Hardware Backdooring is practical

Jonathan Brossard (Toucan System)

toucan system

IT serenity®

SHAKACON

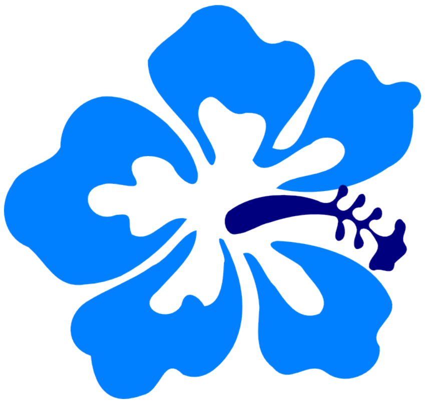
SUN, SURF, & C SHELLS™

DISCLAIMER

- We are not « terrorists ». We won't release our PoC backdoor.
- The x86 architecture is plagued by legacy. Governments know. The rest of the industry : not so much.
- There is a need to discuss the problems in order to find solutions...
- This is belived to be order of magnitudes better over existing backdoors/malware

PARENTAL
ADVISORY
EXPLICIT CONTENT

Agenda



- Motivation : state level backdooring ?
- Coreboot & x86 architecture
- State of the art in rootkitting, romkitting
- Introducing Rakshasa
- Epic evil remote carnal pwnage (of death)
- Why cryptography (Truecrypt/Bitlocker/TPM) won't save us...
- Backdooring like a state

Could a state (eg : China) backdoor all new computers on earth ?

Occupying the Information High Ground:

Chinese Capabilities for Computer Network Operations and Cyber Espionage

This close relationship between some of China's—and the world's—largest telecommunications hardware manufacturers creates a potential vector for state sponsored or state directed penetrations of the supply chains for microelectronics supporting U.S. military, civilian government, and high value civilian industry such as defense and telecommunications, though no evidence for such a connection is publicly available.

Prepared for the U.S.-China Economic and
Security Review Commission
by Northrop Grumman Corp

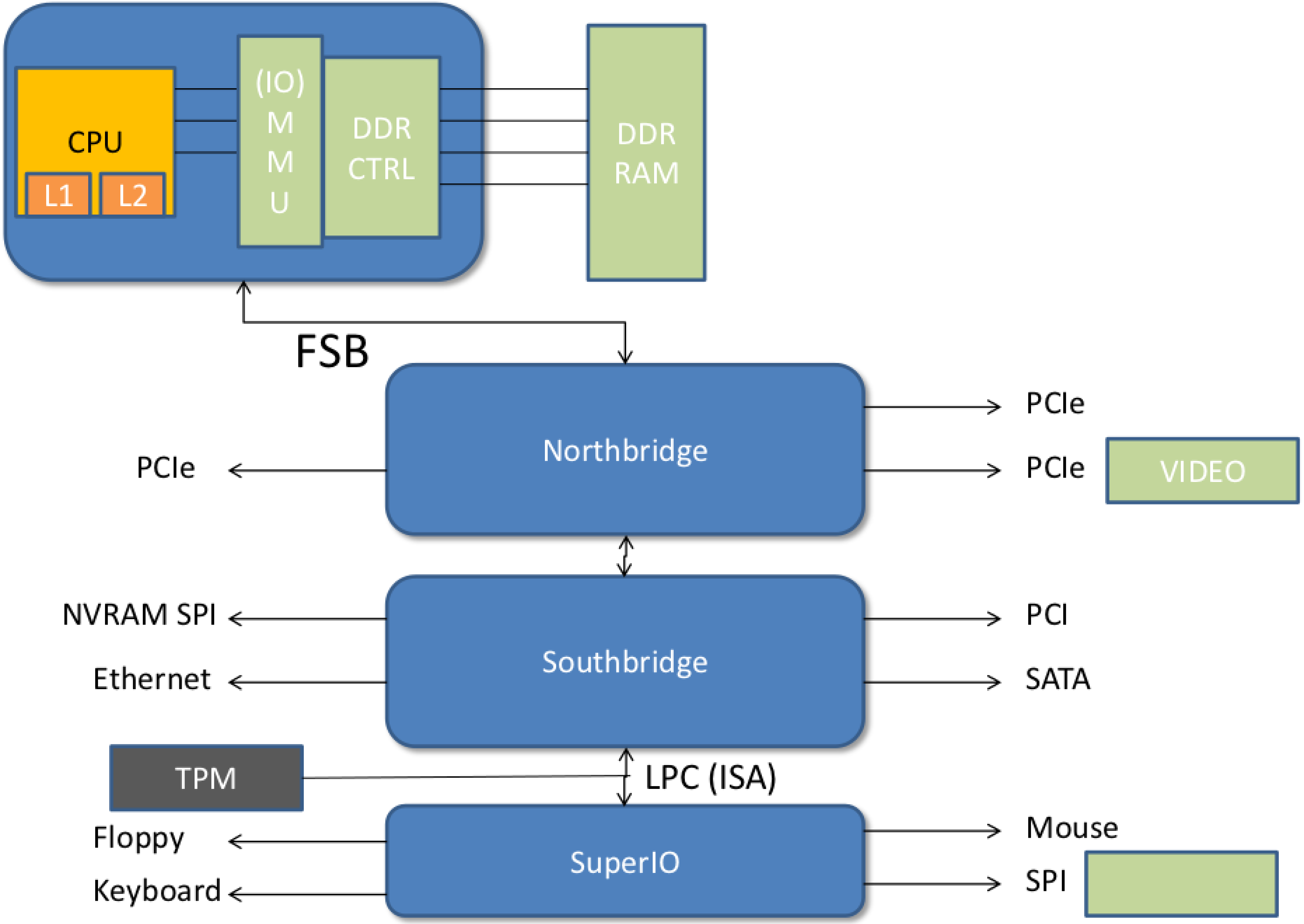
U.S.-China Economic and Security Review Commission

Bryan Krekel
Patton Adams
George Bakos

March 7, 2012

NORTHROP GRUMMAN

A bit of x86 architecture


CPU
L1
L2
(IO)
M
M
U
DDR
CTRL
DDR
RAM
FSB
Northbridge
PCIe
PCIe
PCIe
VIDEO
NVRAM SPI
Ethernet
Southbridge
PCI
SATA
TPM
LPC (ISA)
Floppy
Keyboard
SuperIO
Mouse
SPI

State of the art, previous work

Previous work

- Early 80s : Brain virus, targets the MBR
- 80s, 90s : thousands of such viruses
- 2007, John Heasman (NGS Software) Blackhat US: backdoor EFI bootloader
- 2009, Anibal Saco and Alfredo Ortega (Core security), CanSecWest : patch/flash a Pheonix-Award Bios
- 2009, Kleissner, Blackhat US : Stoned bootkit. Bootkit Windows, Truecrypt. Load arbitrary unsigned kernel module.
- 2010, Kumar and Kumar (HITB Malaysia) : vbootkit bootkitting of Windows 7.
- Piotr Bania, Konboot : bootkit any Windows (32/64b)
- 2012 : Snare (Syscan) : EFI rootkitting

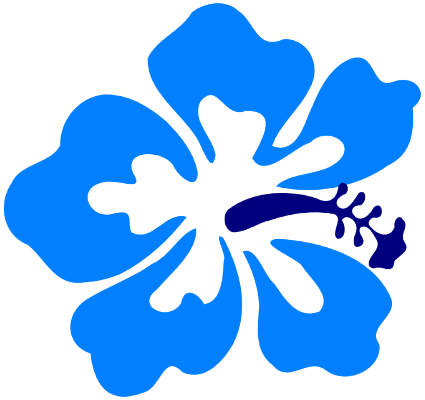
DEMO : Bootkitting Windows

Introducing Rakshasa

Goals : create the perfect backdoor

- Persistant
- Stealth (virtually undetectable)
- Portable (OS independant)
- Remote access, remote updates
- State level quality : plausible deniability, non attribution
- Cross network perimeters (firewalls...)
- Redundancy

Rakshasa : design



- Core components :

  Coreboot
  SeaBios
  iPXE
  payloads

  Built on top of free software : portability, non attribution, cheap dev (~4 weeks of work), really hard to detect (without false positives).
- Payload : Reverse Engineered/Refactored konboot payload (2 days of work).

Rakshasa

- Flash the BIOS (Coreboot + PCI roms such as iPXE)
- Flash the network card or any other PCI device (redundancy)
- Boot a payload over the network (bootkit)
- Boot a payload over wifi/wimax (breach the network perimeter, bypasses network detection, I(P|D)S)
- Remotely reflash the BIOS/network card if necessary

Rakshasa : embedded features

- Remove NX bit (from BIOS or PCI)
  → executable heap/stack.
- Remove CPU updates (microcodes)
- Remove anti-SMM protections (=>local root)

  → Permantent lowering of the security level on any OS. Welcome back to the security level of 1999.
  → Persistant, even if HD is remove/restored.

Optionally : Disable ASLR (bootkitting) by patching the seed in kernel land on the fly on Windows.

Rakshasa : remote payload

- Bootkit future OSes
- Update/remove/reflash firmwares (PCI, BIOS)
- Currently capable of Bootkitting any version of Windows (32b/64b)
- Use a minimal linux initrd in case we want to mount/modify the filesystem (/etc/shadow on any UNIX like, add new account with ADMIN privileges on Windows, enable remote desktop – possibly enable dual remote desktop on Windows XP Pro by patching 2 dlls...)

Rakshasa : stealthness

- We don't touch the disk. 0 evidence on the filesystem.
- We can remotely boot from an alternate payload or even OS : fake Truecrypt/Bitlocker prompt !
- Optionally boot from a WIFI/WMAX stack : 0 network evidence on the LAN.
- Fake BIOS menus if necessary. We use an embedded CMOS image. We can use the real CMOS nvram to store encryption keys/backdoor states between reboots.

Rakshasa : why using Coreboot/SeaBios/iPXE is the good approach

- Portability : benefit from all the gory reverse engineering work already done !
- Awesome modularity : embbed existing payloads (as floppy or cdrom images) and PCI roms directly in the main Coreboot rom !
  Eg : bruteforce bootloaders (Brossard, H2HC 2010), bootkits without modification.
- Network stack : ip/udp/tcp, dns, http(s), tftp, ftp... make your own (tcp over dns? Over ntp ?)

PCI rom from scratch (asm)

```
section .text


;---------------------------
; Bios expension ROM header
;---------------------------
        db 0x55         ; Signature
        db 0xaa         ; Signature
        db 17           ; number of sectors
```

DEMO : Evil remote carnal pwnage (of death)

I can write blogs too... Muhahahaha...

DEMO : Evil remote carnal pwnage (of death)

I can write blogs too... Muhahahaha...

How to properly build a botnet ?

- HTTPS + assymetric cryptography (client side certificates, signed updates)
- Fastflux and/or precomputed IP addresses

  If Microsoft can do secure remote updates, so can a malware !
-

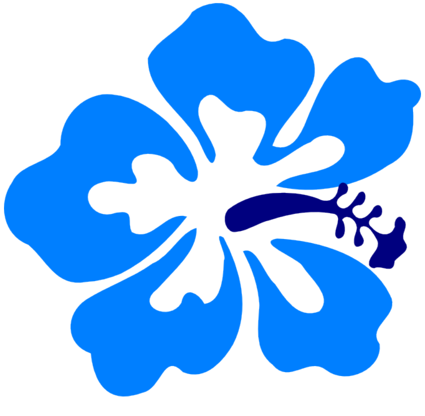
  Avoid DNS take overs by law enforcement agencies by directing the C&C rotatively on innocent web sites (are you gonna shut down Google.com?), use assymetric crypto to push updates.

Why crypto won't save you...

Why crypto won't save you...

- We can fake the bootking/password prompt by booting a remote OS (Truecrypt/Bitlocker)
- Once we know the password, the BIOS backdoor can emulate keyboard typing in 16b real mode by programming the keyboard/motherboard PIC microcontrolers (Brossard, Defcon 2008)
- If necessary, patch back original BIOS/firmwares remotely.

DEMOS

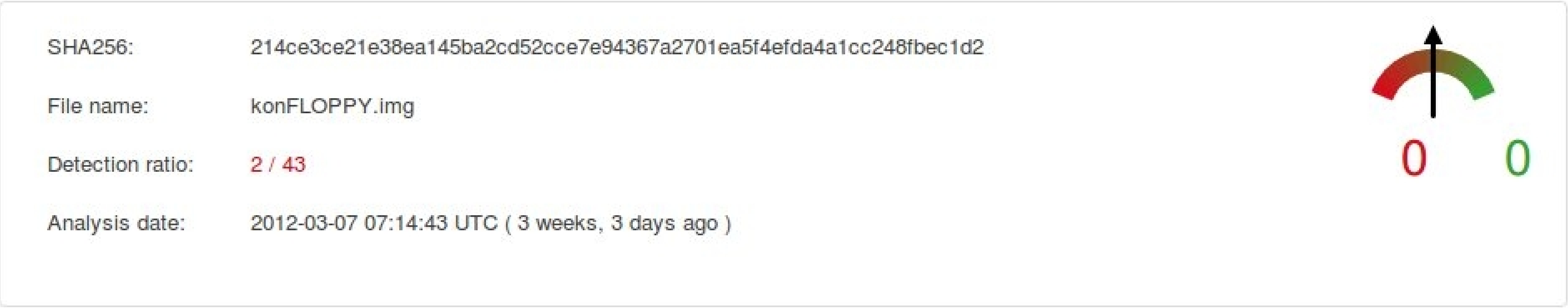


How about Avs ??

- Putting an AV on a server to protect against unknown threats is purely cosmetic.
- You may as well put lipstick on your servers...

Example : 3 years old bootkit

virustotal

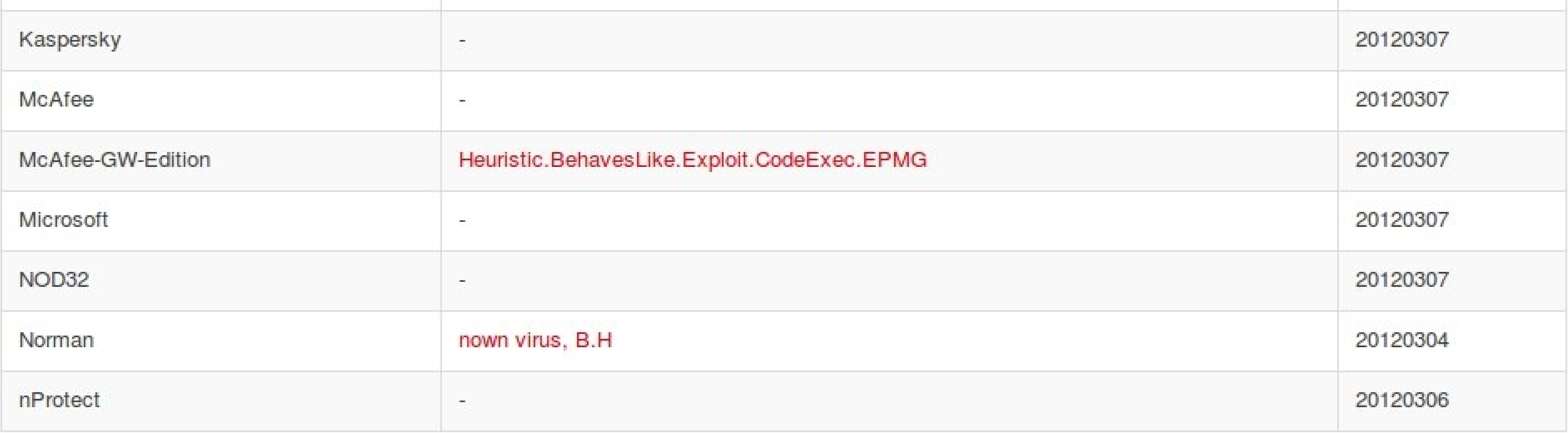


SHA256: 214ce3ce21e38ea145ba2cd52cce7e94367a2701ea5f4efda4a1cc248fbec1d2

File name: konFLOPPY.img

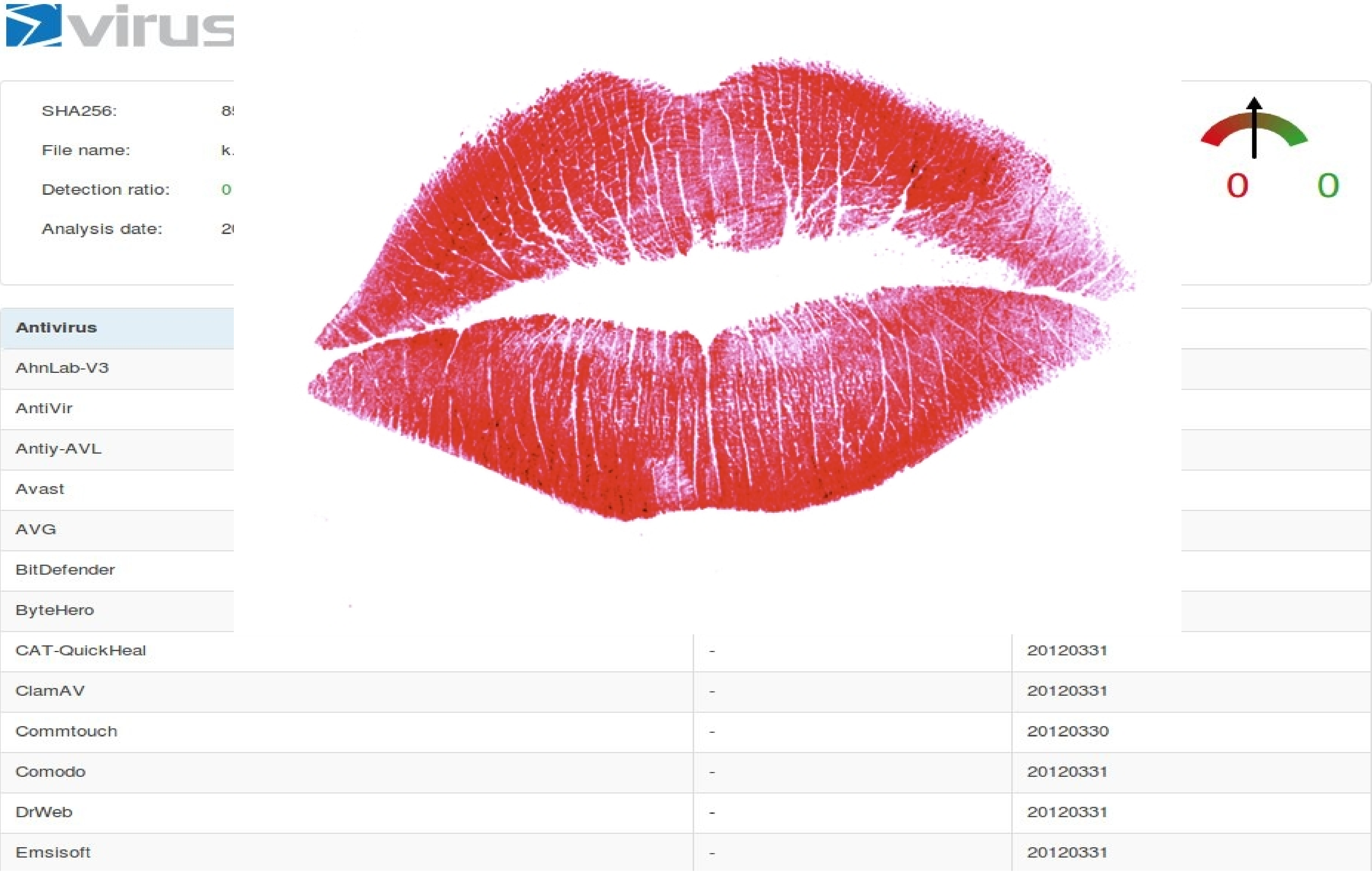
Detection ratio: 2 / 43

Analysis date: 2012-03-07 07:14:43 UTC (3 weeks, 3 days ago)



Kaspersky	-	20120307
McAfee	-	20120307
McAfee-GW-Edition	Heuristic.BehavesLike.Exploit.CodeExec.EPMG	20120307
Microsoft	-	20120307
NOD32	-	20120307
Norman	nown virus, B.H	20120304
nProtect	-	20120306

Example : 3 years old bootkit (+ simple packer)



virus

SHA256: 8

File name: k.

Detection ratio: 0

Analysis date: 2

0 0

Antivirus		
AhnLab-V3		
AntiVir		
Antiy-AVL		
Avast		
AVG		
BitDefender		
ByteHero		
CAT-QuickHeal	-	20120331
ClamAV	-	20120331
Commtouch	-	20120330
Comodo	-	20120331
DrWeb	-	20120331
Emsisoft	-	20120331

Realistic attack scenarii

Realistic attack scenarii

- Physical access :

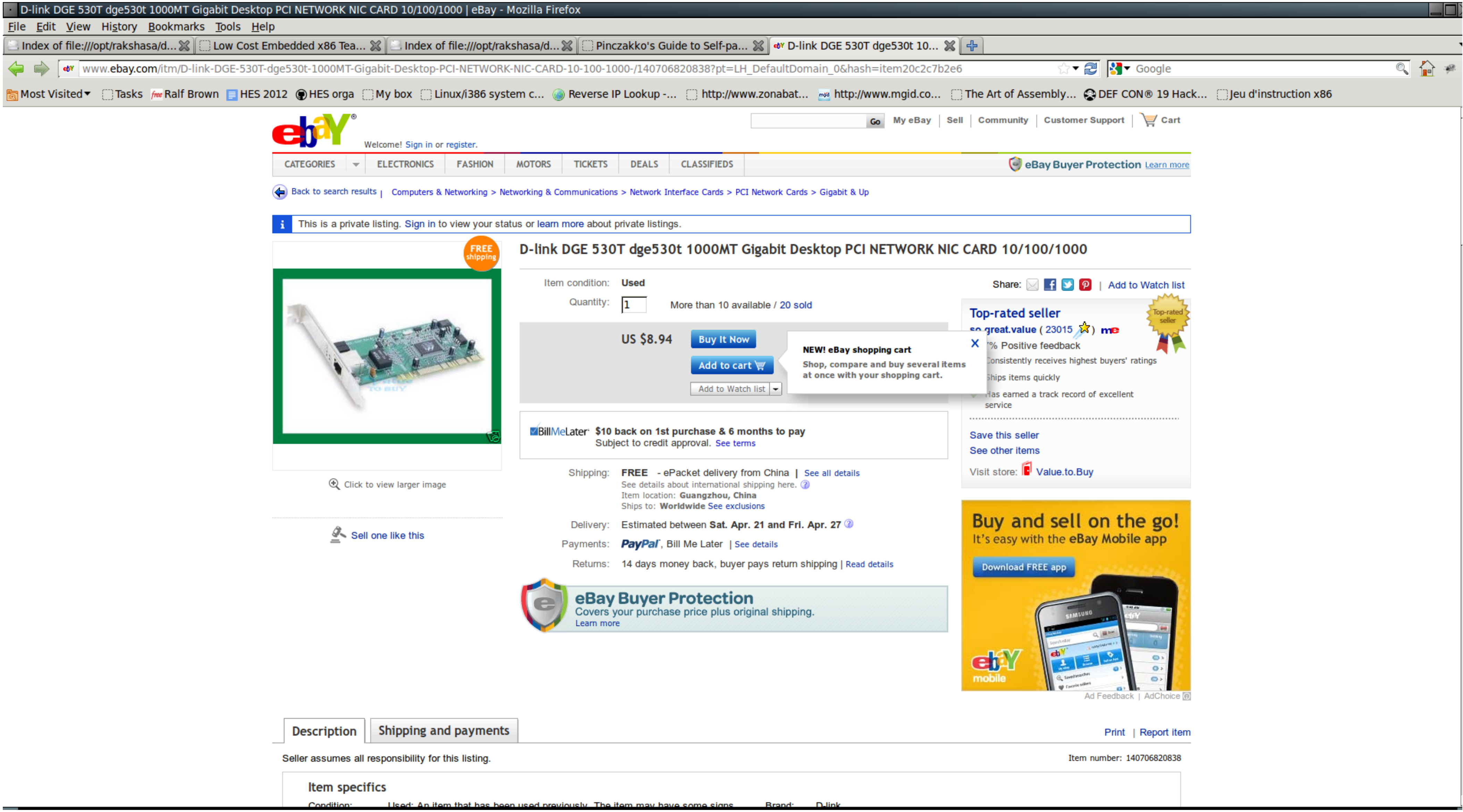
Anybody in the supply chain can backdoor your hardware. Period.
Flash from a bootable USB stick (< 3mins).

- Remote root compromise :

```
If (OS == Linux) {
        flash_bios;

} else {
  Pivot_over_the_MBR ;
}
```

Realistic attack scenarii


D-link DGE 530T dge530t 1000MT Gigabit Desktop PCI NETWORK NIC CARD 10/100/1000 | eBay - Mozilla Firefox
File Edit View History Bookmarks Tools Help
Index of file:///opt/rakshasa/d...
Low Cost Embedded x86 Tea...
Index of file:///opt/rakshasa/d...
Pinczakko's Guide to Self-pa...
D-link DGE 530T dge530t 10...
www.ebay.com/itm/D-link-DGE-530T-dge530t-1000MT-Gigabit-Desktop-PCI-NETWORK-NIC-CARD-10-100-1000-/140706820838?pt=LH_DefaultDomain_0&hash=item20c2c7b2e6
Google
Most Visited
Tasks
Ralf Brown
HES 2012
HES orga
My box
Linux/i386 system c...
Reverse IP Lookup -...
http://www.zonabat...
http://www.mgid.co...
The Art of Assembly...
DEF CON® 19 Hack...
Jeu d'instruction x86
Go
My eBay
Sell
Community
Customer Support
Cart
CATEGORIES
ELECTRONICS
FASHION
MOTORS
TICKETS
DEALS
CLASSIFIEDS
eBay Buyer Protection Learn more
FREE shipping
Buy It Now
Add to cart
Add to Watch list
Download FREE app
Description
Shipping and payments
Print | Report item

Welcome! Sign in or register.

Back to search results | Computers & Networking > Networking & Communications > Network Interface Cards > PCI Network Cards > Gigabit & Up

This is a private listing. Sign in to view your status or learn more about private listings.

D-link DGE 530T dge530t 1000MT Gigabit Desktop PCI NETWORK NIC CARD 10/100/1000

Item condition: Used

Quantity: 1 More than 10 available / 20 sold

Share: | Add to Watch list

US $8.94

NEW! eBay shopping cart

Shop, compare and buy several items at once with your shopping cart.

Top-rated seller

so great.value (23015) me

% Positive feedback

Consistently receives highest buyers' ratings

Ships items quickly

Has earned a track record of excellent service

Save this seller

See other items

Visit store: Value.to.Buy

BillMeLater $10 back on 1st purchase & 6 months to pay
Subject to credit approval. See terms

Click to view larger image

Sell one like this

Shipping: FREE - ePacket delivery from China | See all details
See details about international shipping here.
Item location: Guangzhou, China
Ships to: Worldwide See exclusions

Delivery: Estimated between Sat. Apr. 21 and Fri. Apr. 27

Payments: PayPal, Bill Me Later | See details

Returns: 14 days money back, buyer pays return shipping | Read details

eBay Buyer Protection
Covers your purchase price plus original shipping.
Learn more

Buy and sell on the go!
It's easy with the eBay Mobile app

Ad Feedback | AdChoice

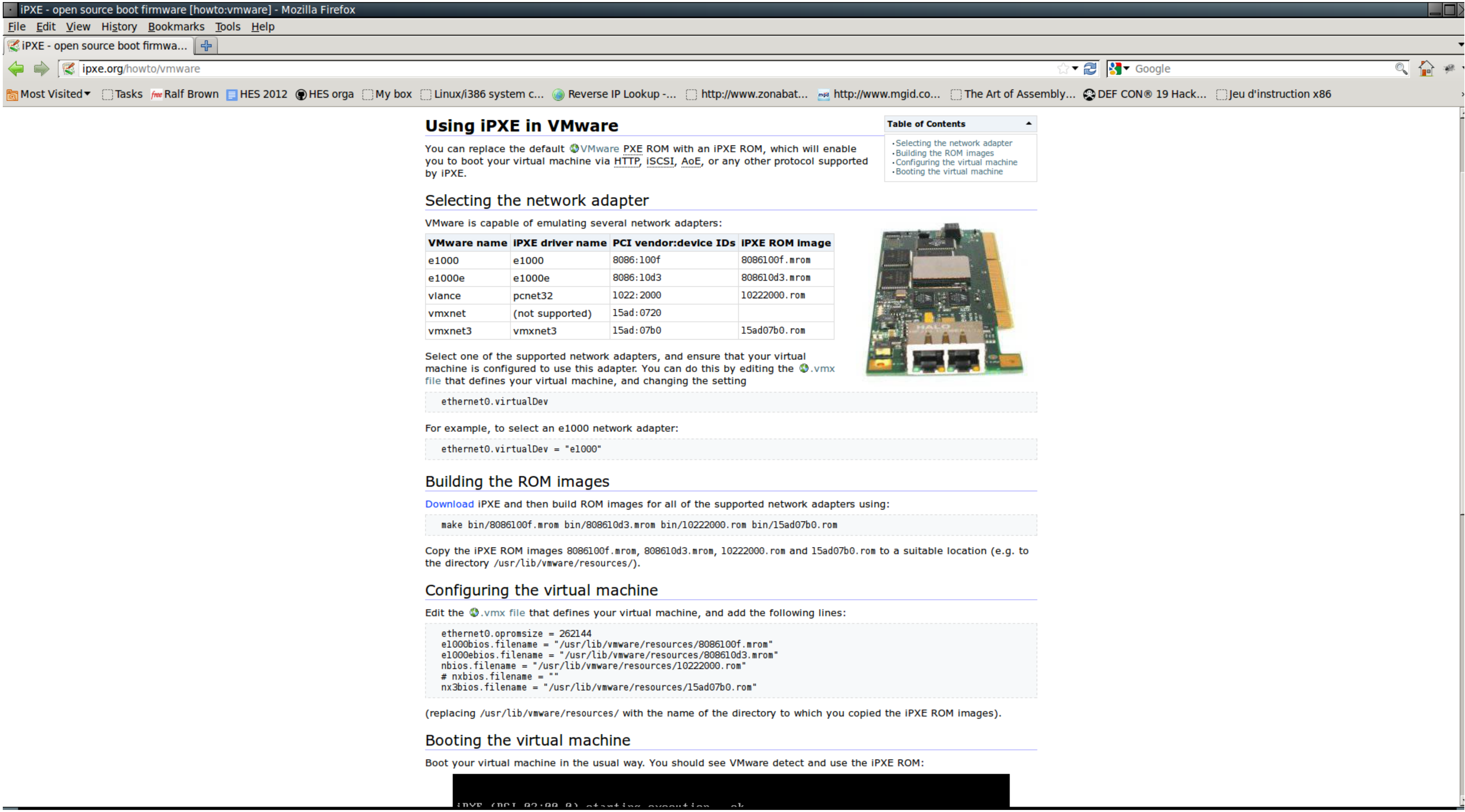
Seller assumes all responsibility for this listing.

Item number: 140706820838

Item specifics

Condition: Used: An item that has been used previously. The item may have some signs Brand: D-link

BONUS : Backdooring the datacenter



iPXE - open source boot firmware [howto:vmware] - Mozilla Firefox
File Edit View History Bookmarks Tools Help
iPXE - open source boot firmwa...
ipxe.org/howto/vmware
Google
Most Visited Tasks Ralf Brown HES 2012 HES orga My box Linux/i386 system c... Reverse IP Lookup -... http://www.zonabat... http://www.mgid.co... The Art of Assembly... DEF CON® 19 Hack... Jeu d'instruction x86

Using iPXE in VMware

You can replace the default VMware PXE ROM with an iPXE ROM, which will enable you to boot your virtual machine via HTTP, iSCSI, AoE, or any other protocol supported by iPXE.

Selecting the network adapter

VMware is capable of emulating several network adapters:

VMware name	iPXE driver name	PCI vendor:device IDs	iPXE ROM image
e1000	e1000	8086:100f	8086100f.mrom
e1000e	e1000e	8086:10d3	808610d3.mrom
vlance	pcnet32	1022:2000	10222000.rom
vmxnet	(not supported)	15ad:0720	
vmxnet3	vmxnet3	15ad:07b0	15ad07b0.rom

Select one of the supported network adapters, and ensure that your virtual machine is configured to use this adapter. You can do this by editing the .vmx file that defines your virtual machine, and changing the setting

```
ethernet0.virtualDev
```

For example, to select an e1000 network adapter:

```
ethernet0.virtualDev = "e1000"
```

Building the ROM images

Download iPXE and then build ROM images for all of the supported network adapters using:

```
make bin/8086100f.mrom bin/808610d3.mrom bin/10222000.rom bin/15ad07b0.rom
```

Copy the iPXE ROM images 8086100f.mrom, 808610d3.mrom, 10222000.rom and 15ad07b0.rom to a suitable location (e.g. to the directory /usr/lib/vmware/resources/).

Configuring the virtual machine

Edit the .vmx file that defines your virtual machine, and add the following lines:

```
ethernet0.opromsize = 262144
e1000bios.filename = "/usr/lib/vmware/resources/8086100f.mrom"
e1000ebios.filename = "/usr/lib/vmware/resources/808610d3.mrom"
nbios.filename = "/usr/lib/vmware/resources/10222000.rom"
# nxbios.filename = ""
nx3bios.filename = "/usr/lib/vmware/resources/15ad07b0.rom"
```

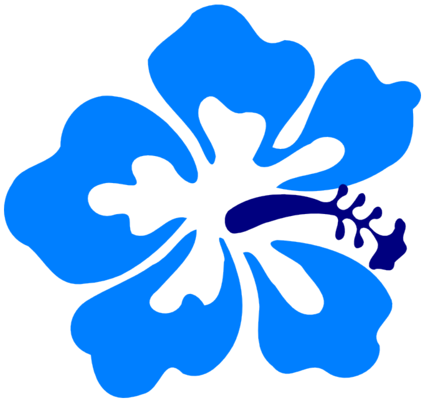
(replacing /usr/lib/vmware/resources/ with the name of the directory to which you copied the iPXE ROM images).

Booting the virtual machine

Boot your virtual machine in the usual way. You should see VMware detect and use the iPXE ROM:

```
iPXE (PCI 02:00.0) starting execution...ok
```

Remediation



Remediation (leads)

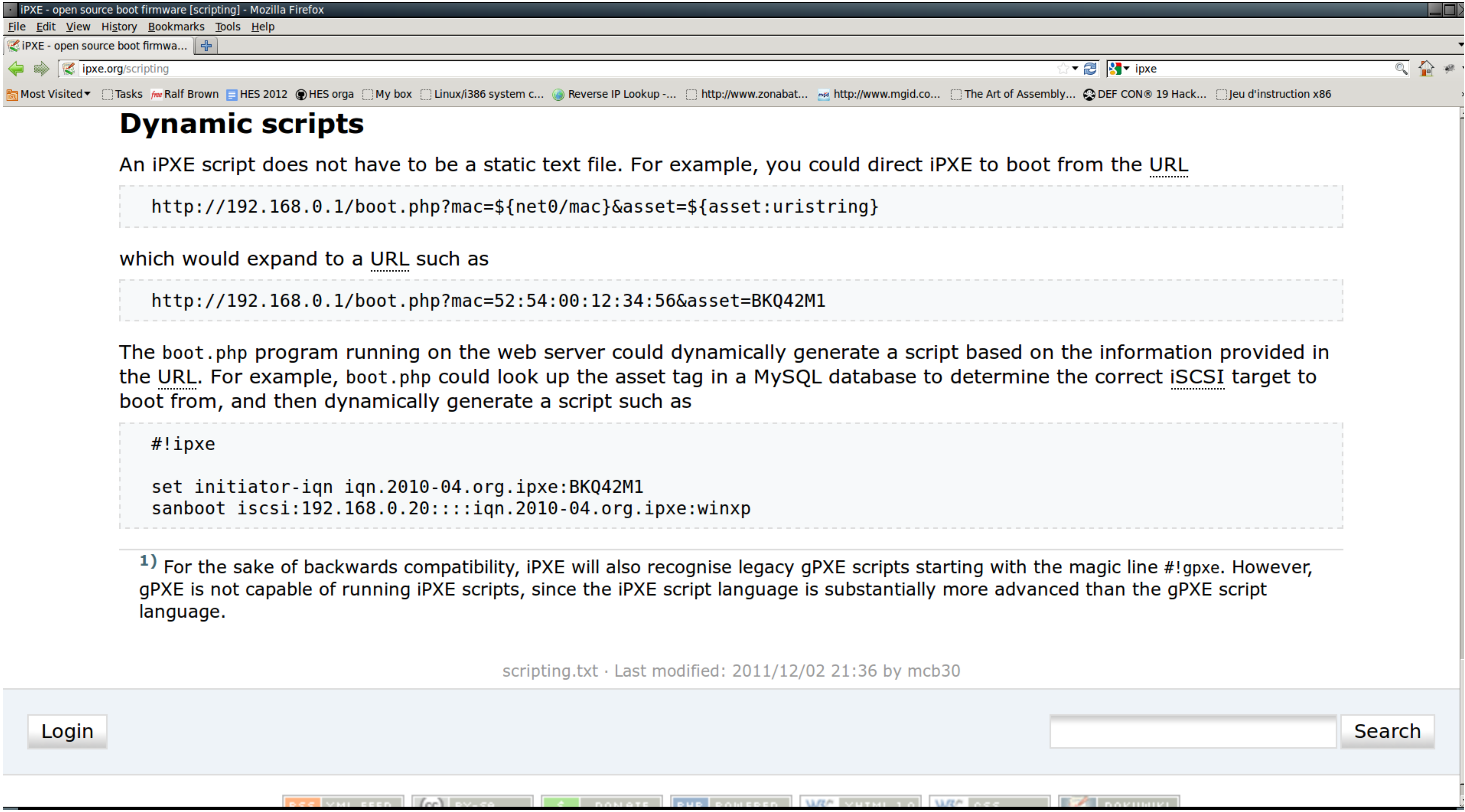
- Flash any firmware uppon reception of new hardware with open source software
- Perform checksums of all firmwares by physically extracting them (FPGA..) : costly !
- Verify the integrity of all firmwares from time to time
- Update forensics best practices :
  1) Include firmwares in SoW
  2) Throw away your computer in case of intrusion

  Even then... not entirely satisfying : the backdoor can flash the original firmwares back remotely.

Side note on remote flashing

- BIOS flashing isn't a problem : the flasher (Linux based) is universal.
- PCI roms flashing is (a bit of) a problem : vendor dependant...

Detecting network card manufacturer from the remote C&C



Dynamic scripts

An iPXE script does not have to be a static text file. For example, you could direct iPXE to boot from the URL

```
http://192.168.0.1/boot.php?mac=${net0/mac}&asset=${asset:uristring}
```

which would expand to a URL such as

```
http://192.168.0.1/boot.php?mac=52:54:00:12:34:56&asset=BKQ42M1
```

The `boot.php` program running on the web server could dynamically generate a script based on the information provided in the URL. For example, `boot.php` could look up the asset tag in a MySQL database to determine the correct iSCSI target to boot from, and then dynamically generate a script such as

```
#!ipxe

set initiator-iqn iqn.2010-04.org.ipxe:BKQ42M1
sanboot iscsi:192.168.0.20::::iqn.2010-04.org.ipxe:winxp
```

[1] For the sake of backwards compatibility, iPXE will also recognise legacy gPXE scripts starting with the magic line `#!gpxe`. However, gPXE is not capable of running iPXE scripts, since the iPXE script language is substantially more advanced than the gPXE script language.

scripting.txt · Last modified: 2011/12/02 21:36 by mcb30

Backdooring like ~~NSA~~ China

Backdooring like a state

Rule #1 : **non attribution**

- you didn't write the free software in first place.
- add a few misleading strings, eg : in mandarin ;)

Rule #2 : **plausible deniability**

- use a bootstrap known remote vulnerability in a network card firmware
  (eg : Duflot's CVE-2010-0104)
  → « honest mistake » if discovered.
- remotely flash the BIOS.
- do your evil thing.
- restore the BIOS remotely.

toucan system

IT serenity®

SHAKACON

SUN, SURF, & C SHELLS™