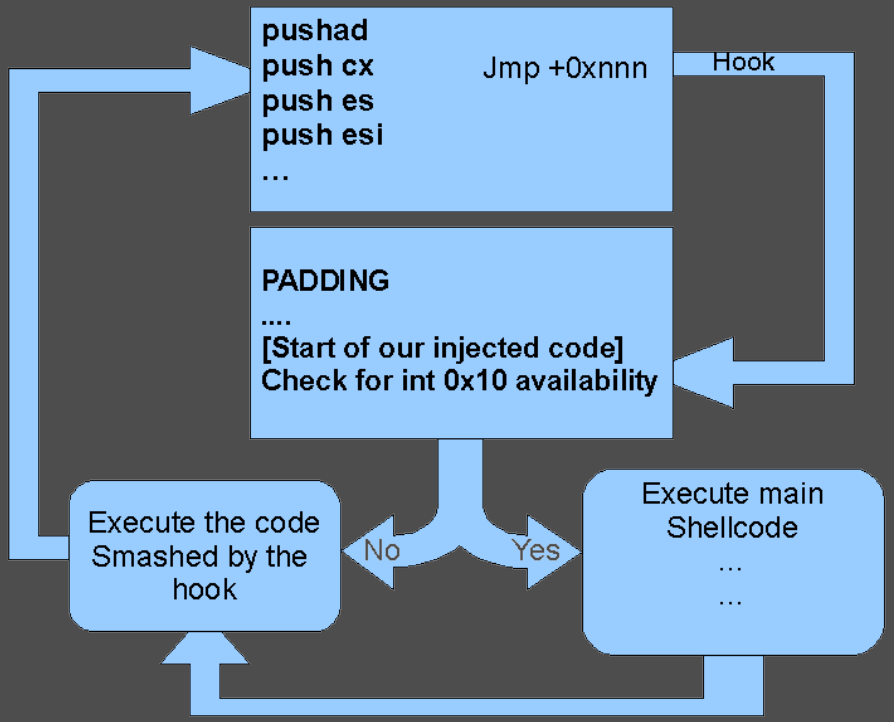
"Researchers unveil persistent BIOS attack" **-** securityfocus.com

"Researchers Demo BIOS attack that survives disk wipes" - slashdot.org

"This is BS, it was discovered/created 20 years ago" **-** KCOp3

Persistent BIOS Infection:

- The only caveat is to know where to patch. We chose the ‘decompression routine’, because its uncompressed and easily findable using pattern matching.



Persistent BIOS Infection:

- We can resume it in three easy steps:
  - 1) Dump BIOS firmware using flashrom
  - 2) Patch and compensate
  - 3) Re-flash

‘pre’ Demo Time

We will show three different demonstrations of malicious code injected on the BIOS:

- Windows code injection (VMWARE)
- OpenBSD file attributes modification
- Real hardware demo

Deactivate the Rootkit:

And... What about notebooks?

- When we started to look into notebook BIOSes...
- We found that there was something already there!

What is the rootkit?

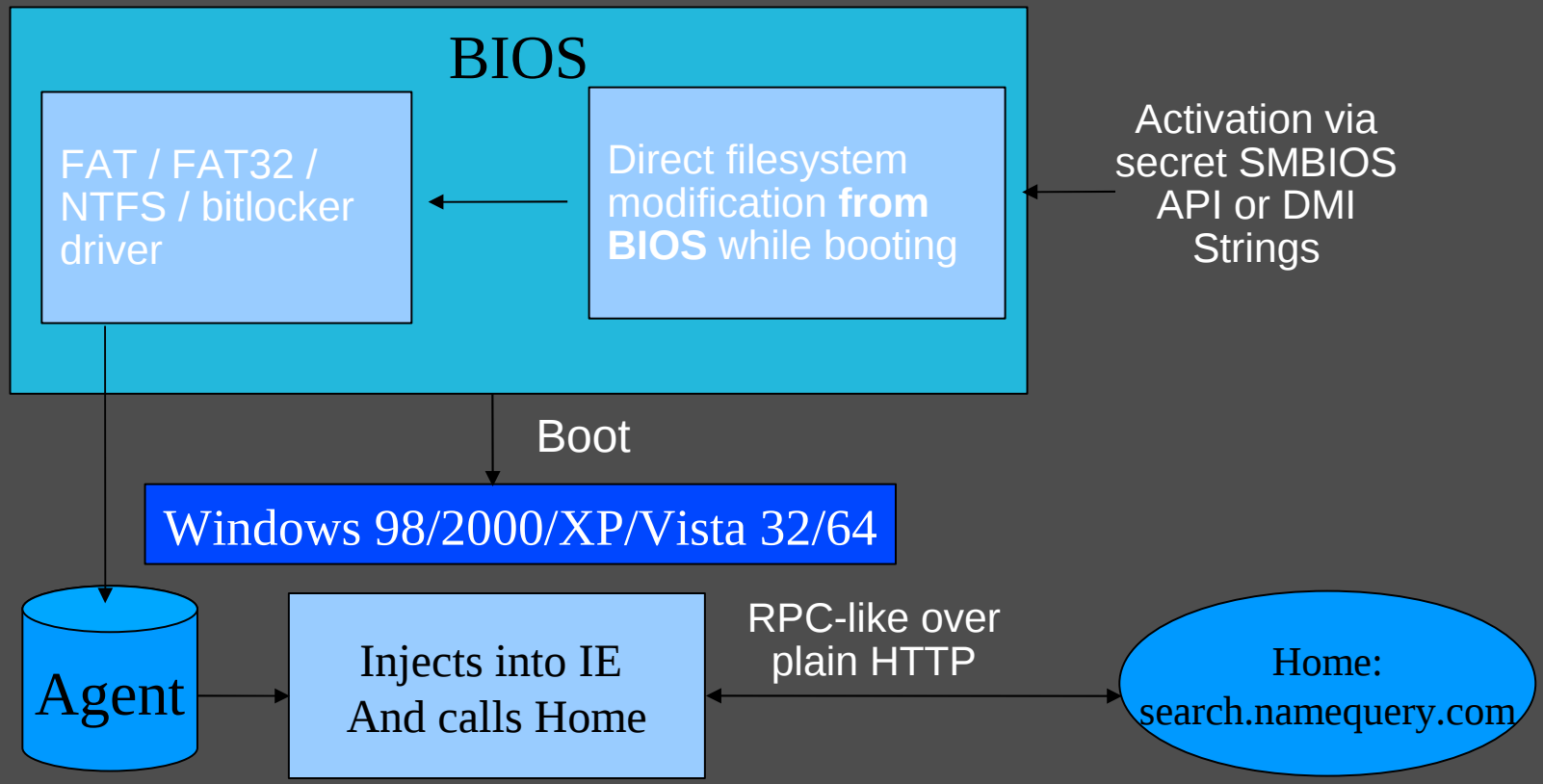
- Absolute Corp. Computrace, Anti-theft agent
- Option ROM Embedded in Phoenix BIOS
- Agreements with law enforcement agencies.
- Inside notebooks from HP, Dell, Lenovo, Toshiba, Gateway, Asus, Panasonic, and more.

Option ROM header:

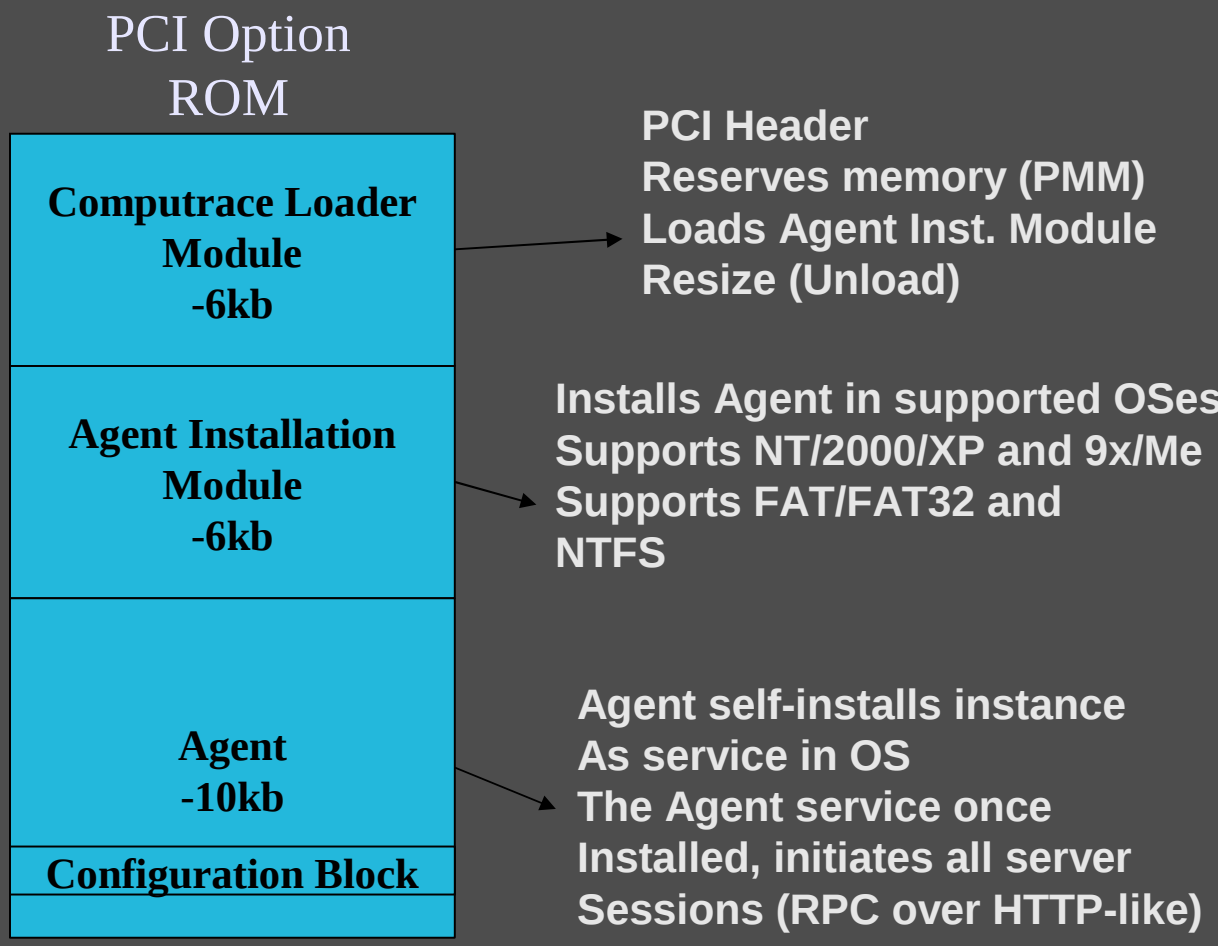
```
00000000  55 aa 2a eb 15 43 6f 6d  70 75 54 72 61 63 65 20  |U.*..CompuTrace |
00000010  56 38 30 2e 38 36 36 78  1d 00 e9 5c 01 50 43 49  |V80.866x...\.PCI|
00000020  52 17 19 34 12 00 00 18  00 00 06 00 00 2a 00 00  |R..4.........*..|
```

Basic Inner workings:

- See patent application US 2006/0272020 A1



Basic Inner workings:



Problems found:

- Huge privacy risk (bad/no authentication)
- Anyone could activate it with enough privileges
- Anyone can change the configuration
- Anyone can de-activate it (at least in certain known cases)
- Whitelisted by AV (potentally indetectable)

More problems found:

- Use of URL instead of IP (hosts redirection)
- Configuration block modification:

Demo if there is time...

```
Configuration block XOR 0xB5:
00000000  b1 b7 b5 b5 35 ab b1 b4  b5 f5 b4 aa b1 b5 b5 b5  |....5...........|
00000010  b5 a5 bf 41 41 30 49 4e  30 30 30 30 30 95 b1 1f  |...AA0IN00000...|
00000020  ee 30 86 a0 b1 8b b5 35  b5 ac ae 4a 4a 4a 4a 4a  |.0.....5...JJJJJ|
00000030  4a 4a 4a 4a 4a 4a 4a 4a  4a 4a 4a 4a 4a 4a 4a 4a  |JJJJJJJJJJJJJJJJ|
00000040  4a 4a 4a 4a 4a 4a af b4  35 ae b3 b5 b5 b5 b5 b5  |JJJJJJ..5.......|
00000050  b5 a8 b7 b5 b5 f3 b3 b5  b5 b5 b5 b5 b5 f2 b3 b5  |................|
00000060  b5 b5 b5 b5 b5 fd af 00  50 d1 35 71 17 73 65 61  |........P.5q.sea|
00000070  72 63 68 2e 6e 61 6d 65  71 75 65 72 79 2e 63 6f  |rch.namequery.co|
00000080  6d bf b7 b2 a5 b3 b3 ac  35 b4 b4 b5 b5 b2 b3 b5  |m.......5.......|
00000090  b5 b5 b5 b5 4a 98 b4 0d  98 b4 0d 9e b1 41 54 44  |....J........ATD|
000000a0  54 81 b7 38 2c 80 b7 39  2c 82 b2 39 2c 39 31 38  |T..8,..9,..9,918|
```

Computrace network dump

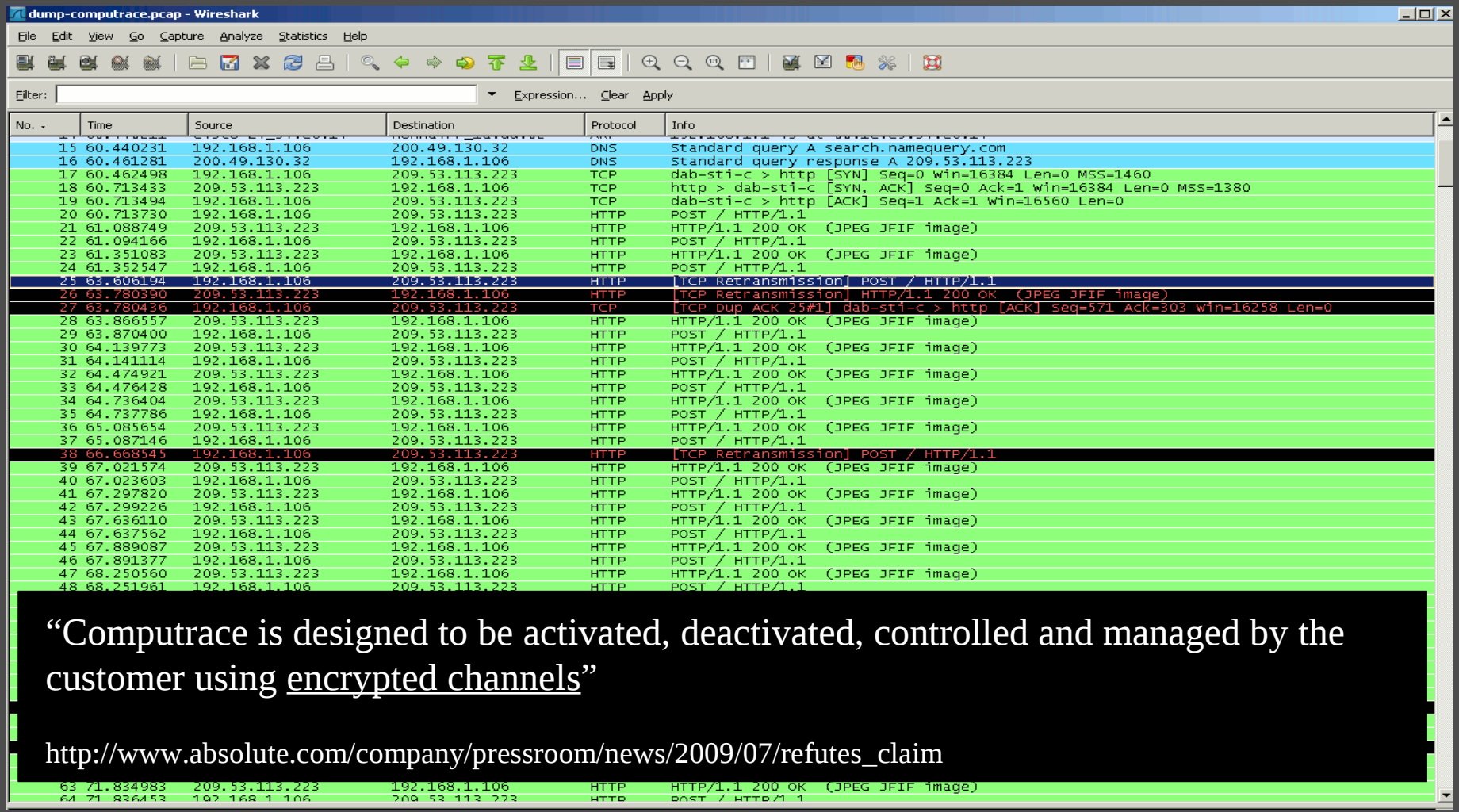
dump-computrace.pcap - Wireshark

File Edit View Go Capture Analyze Statistics Help

Filter: Expression... Clear Apply

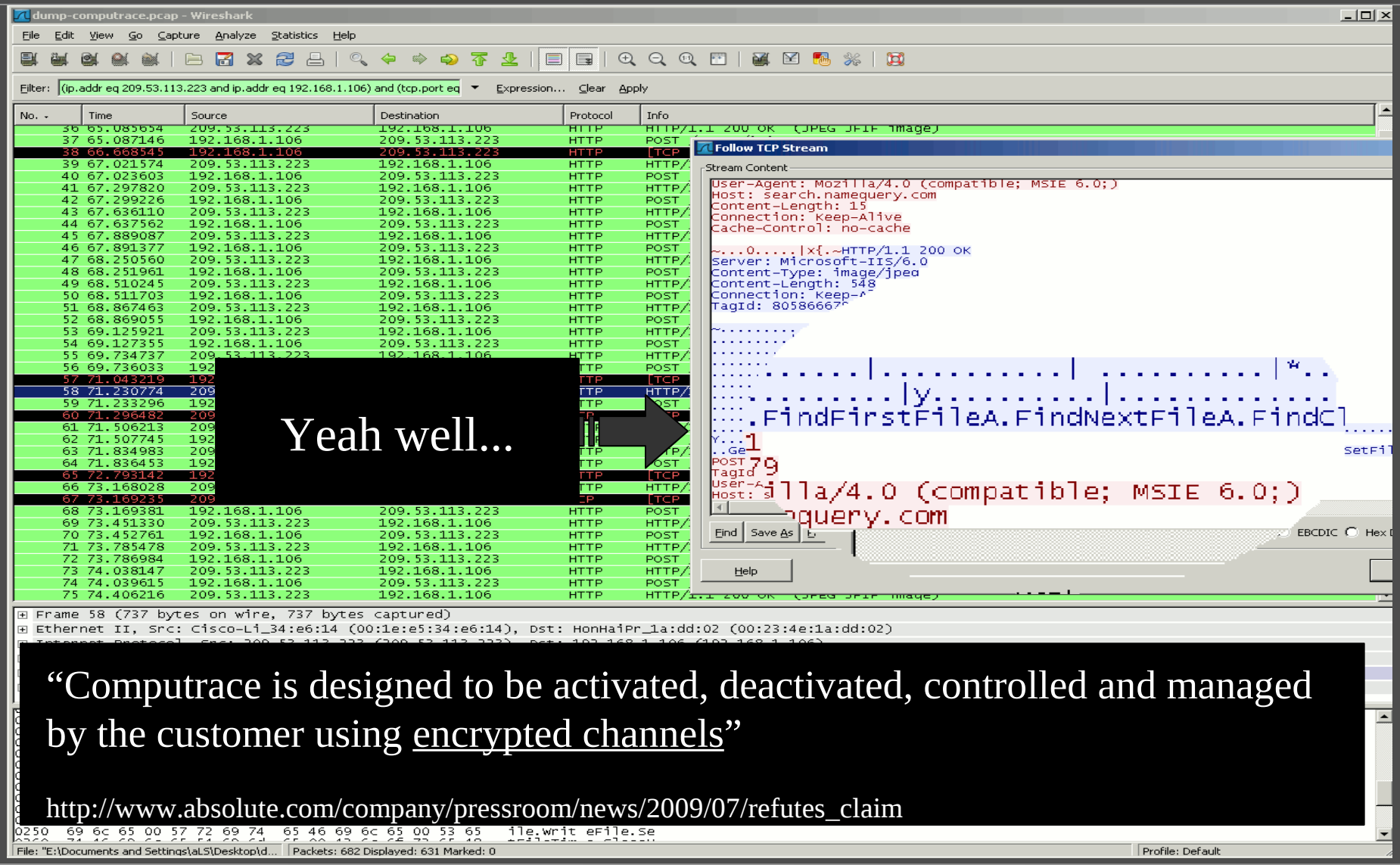
No.	Time	Source	Destination	Protocol	Info
15	60.440231	192.168.1.106	200.49.130.32	DNS	Standard query A search.namequery.com
16	60.461281	200.49.130.32	192.168.1.106	DNS	Standard query response A 209.53.113.223
17	60.462498	192.168.1.106	209.53.113.223	TCP	dab-sti-c > http [SYN] Seq=0 Win=16384 Len=0 MSS=1460
18	60.713433	209.53.113.223	192.168.1.106	TCP	http > dab-sti-c [SYN, ACK] Seq=0 Ack=1 Win=16384 Len=0 MSS=1380
19	60.713494	192.168.1.106	209.53.113.223	TCP	dab-sti-c > http [ACK] Seq=1 Ack=1 Win=16560 Len=0
20	60.713730	192.168.1.106	209.53.113.223	HTTP	POST / HTTP/1.1
21	61.088749	209.53.113.223	192.168.1.106	HTTP	HTTP/1.1 200 OK (JPEG JFIF image)
22	61.094166	192.168.1.106	209.53.113.223	HTTP	POST / HTTP/1.1
23	61.351083	209.53.113.223	192.168.1.106	HTTP	HTTP/1.1 200 OK (JPEG JFIF image)
24	61.352547	192.168.1.106	209.53.113.223	HTTP	POST / HTTP/1.1
25	63.606194	192.168.1.106	209.53.113.223	HTTP	[TCP Retransmission] POST / HTTP/1.1
26	63.780390	209.53.113.223	192.168.1.106	HTTP	[TCP Retransmission] HTTP/1.1 200 OK (JPEG JFIF image)
27	63.780436	192.168.1.106	209.53.113.223	TCP	[TCP Dup ACK 25#1] dab-sti-c > http [ACK] Seq=571 Ack=303 Win=16258 Len=0
28	63.866557	209.53.113.223	192.168.1.106	HTTP	HTTP/1.1 200 OK (JPEG JFIF image)
29	63.870400	192.168.1.106	209.53.113.223	HTTP	POST / HTTP/1.1
30	64.139773	209.53.113.223	192.168.1.106	HTTP	HTTP/1.1 200 OK (JPEG JFIF image)
31	64.141114	192.168.1.106	209.53.113.223	HTTP	POST / HTTP/1.1
32	64.474921	209.53.113.223	192.168.1.106	HTTP	HTTP/1.1 200 OK (JPEG JFIF image)
33	64.476428	192.168.1.106	209.53.113.223	HTTP	POST / HTTP/1.1
34	64.736404	209.53.113.223	192.168.1.106	HTTP	HTTP/1.1 200 OK (JPEG JFIF image)
35	64.737786	192.168.1.106	209.53.113.223	HTTP	POST / HTTP/1.1
36	65.085654	209.53.113.223	192.168.1.106	HTTP	HTTP/1.1 200 OK (JPEG JFIF image)
37	65.087146	192.168.1.106	209.53.113.223	HTTP	POST / HTTP/1.1
38	66.668545	192.168.1.106	209.53.113.223	HTTP	[TCP Retransmission] POST / HTTP/1.1
39	67.021574	209.53.113.223	192.168.1.106	HTTP	HTTP/1.1 200 OK (JPEG JFIF image)
40	67.023603	192.168.1.106	209.53.113.223	HTTP	POST / HTTP/1.1
41	67.297820	209.53.113.223	192.168.1.106	HTTP	HTTP/1.1 200 OK (JPEG JFIF image)
42	67.299226	192.168.1.106	209.53.113.223	HTTP	POST / HTTP/1.1
43	67.636110	209.53.113.223	192.168.1.106	HTTP	HTTP/1.1 200 OK (JPEG JFIF image)
44	67.637562	192.168.1.106	209.53.113.223	HTTP	POST / HTTP/1.1
45	67.889087	209.53.113.223	192.168.1.106	HTTP	HTTP/1.1 200 OK (JPEG JFIF image)
46	67.891377	192.168.1.106	209.53.113.223	HTTP	POST / HTTP/1.1
47	68.250560	209.53.113.223	192.168.1.106	HTTP	HTTP/1.1 200 OK (JPEG JFIF image)
48	68.251961	192.168.1.106	209.53.113.223	HTTP	POST / HTTP/1.1
49	68.510245	209.53.113.223	192.168.1.106	HTTP	HTTP/1.1 200 OK (JPEG JFIF image)
50	68.511703	192.168.1.106	209.53.113.223	HTTP	POST / HTTP/1.1
51	68.867463	209.53.113.223	192.168.1.106	HTTP	HTTP/1.1 200 OK (JPEG JFIF image)
52	68.869055	192.168.1.106	209.53.113.223	HTTP	POST / HTTP/1.1
53	69.125921	209.53.113.223	192.168.1.106	HTTP	HTTP/1.1 200 OK (JPEG JFIF image)
54	69.127355	192.168.1.106	209.53.113.223	HTTP	POST / HTTP/1.1
55	69.734737	209.53.113.223	192.168.1.106	HTTP	HTTP/1.1 200 OK (JPEG JFIF image)
56	69.736033	192.168.1.106	209.53.113.223	HTTP	POST / HTTP/1.1
57	71.043219	192.168.1.106	209.53.113.223	HTTP	[TCP Retransmission] POST / HTTP/1.1
58	71.230774	209.53.113.223	192.168.1.106	HTTP	HTTP/1.1 200 OK (JPEG JFIF image)
59	71.233296	192.168.1.106	209.53.113.223	HTTP	POST / HTTP/1.1
60	71.296482	209.53.113.223	192.168.1.106	TCP	[TCP Dup ACK 58#1] http > dab-sti-c [PSH, ACK] Seq=3101 Ack=3351 Win=65139 Len=0
61	71.506213	209.53.113.223	192.168.1.106	HTTP	HTTP/1.1 200 OK (JPEG JFIF image)
62	71.507745	192.168.1.106	209.53.113.223	HTTP	POST / HTTP/1.1
63	71.834983	209.53.113.223	192.168.1.106	HTTP	HTTP/1.1 200 OK (JPEG JFIF image)

Computrace network dump



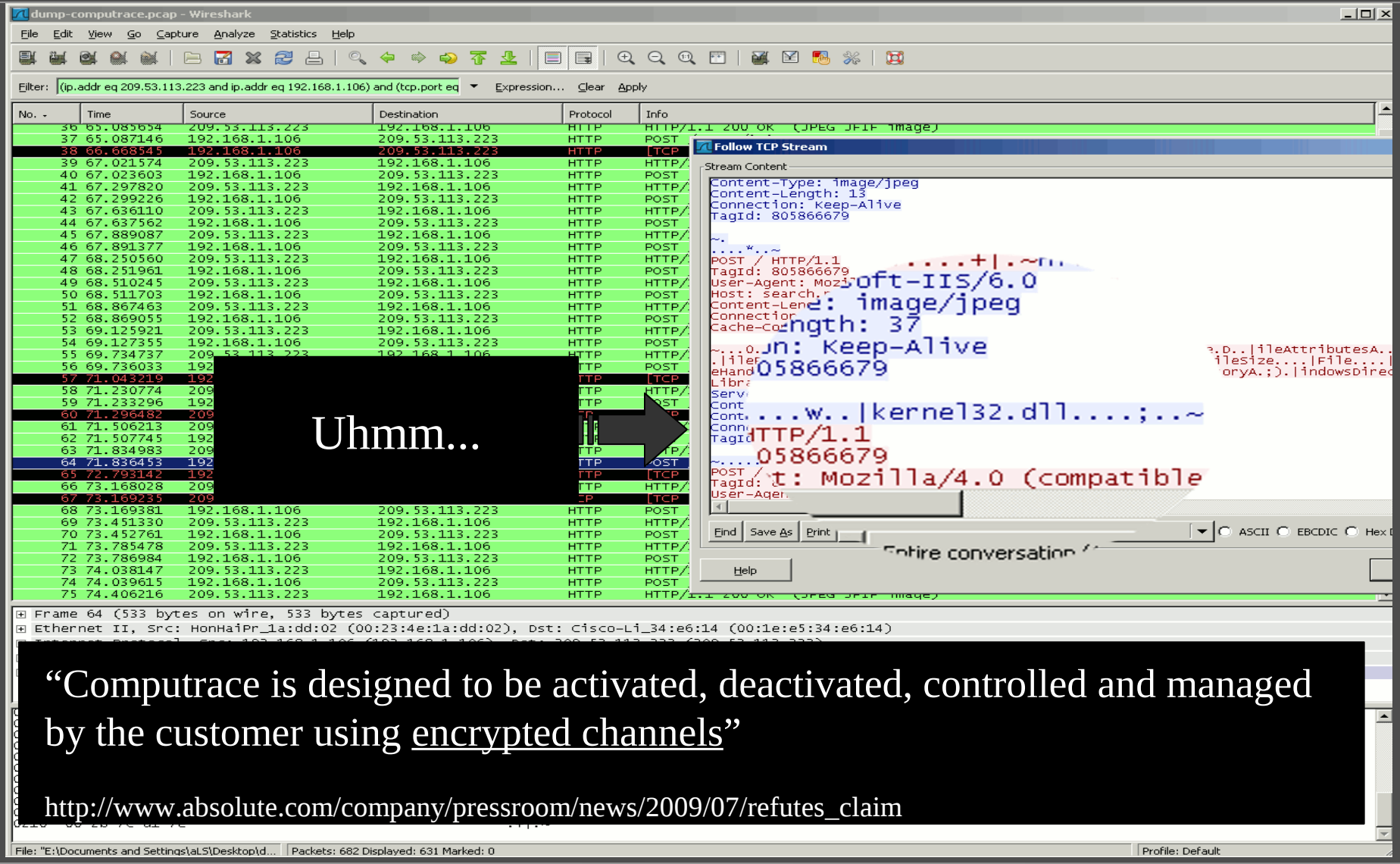
"Computrace is designed to be activated, deactivated, controlled and managed by the customer using encrypted channels"

http://www.absolute.com/company/pressroom/news/2009/07/refutes_claim



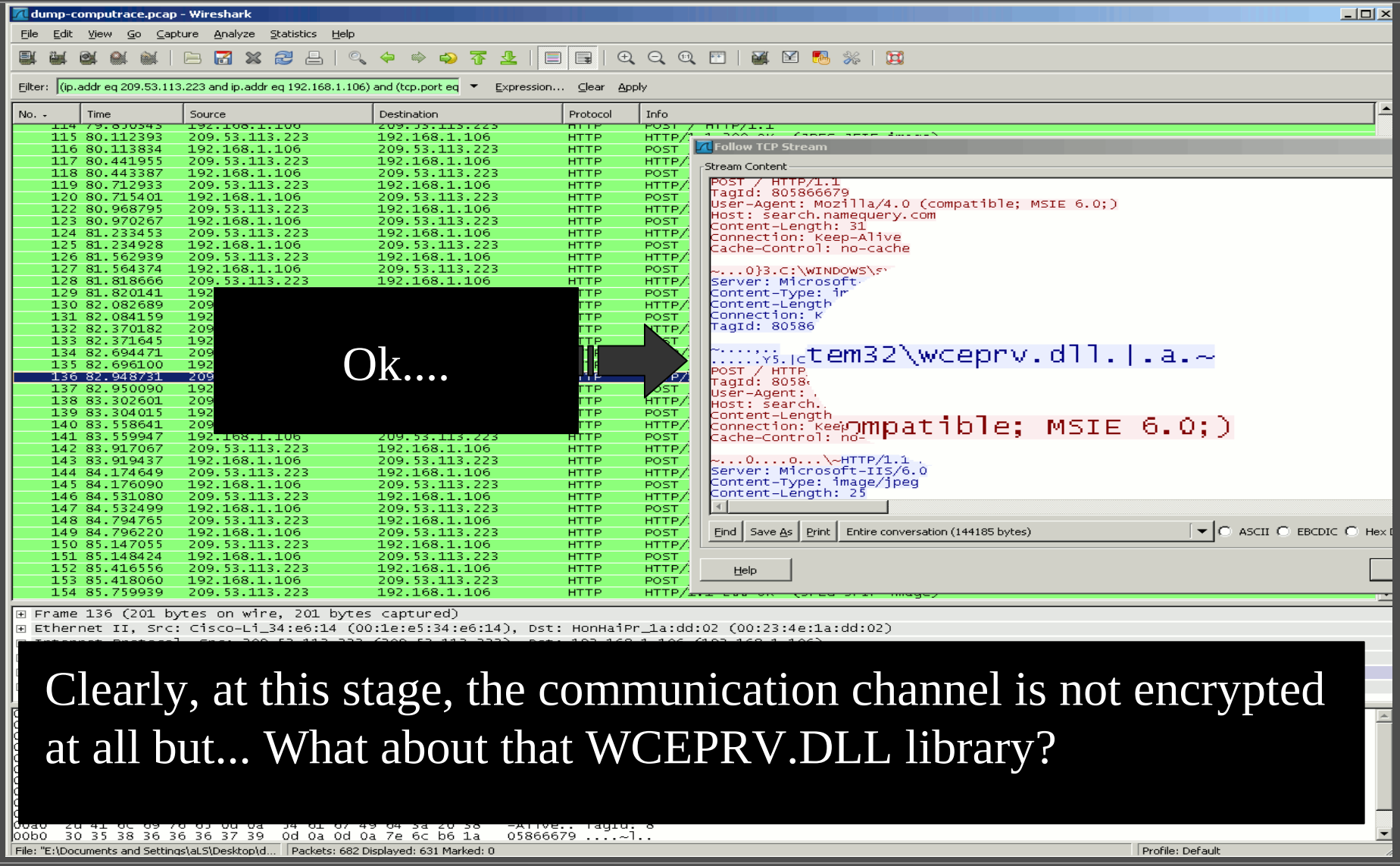
"Computrace is designed to be activated, deactivated, controlled and managed by the customer using encrypted channels"

http://www.absolute.com/company/pressroom/news/2009/07/refutes_claim



"Computrace is designed to be activated, deactivated, controlled and managed by the customer using encrypted channels"

http://www.absolute.com/company/pressroom/news/2009/07/refutes_claim



Clearly, at this stage, the communication channel is not encrypted at all but... What about that WCEPRV.DLL library?

Encrypted channel: Analysis

- WCEPRV.DLL downloaded on the first run.
- Encryption algorithm: RC4 stream cipher
- Session key generated on the client
- Key Transmitted on plaintext!

Action	Packets
Checking encryption DLL timestamp, call Kernel32 FindFirstFile function on client	66–71
Call Kernel32 FindClose function on client	72–75
Load WCEPRV.DLL on client	76–79
Set encryption communication, read old transmit address	80, 81
Read old receive address	82, 83
Call WceSet on client	84–91
Setup encryption key on client, call WceStartup	92–99
Get WceSend procedure address	100–103
Get WceRecv procedure address	104–107
Set new transmit address	108, 109
Set new receive address	110, 111
Enable encryption on client, call WceEnable	112–119
Check transmit (WceSend) procedure address	120–123

Plaintext key exchange

And one more thing... Stub agent: Unauthenticated BIOS code execution

Second Stage (AIM) loader, Stub Agent (DELL Vostro 1510 Computrace V 70.785)

```
seg000:01CF sub_1CF         proc near               ; CODE XREF: sub_27F+20↓p
seg000:01CF                 push    cx
seg000:01D0                 pop     es
seg000:01D1                 assume es:nothing
seg000:01D1                 mov     si, 0BFh ; '+'
seg000:01D4                 mov     [si+6], cx
seg000:01D7                 mov     dl, 80h ; 'Ç'
seg000:01D9                 mov     ah, 42h ; 'B'
seg000:01DB                 int     13h             ; DISK -
seg000:01DD                 push    es
seg000:01DE                 pop     ds
seg000:01DF                 jnb     short loc_1E2
seg000:01E1
seg000:01E1 locret_1E1:                             ; CODE XREF: sub_1CF+1B↓j
seg000:01E1                                         ; sub_1CF+72↓j
seg000:01E1                 retn
seg000:01E2 ; ---------------------------------------------------------------------------
seg000:01E2
seg000:01E2 loc_1E2:                                ; CODE XREF: sub_1CF+10↑j
seg000:01E2                 xor     ecx, ecx
seg000:01E5
seg000:01E5 loc_1E5:                                ; CODE XREF: sub_1CF+2D↓j
seg000:01E5                                         ; sub_1CF+33↓j ...
seg000:01E5                 inc     cl
seg000:01E7                 cmp     cl, 3Eh ; '>'
seg000:01EA                 ja      short locret_1E1
seg000:01EC                 mov     ebx, ecx
seg000:01EF                 shl     bx, 9
seg000:01F2                 lea     bx, [bx+7E00h]
seg000:01F6                 movzx   eax, byte ptr [bx]
seg000:01FA                 cmp     al, 3Eh ; '>'
seg000:01FC                 ja      short loc_1E5
seg000:01FE
seg000:01FE loc_1FE:                                ; CODE XREF: sub_27F+33↓J
seg000:01FE                                         ; DATA XREF: sub_27F+30↓o
seg000:01FE                 cmp     eax, [bx+4]
seg000:0202                 jbe     short loc_1E5
seg000:0204                 cmp     ecx, [ebx+eax*4]
seg000:0209                 jnz     short loc_1E5
seg000:020B                 cmp     eax, [ebx+eax*4+4]
seg000:0211                 jnz     short loc_1E5
seg000:0213                 mov     dx, [bx+2]
seg000:0216                 movzx   ebp, byte ptr [bx+1]
seg000:021B                 mov     si, bp
seg000:021D                 lea     bp, [ebx+ebp*4+4]
seg000:0222                 lea     bx, [ebx+eax*4-4]
```

```
seg000:0227                 mov     di, bx
seg000:0229                 sub     di, bp
seg000:022B                 shr     di, 2
seg000:022E                 add     di, si
seg000:0230                 inc     di
seg000:0231                 inc     di
seg000:0232                 cmp     di, ax
seg000:0234                 jnz     short loc_1E5
seg000:0236                 shl     edx, 10h
seg000:023A
seg000:023A loc_23A:                                ; CODE XREF: sub_1CF+A6↓j
seg000:023A                 mov     esi, [bx]
seg000:023D                 cmp     esi, 3Eh ; '>'
seg000:0241                 ja      short locret_1E1
seg000:0243                 shl     si, 9
seg000:0246                 lea     si, [si+7E00h]
seg000:024A                 mov     di, bx
seg000:024C                 sub     di, bp
seg000:024E                 shr     di, 2
seg000:0251                 dec     di
seg000:0252                 shl     di, 9
seg000:0255                 lea     di, [di+100h]
seg000:0259                 mov     cx, 200h
seg000:025C
seg000:025C loc_25C:                                ; CODE XREF: sub_1CF+9F↓j
seg000:025C                 lodsb
seg000:025D                 xor     dh, al
seg000:025F                 mov     ah, 8
seg000:0261
seg000:0261 loc_261:                                ; CODE XREF: sub_1CF+9C↓j
seg000:0261                 shl     dx, 1
seg000:0263                 jnb     short loc_269
seg000:0265                 xor     dx, 1021h
seg000:0269
seg000:0269 loc_269:                                ; CODE XREF: sub_1CF+94↑j
seg000:0269                 dec     ah
seg000:026B                 jnz     short loc_261
seg000:026D                 stosb
seg000:026E                 loop    loc_25C
seg000:0270                 sub     bx, 4
seg000:0273                 cmp     bx, bp
seg000:0275                 jnz     short loc_23A
seg000:0277                 shld    eax, edx, 10h
seg000:027C                 sub     ax, dx
seg000:027E                 retn
seg000:027E sub_1CF         endp
```

Detecting the Rootkit Agent

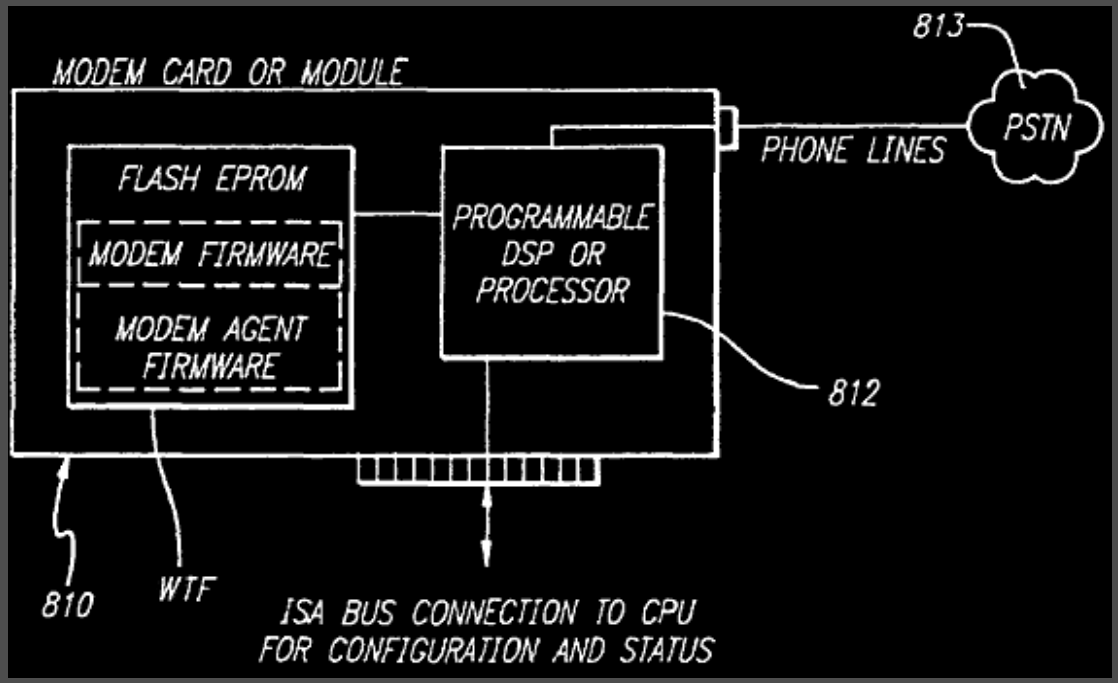
- A single file to look for:
  - system32\rpcnet.exe (Normal Agent)
  - system32\rpcnetp.exe (BIOS Persistent Agent)
- A service called ”Remote Procedure Call (RPC) Net” with no description
- Outgoing connections to search.namequery.com (209.53.113.223)
- Our Computrace Option Rom Dumper tool

Deactivating:

- Easiest way: hosts file redirection
- Modifying BIOS (only **unsigned BIOS!**)
- Modifying configuration block (Registry, hard-disk, etc.)
- Modifying nvram, then full HD Wipe.

The Past:

- US 6,300,863 B1 Pat. Figure 8A
- Filed Mar 24 **1998**, Absolute Corporation
- Agent inside modem Option ROM
- Support for DOS Backdooring



See “Implementing and Detecting a PCI Rootkit”, Heasman, BlackHat **2007**

The Future:

- Phoenix Failsafe:
  - Inside SMM, sounds familiar?
  - Always-on OS-independent, Wifi and GPS tracking
  - It has “safe” in the name instead of “trace”
- Intel Anti-theft technology:
  - vPro technology
  - Using AMT secondary processor
  - Works even with the notebook turned off!
- Other security aplications residing in BIOS

Strong authentication: *“Trust us, is for your own protection”***.**

This is only the begginning

- More research is needed in this area!
- CoreBoot (LinuxBIOS) project, is computrace-free
- Questions?
- Thanks! Now if you'll just look into the light: