

Black-box analysis of malware



Vít Bukač

CROCS, Faculty of Informatics, Masaryk University

IT Security Specialist, CIRT, Honeywell Global Security

PV204 Security Technologies



Black-box analysis of malware – Outline

- Lecture
 - Malware
 - Black-box principle
 - Tools
 - Automatic sandbox analysis
 - Document analysis
- Hands-on lab
 - Analysis of provided malware samples

Malware

Malware types

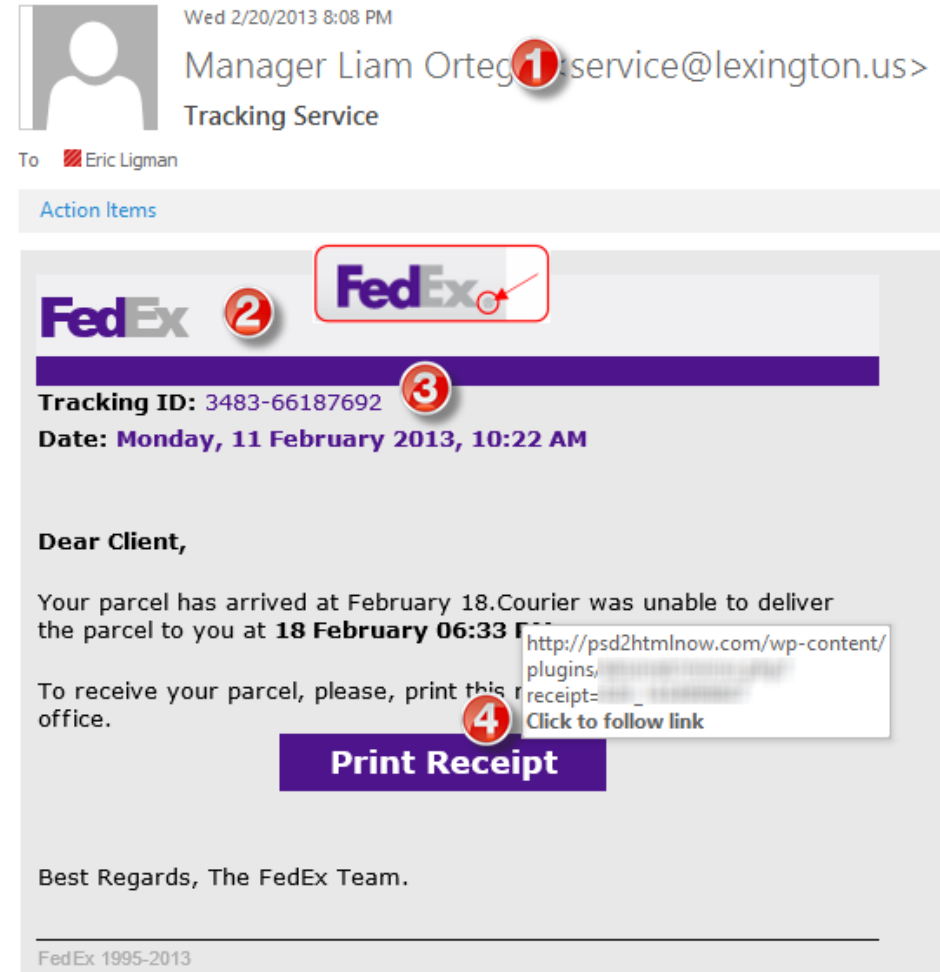
- Trojan
- Fake AV
- Backdoor
- Remote Access Tool (RAT)
- Dropper
- Downloader
- Information stealer
- Keylogger
- Ransomware
- Sniffer
- Virus
- Worm
- Spyware
- Adware
- Botnet

Malware infection vectors

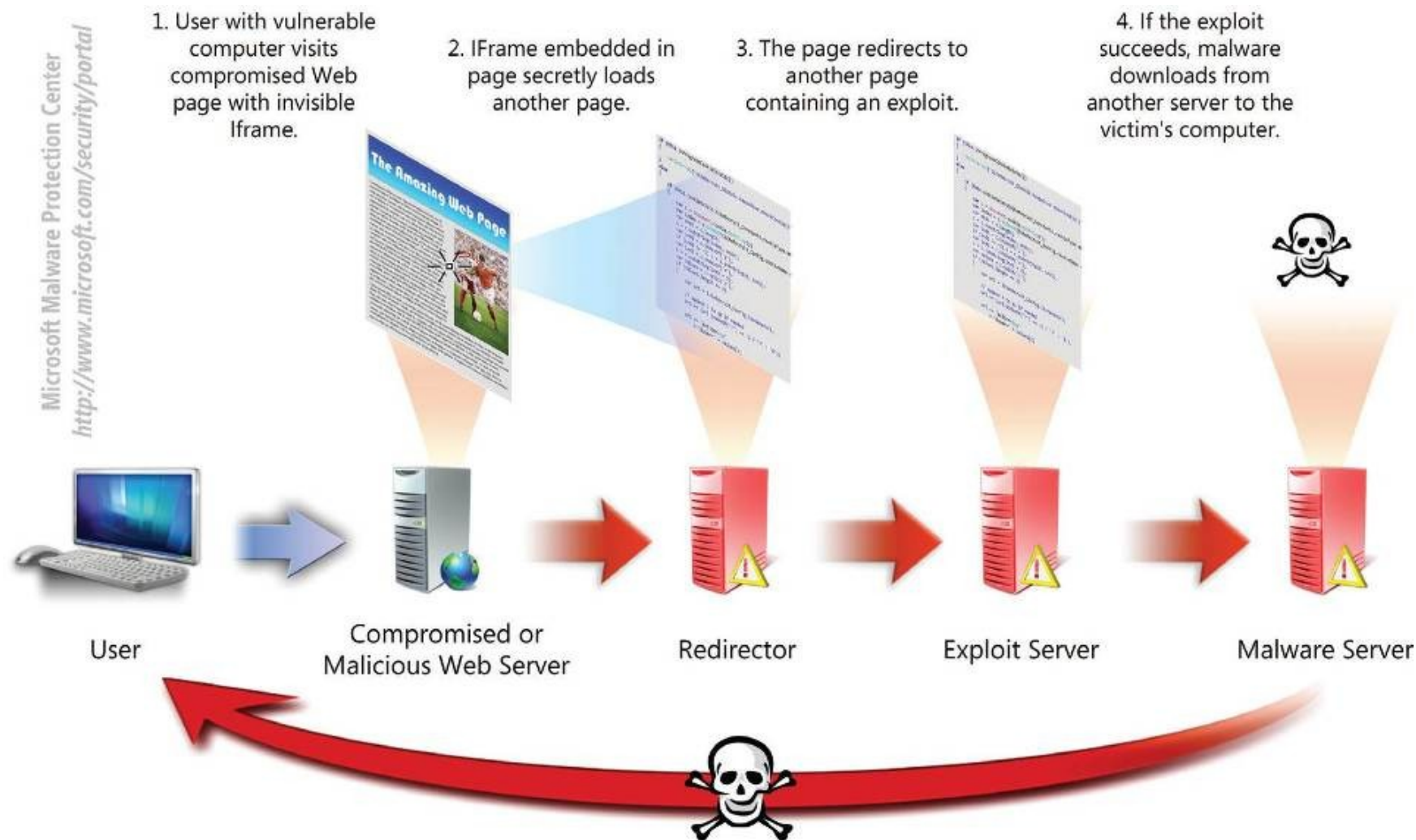
- Email
 - Link
 - Attachment
- Malicious website
 - Drive-by download
- USB
- Cracked software
- Worms
- Social engineering

Infection vector – Phishing

- Subject
 - “Account blocked”
 - “Package to be delivered”
 - “Expiring subscription”
 - “Please process payment”
- Signs
 - Unexpected sender address (1)
 - Graphic errors (2)
 - Erroneous info (3)
 - Links to unexpected URL (4)
 - Links to same URL
 - Use of threats
 - Sense of urgency



Infection vector – Drive-by download



Infection vector – USB

- Autoruns



- BadUSB (Q3 2014)

No response from chip vendors

No response from peripheral vendors

No OS vendor response

Fixes are not yet in sight

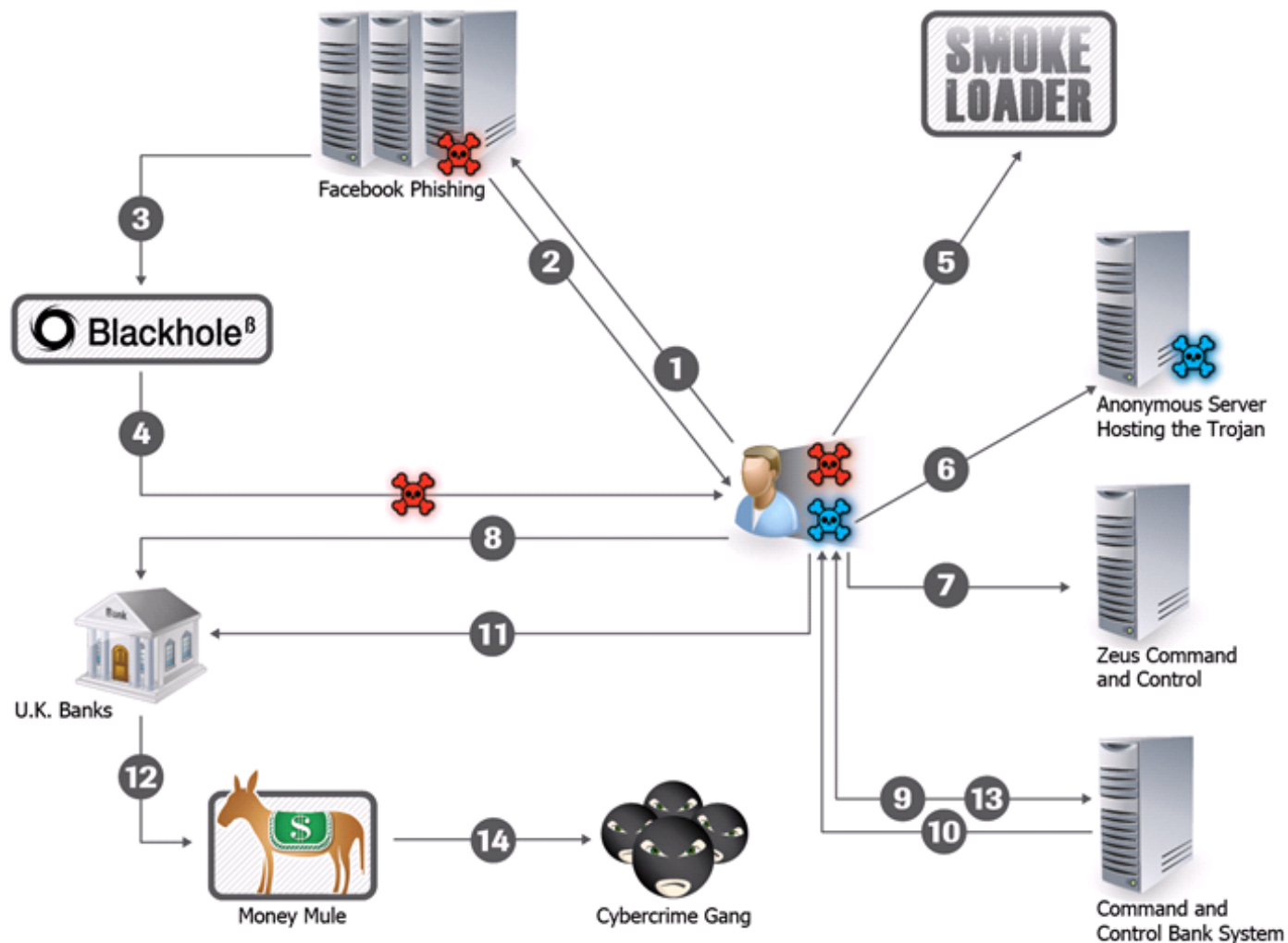
- Phison, the mostly discussed vendor, notes that they are already offering better chips. Their customers don't seem to chose them often
- Other affected vendors have stayed quiet
- No affected vendor offers patches or a threat advisory
- OS implementers do not appear to work on solution; with one exception: FreeBSD adds an option to switch off USB enumeration

vs.

BadUSB malware becomes more realistic

- Sample exploit code for Phison USB 3 controllers was released by Adam Caudill and Brandon Wilson at Derbycon in September
- Only mitigation attempts right now are quick fixes such as GData's Keyboard Guard

Example – Zeus infection



Malware Kill Chain

| Phase | Detect | Deny | Disrupt | Degrade | Deceive | Destroy |
|-----------------------|---------------|---------------|------------|--------------------|--------------|---------|
| Reconnaissance | Web analytics | Firewall ACL | | | | |
| Weaponization | NIDS | NIPS | | | | |
| Delivery | Vigilant user | Proxy filter | In-line AV | Queuing | | |
| Exploitation | HIDS | Patch | DEP | | | |
| Installation | HIDS | “chroot” jail | AV | | | |
| C2 | NIDS | Firewall ACL | NIPS | Tarpit | DNS redirect | |
| Actions on Objectives | Audit log | | | Quality of Service | Honeypot | |

Black box malware analysis

Use cases

- Communication between local file server and an unknown IP address in China has been observed. **What** process is responsible for the communication?
- Malware is creating temporary files. **Where** are these files located?
- Malware executable is created again after system reboot. **How** is it possible and what is causing it?
- A new type of malware has been spreading through internal network. How to quickly **assess the malware** capabilities? What is its purpose? Is it based on any well-known tool?

Black box malware analysis

- Dynamic analysis – file is executed
- Analysis without internal knowledge
 - Observable inputs
 - Observable outputs
- Quick, simple
- Common monitoring tools
- Collected indicators about
 - Filenames, process names, process parent/child relationships, temporal relationships, domain names, IP addresses, registry keys, persistence methods, cleanup operations etc.
- Can be highly automated

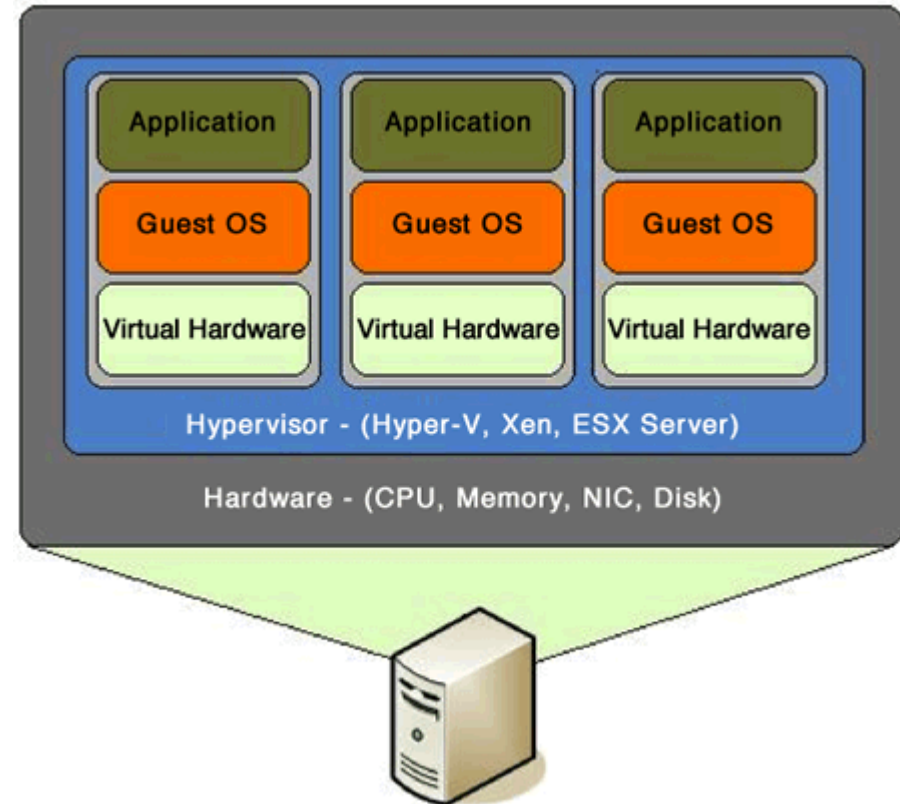


Black box malware analysis – Basic

1. Prepare analysis environment
2. Create snapshot
3. Run monitoring tools
4. Run malware
5. Collect and observe interactions between malware and VM
6. Restore snapshot
7. Repeat 3-6 as needed

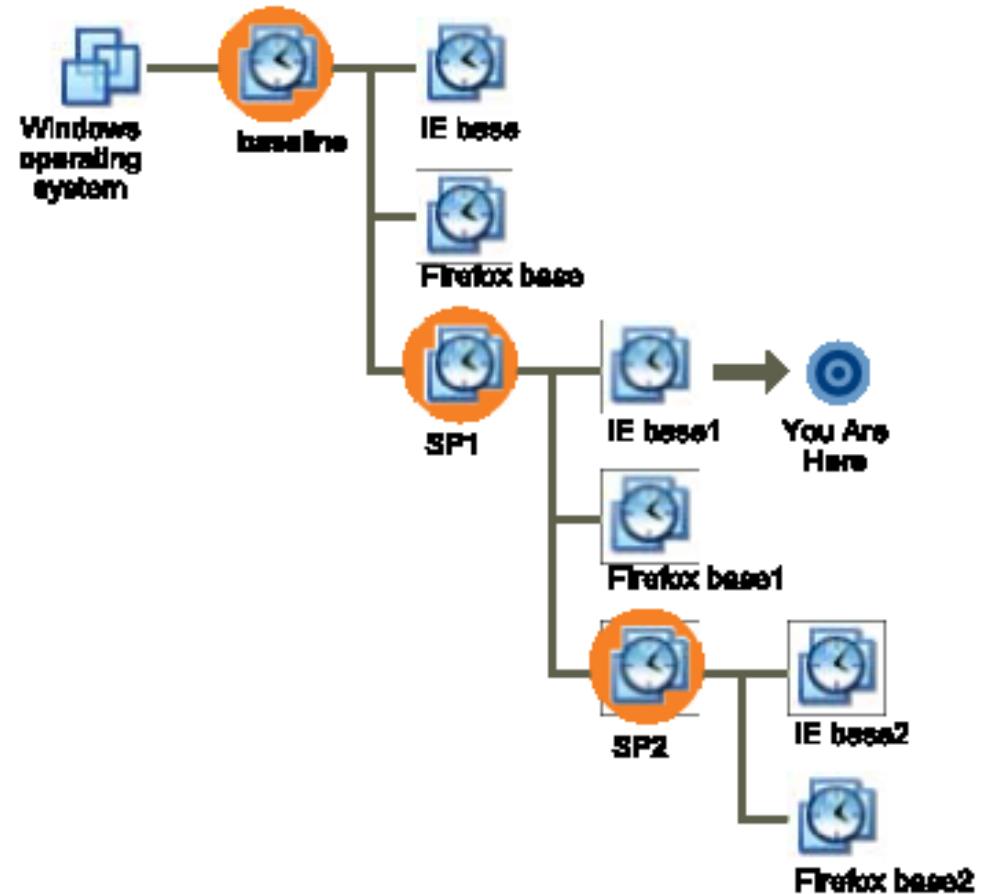
Analysis environment

- Virtual Machine
 - Limited/no connectivity
 - Virtualized services (DNS, HTTP,...)
 - Several VMs for various host types
- Software
 - Monitoring tools
 - Often exploited applications
- Risks
 - VM isolation breach
 - Malware inactivity in VM



Virtual machine snapshot

- Snapshots
 - Saved state of VM
 - Disk state, memory state
- Quick restoration of previous state



Tools

Network analysis

- Capturing **sent/received packets**
- Protocol dissection
- Promiscuous mode
- Tools
 - Tcpdump, Wireshark, NetworkMiner
- Indicators
 - Domain names, IP addresses, protocols, ports, HTTP parameters
- Q&A
 - Who is this program communicating with? What reputation does the partner have? What data is exchanged? Is it encrypted or obfuscated?

Network analysis – What to look for

- New established connections – HTTP 80/8080
 - Direct calls for domains without DNS lookup
 - Random domain names (e.g., rpxiodffd.biz)
 - Suspicious domain names (e.g., google.org)
 - Similarly looking domain names (e.g., osinstall.biz, swinstall.biz, swinstall.com)
- Outgoing portscans
- Ping/DNS request for well known services
 - Connection availability test
- Be aware of background OS/processes activities!

Example – Wireshark

The screenshot displays the Wireshark interface for a capture file named 'test.cap'. The main window shows a list of 12 captured packets. Packet 11 is selected, and its details are expanded in the lower pane. The status bar at the bottom indicates that 120 packets are displayed from a 14 KB capture file.

| No. | Time | Source | Destination | Protocol | Length | Info |
|-----|----------|-------------|-----------------|----------|--------|-----------------------------------|
| 1 | 0.000000 | 192.168.0.2 | Broadcast | ARP | 42 | Gratuitous ARP for 192.168.0.2 (F |
| 2 | 0.299139 | 192.168.0.1 | 192.168.0.2 | NBNS | 92 | Name query NBSTAT *<00><00><00><0 |
| 3 | 0.299214 | 192.168.0.2 | 192.168.0.1 | ICMP | 70 | Destination unreachable (Port un |
| 4 | 1.025659 | 192.168.0.2 | 224.0.0.22 | IGMP | 54 | v3 Membership Report / Join group |
| 5 | 1.044366 | 192.168.0.2 | 192.168.0.1 | DNS | 110 | standard query SRV _ldap._tcp.nbg |
| 6 | 1.048652 | 192.168.0.2 | 239.255.255.250 | SSDP | 175 | M-SEARCH * HTTP/1.1 |
| 7 | 1.050784 | 192.168.0.2 | 192.168.0.1 | DNS | 86 | standard query SOA nb10061d.ww004 |
| 8 | 1.055053 | 192.168.0.1 | 192.168.0.2 | SSDP | 337 | HTTP/1.1 200 OK |
| 9 | 1.082038 | 192.168.0.2 | 192.168.0.255 | NBNS | 110 | Registration NB NB10061D<00> |
| 10 | 1.111945 | 192.168.0.2 | 192.168.0.1 | DNS | 87 | standard query A proxyconf.ww004. |
| 11 | 1.226156 | 192.168.0.2 | 192.168.0.1 | TCP | 62 | ncu-2 > http [SYN] Seq=0 win=6424 |
| 12 | 1.227282 | 192.168.0.1 | 192.168.0.2 | TCP | 60 | http > ncu-2 [SYN, ACK] Seq=0 Ack |

Frame 11: 62 bytes on wire (496 bits), 62 bytes captured (496 bits)

- Ethernet II, Src: 192.168.0.2 (00:0b:5d:20:cd:02), Dst: Netgear_2d:75:9a (00:09:5b:2d:75:9a)
- Internet Protocol, src: 192.168.0.2 (192.168.0.2), dst: 192.168.0.1 (192.168.0.1)
- Transmission Control Protocol, Src Port: ncu-2 (3196), Dst Port: http (80), Seq: 0, Len: 0
 - Source port: ncu-2 (3196)
 - Destination port: http (80)
 - [Stream index: 5]
 - Sequence number: 0 (relative sequence number)
 - Header length: 28 bytes
 - Flags: 0x02 (SYN)
 - window size value: 64240

```
0000 00 09 5b 2d 75 9a 00 0b 5d 20 cd 02 08 00 45 00  ..[-u... ] ....E.
0010 00 30 18 48 40 00 80 06 61 2c c0 a8 00 02 c0 a8  .0.H@... a,.....
0020 00 01 0c 7c 00 50 3c 36 95 f8 00 00 00 00 70 02  ...|.P<6 .....p.
0030 fa f0 27 e0 00 00 02 04 05 b4 01 01 04 02  .. ..... .....
```

File: "C:/test.cap" 14 KB 00:00:02 | Packets: 120 Displayed: 120 Marked: 0 Load time: 0:00:00 | Profile: Default

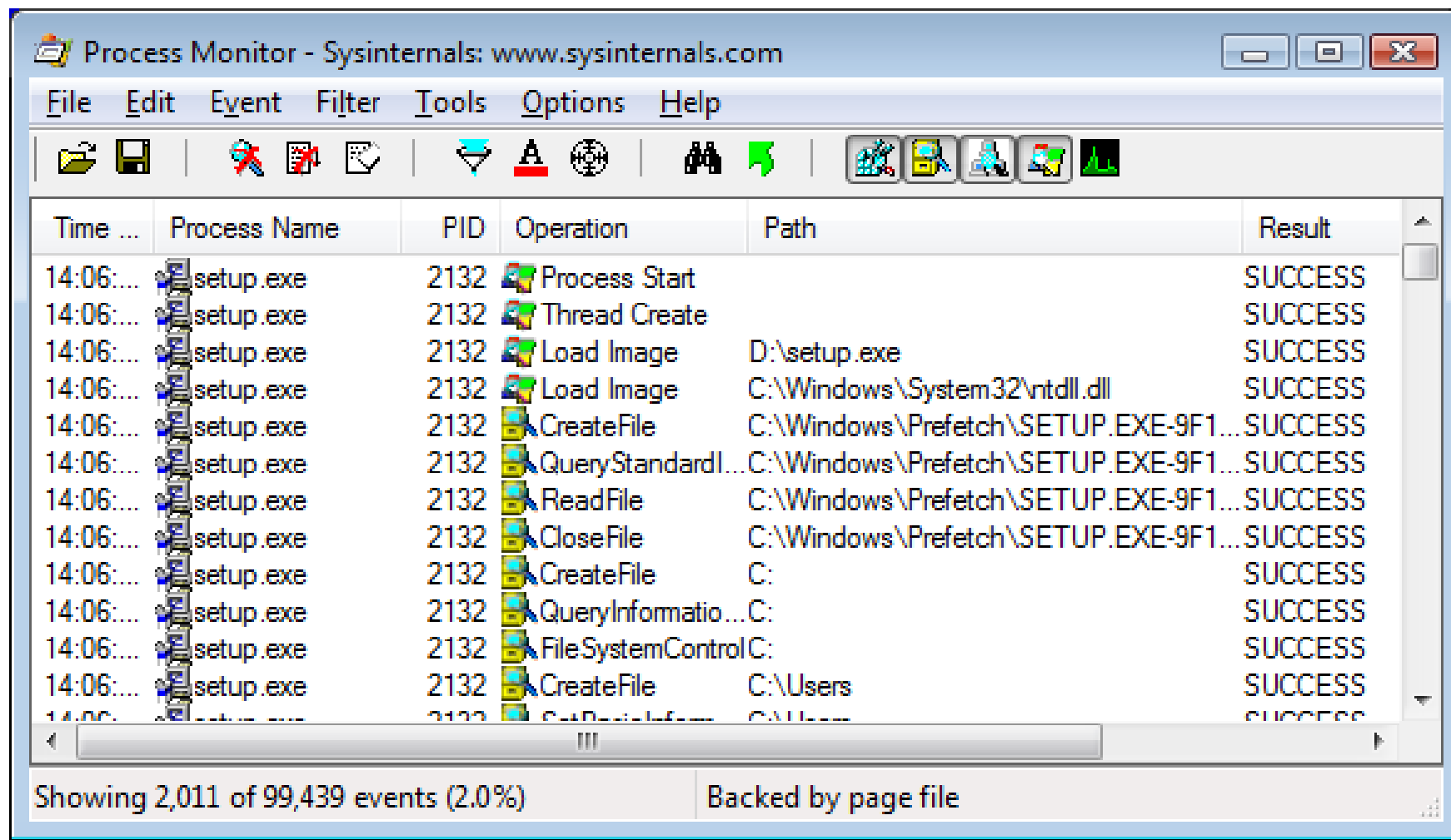
File system

- Observing **file accesses and modifications**
- Background file manipulation
- Tools
 - Procmon, Handle
- Indicators
 - File names, folder names, order of actions, compromise spread through local system
- Q&A
 - Where is malware copied after the initial infection? What filenames are used? Where is the collected data stored?

File system – What to look for

- New file names & folders
 - New created files and folders
 - Batch files (.cmd, .bat, .vbs, .ps1)
 - Known favorite malware file names (e.g., 1.exe, test.exe, new.exe)
 - Known file names in uncommon folders (e.g., C:\Temp\svchost.exe)
 - Recycler
- Modifications of system files
- Temporary storage files, encrypted archives

Example – Procmon



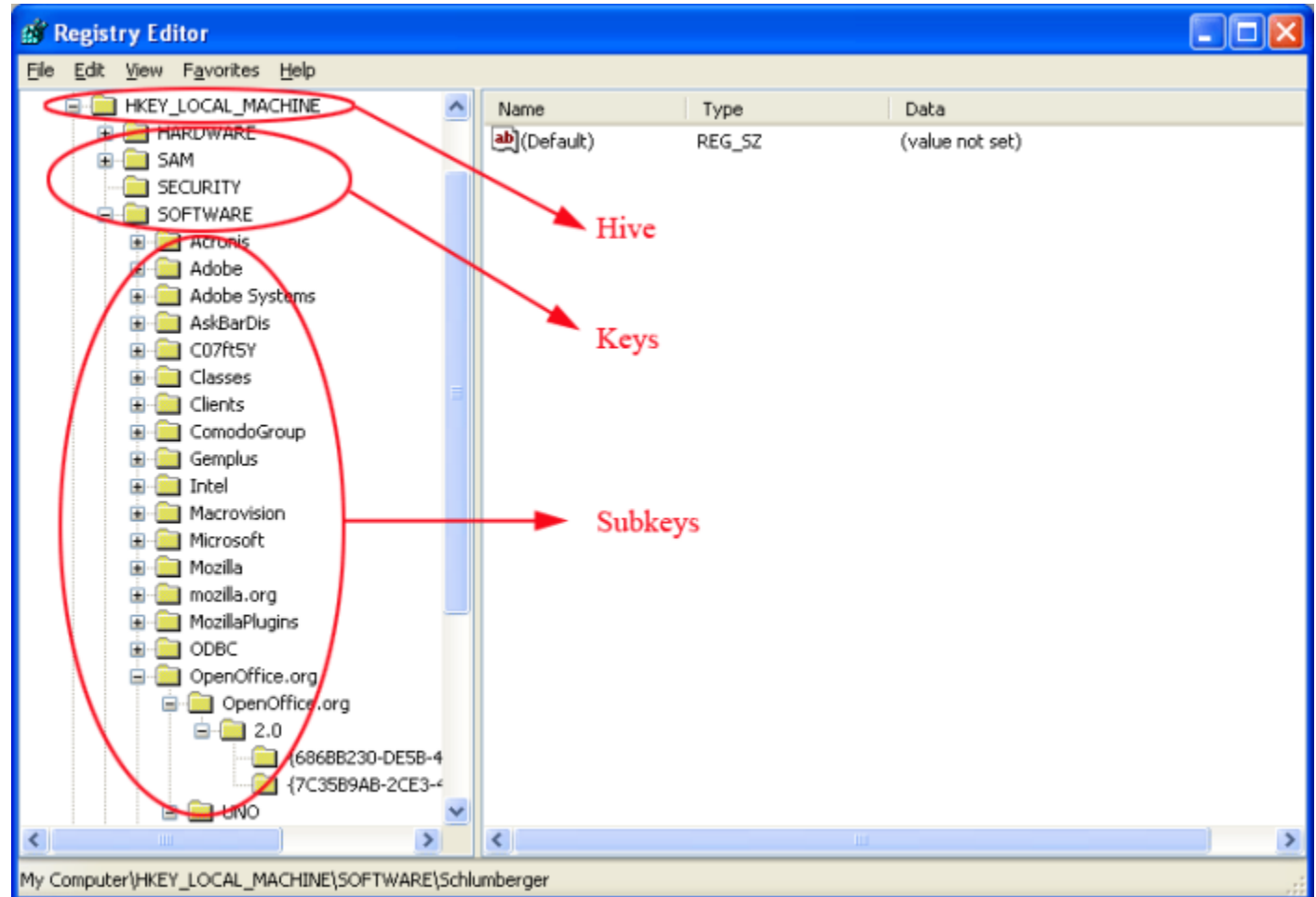
The screenshot shows the Process Monitor application window. The title bar reads "Process Monitor - Sysinternals: www.sysinternals.com". The menu bar includes "File", "Edit", "Event", "Filter", "Tools", "Options", and "Help". The toolbar contains various icons for file operations, filters, and monitoring. The main area is a table with the following columns: "Time ...", "Process Name", "PID", "Operation", "Path", and "Result". The table displays a series of events for the process "setup.exe" with PID 2132, all of which resulted in "SUCCESS".

| Time ... | Process Name | PID | Operation | Path | Result |
|-----------|--------------|------|--------------------|--------------------------------------|---------|
| 14:06:... | setup.exe | 2132 | Process Start | | SUCCESS |
| 14:06:... | setup.exe | 2132 | Thread Create | | SUCCESS |
| 14:06:... | setup.exe | 2132 | Load Image | D:\setup.exe | SUCCESS |
| 14:06:... | setup.exe | 2132 | Load Image | C:\Windows\System32\ntdll.dll | SUCCESS |
| 14:06:... | setup.exe | 2132 | CreateFile | C:\Windows\Prefetch\SETUP.EXE-9F1... | SUCCESS |
| 14:06:... | setup.exe | 2132 | QueryStandardI... | C:\Windows\Prefetch\SETUP.EXE-9F1... | SUCCESS |
| 14:06:... | setup.exe | 2132 | ReadFile | C:\Windows\Prefetch\SETUP.EXE-9F1... | SUCCESS |
| 14:06:... | setup.exe | 2132 | CloseFile | C:\Windows\Prefetch\SETUP.EXE-9F1... | SUCCESS |
| 14:06:... | setup.exe | 2132 | CreateFile | C: | SUCCESS |
| 14:06:... | setup.exe | 2132 | QueryInformatio... | C: | SUCCESS |
| 14:06:... | setup.exe | 2132 | FileSystemControl | C: | SUCCESS |
| 14:06:... | setup.exe | 2132 | CreateFile | C:\Users | SUCCESS |
| 14:06:... | setup.exe | 2132 | SetDesktopForm... | C:\Users | SUCCESS |

Showing 2,011 of 99,439 events (2.0%) Backed by page file

Registry

- Registry DB changes
- Persistence
- Tools
 - Regedit, RegRipper, Autoruns



Registry – What to look for

- Well-known locations
 - Autorun locations
 - Task scheduler
- Changes tracking
- Keywords fulltext search
 - Filenames
 - Processes
 - Domain names



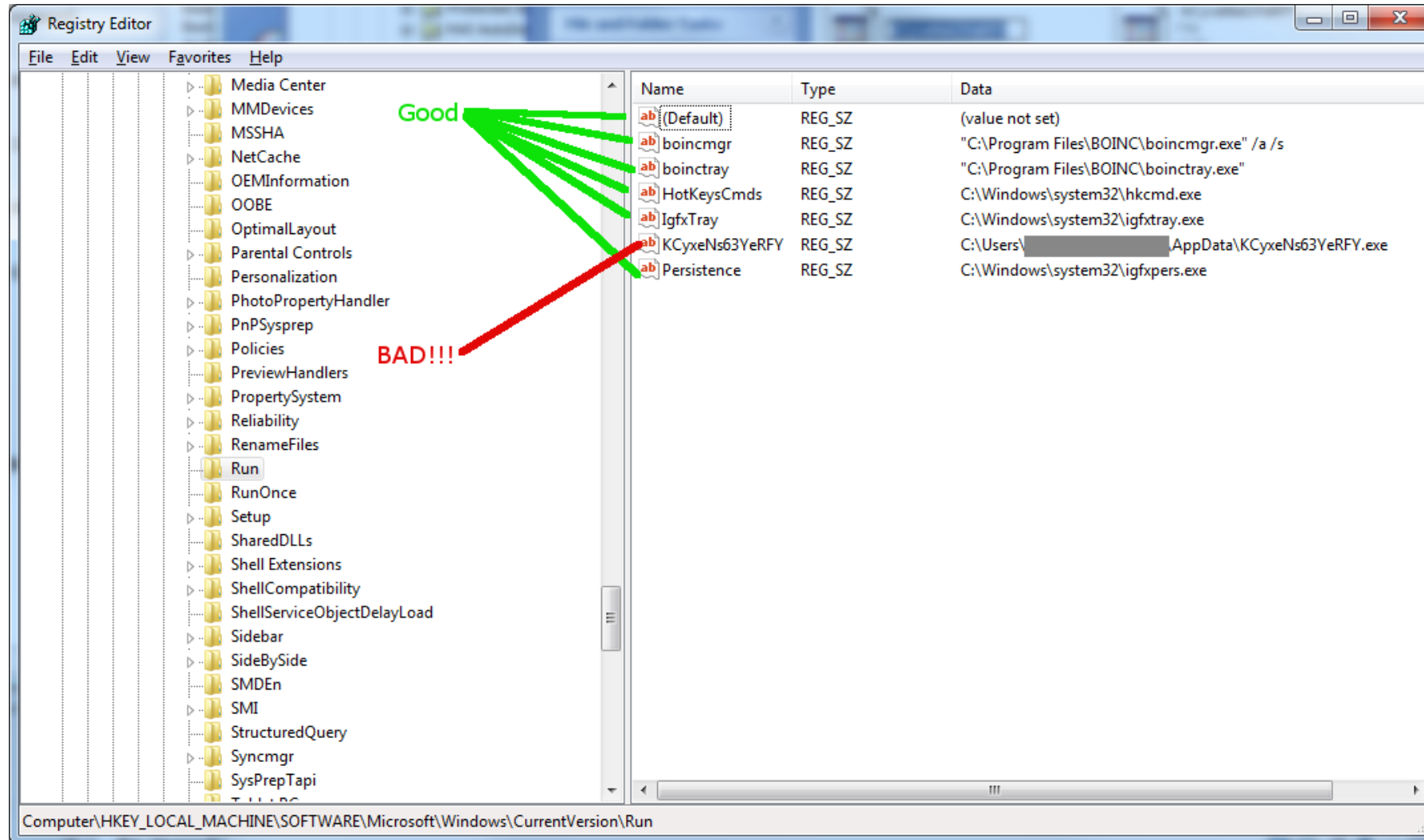
ThreatExpert

Submission Summary:

- ▣ The newly created Registry Values are:
 - [HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Security Center]
 - ┆ UacDisableNotify = 0x00000001
 - [HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Security Center\Svc]
 - ┆ AntiVirusOverride = 0x00000001
 - ┆ AntiVirusDisableNotify = 0x00000001
 - ┆ FirewallDisableNotify = 0x00000001
 - ┆ FirewallOverride = 0x00000001
 - ┆ UpdatesDisableNotify = 0x00000001
 - ┆ UacDisableNotify = 0x00000001

to disable notification of firewall, antivirus and/or update status through the Windows Security Center

Registry – Regedit



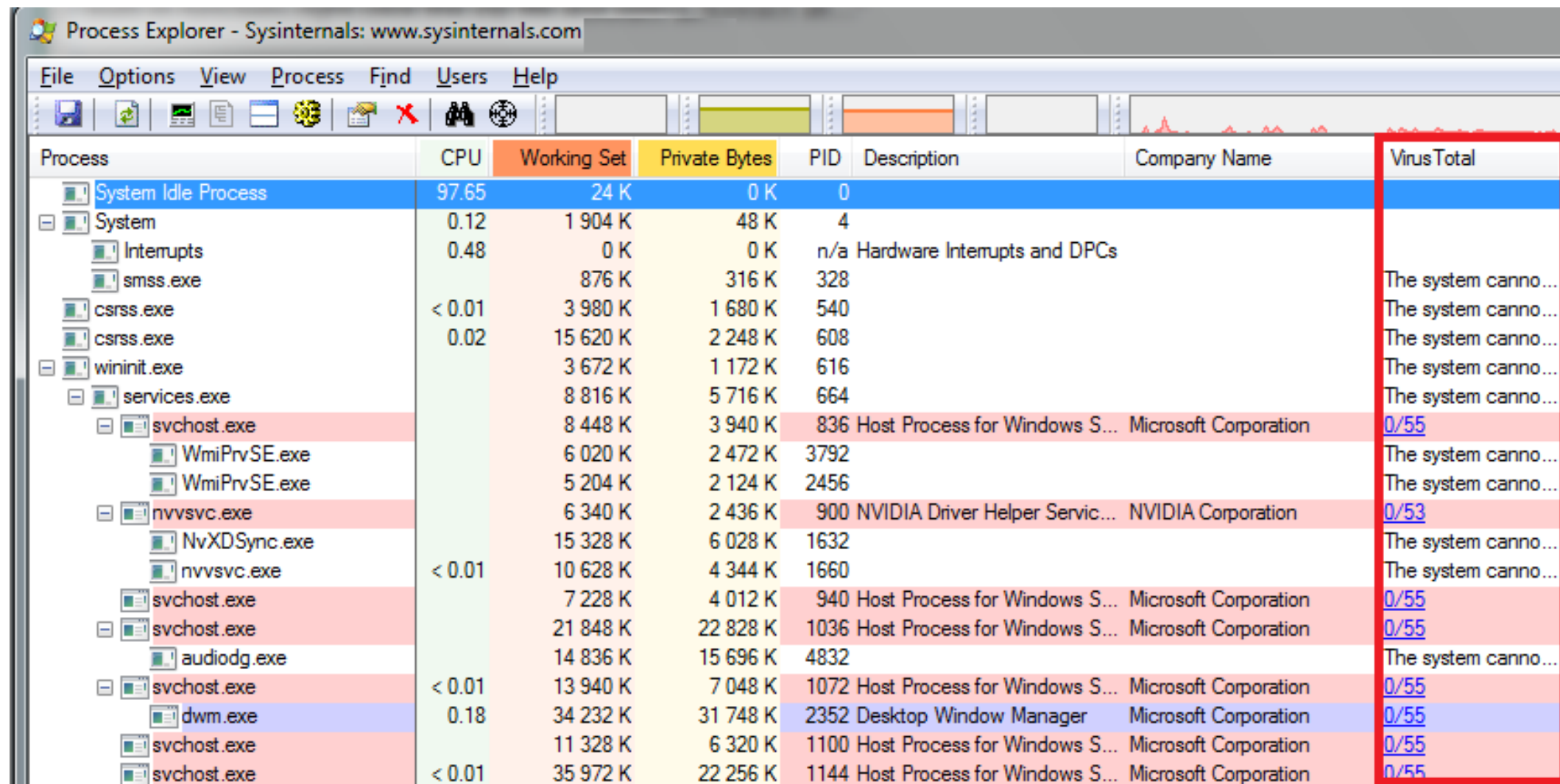
Processes

- Observing initial system compromise
- **Processes** parent/child **relationships**
- Tools
 - Process Explorer, Procmon
- Indicators
 - Process names, order of execution, dropper activity
- Q&A
 - What processes are run after malware binary is executed? Are batch files involved? Are there watcher processes?

Processes – What to look for

- Order of executables
 - Initial malware
 - Dropper/downloader
 - Persistence executable
 - Final malware
- Command line interpreters
 - cmd.exe
 - Powershell
 - Cscript, wscript

Example – Process Explorer



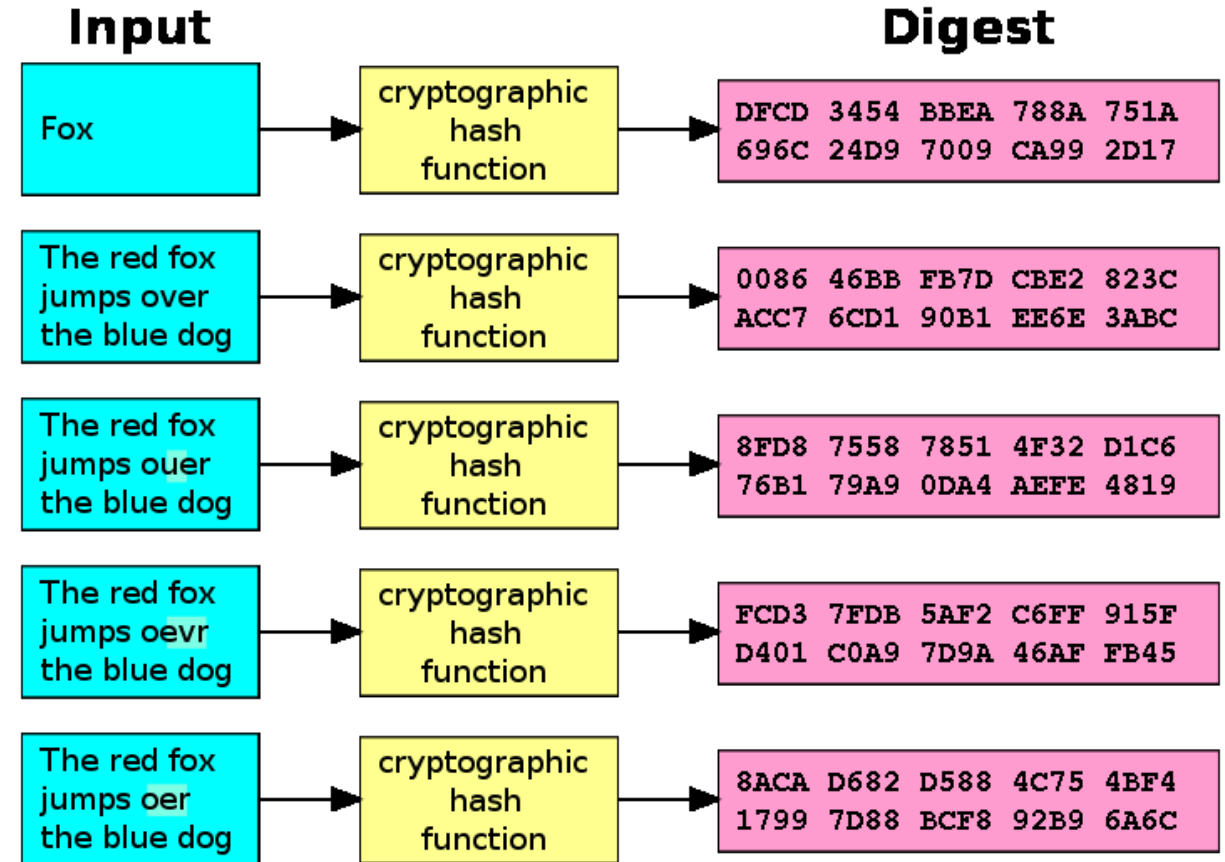
Process Explorer - Sysinternals: www.sysinternals.com

File Options View Process Find Users Help

| Process | CPU | Working Set | Private Bytes | PID | Description | Company Name | Virus Total |
|---------------------|--------|-------------|---------------|------|--------------------------------|-----------------------|--|
| System Idle Process | 97.65 | 24 K | 0 K | 0 | | | |
| System | 0.12 | 1 904 K | 48 K | 4 | | | |
| Interrupts | 0.48 | 0 K | 0 K | n/a | Hardware Interrupts and DPCs | | |
| smss.exe | | 876 K | 316 K | 328 | | | The system cannot find the file specified. |
| csrss.exe | < 0.01 | 3 980 K | 1 680 K | 540 | | | The system cannot find the file specified. |
| csrss.exe | 0.02 | 15 620 K | 2 248 K | 608 | | | The system cannot find the file specified. |
| wininit.exe | | 3 672 K | 1 172 K | 616 | | | The system cannot find the file specified. |
| services.exe | | 8 816 K | 5 716 K | 664 | | | The system cannot find the file specified. |
| svchost.exe | | 8 448 K | 3 940 K | 836 | Host Process for Windows S... | Microsoft Corporation | 0/55 |
| WmiPrvSE.exe | | 6 020 K | 2 472 K | 3792 | | | The system cannot find the file specified. |
| WmiPrvSE.exe | | 5 204 K | 2 124 K | 2456 | | | The system cannot find the file specified. |
| nvsvsvc.exe | | 6 340 K | 2 436 K | 900 | NVIDIA Driver Helper Servic... | NVIDIA Corporation | 0/53 |
| NvXDSync.exe | | 15 328 K | 6 028 K | 1632 | | | The system cannot find the file specified. |
| nvsvsvc.exe | < 0.01 | 10 628 K | 4 344 K | 1660 | | | The system cannot find the file specified. |
| svchost.exe | | 7 228 K | 4 012 K | 940 | Host Process for Windows S... | Microsoft Corporation | 0/55 |
| svchost.exe | | 21 848 K | 22 828 K | 1036 | Host Process for Windows S... | Microsoft Corporation | 0/55 |
| audiodg.exe | | 14 836 K | 15 696 K | 4832 | | | The system cannot find the file specified. |
| svchost.exe | < 0.01 | 13 940 K | 7 048 K | 1072 | Host Process for Windows S... | Microsoft Corporation | 0/55 |
| dwm.exe | 0.18 | 34 232 K | 31 748 K | 2352 | Desktop Window Manager | Microsoft Corporation | 0/55 |
| svchost.exe | | 11 328 K | 6 320 K | 1100 | Host Process for Windows S... | Microsoft Corporation | 0/55 |
| svchost.exe | < 0.01 | 35 972 K | 22 256 K | 1144 | Host Process for Windows S... | Microsoft Corporation | 0/55 |

Executable file analysis

- Cryptographic hash
 - Hash function which is considered practically impossible to invert
 - Unique identification of file
 - Counter: Polymorphism
 - MD5, SHA1
- Fuzzy hash
 - Context triggered piecewise hash
 - Families of files
 - ssdeep
- Strings



Example – Strings

server.exe

AppData

4bcce4de98bcd4d29f66c0fe1ffe002

hackerhani.no-ip.biz **Domain name**

Software\Microsoft\Windows\CurrentVersion\Run **Persistence registry key**

Software\

yy-MM-dd

??-??-??

Microsoft

Windows

SystemDrive

netsh firewall delete allowedprogram " **Commands to be executed**

Software

cmd.exe /c ping 0 -n 2 & del "

SEE_MASK_NOZONECHECKS

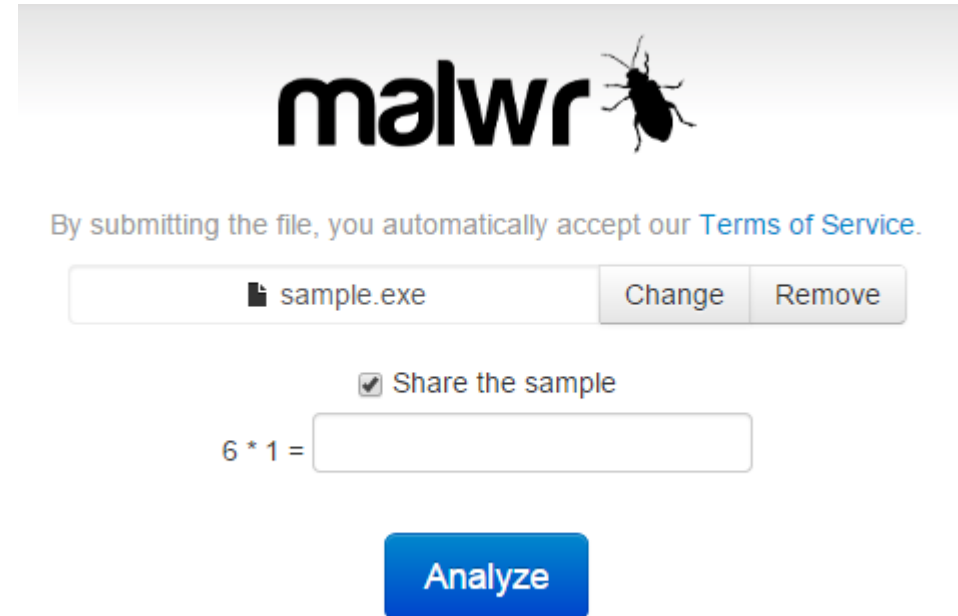
netsh firewall add allowedprogram "

MD5: 5d347384ea978a96bc842ad9f29e95f2

Automated sandbox analysis

Automated sandboxing

- Automated
 1. Execute malware in sandbox
 2. Wait a few seconds
 3. Receive summary report
 4. Investigate report
- Non-interactive
- Known tools
 - Cuckoo, Norman, Anubis etc.



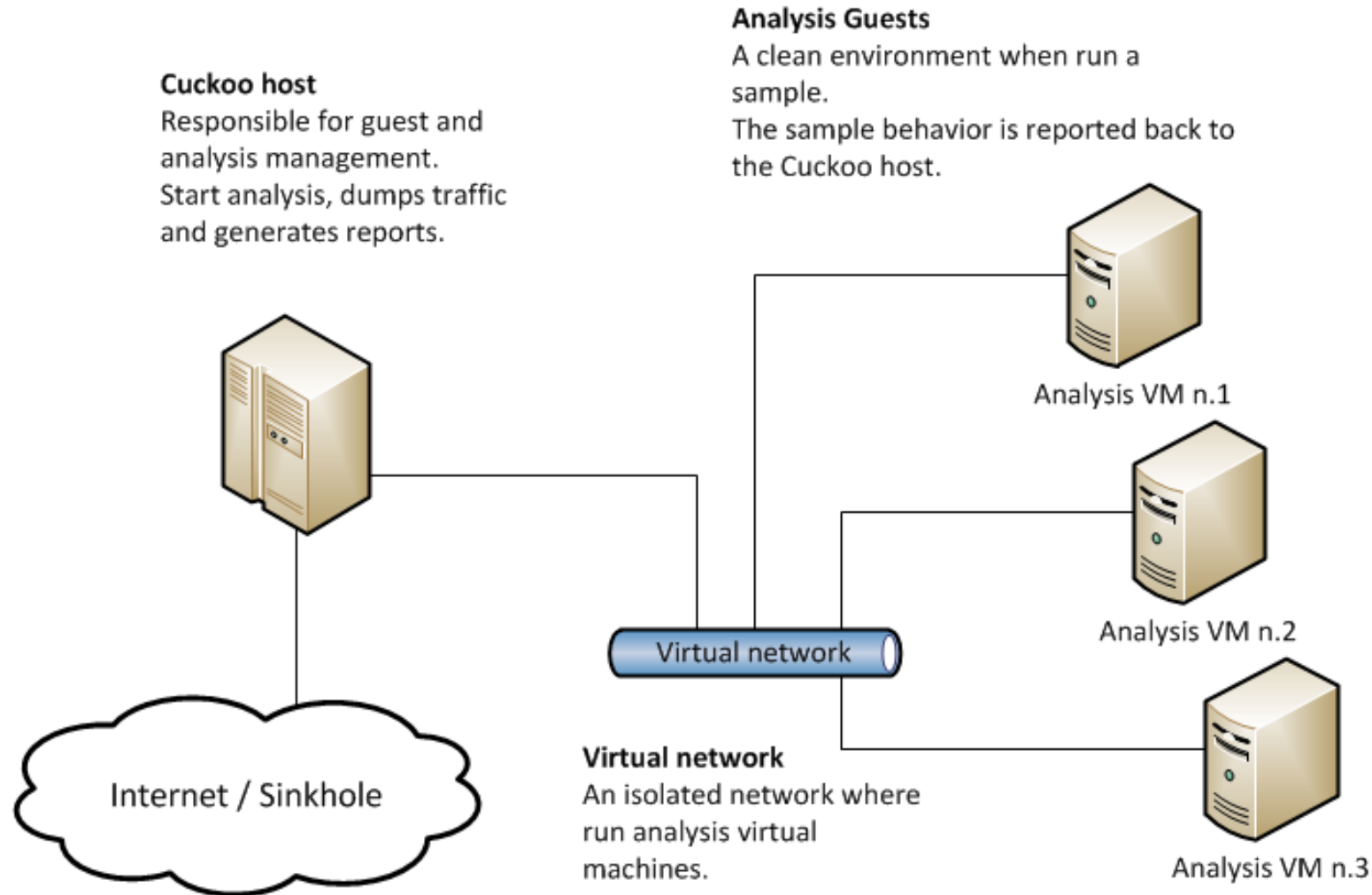
The screenshot shows the 'malwr' website interface. At the top, the logo 'malwr' is displayed next to a black beetle icon. Below the logo, a grey banner contains the text: 'By submitting the file, you automatically accept our [Terms of Service](#).' Underneath this banner is a file upload area with a text box containing 'sample.exe', a 'Change' button, and a 'Remove' button. Below the file upload area is a checkbox labeled 'Share the sample' which is checked. Underneath the checkbox is a CAPTCHA question '6 * 1 =' followed by an empty input box. At the bottom of the form is a blue button labeled 'Analyze'.

Cuckoo sandbox



- Open source malware analysis system
- Can analyze
 - Windows executables, DLLs, PDF documents, URLs, HTML files, PHP scripts, Visual Basic scripts, ZIP archives, Python files, etc.
- Modular, scriptable
- Full memory dump (for Volatility Framework)
- Django web interface
- Mongo (NoSQL) database

Cuckoo – Architecture



Cuckoo – GUI

| | | | | | | | |
|------|------|------------|-------------|--------|---------|---------|----------|
| Info | File | Signatures | Screenshots | Static | Dropped | Network | Behavior |
|------|------|------------|-------------|--------|---------|---------|----------|

| Category | Started On | Completed On | Duration | Cuckoo Version |
|----------|---------------------|---------------------|-------------|----------------|
| FILE | 2013-05-09 20:47:13 | 2013-05-09 20:49:56 | 163 seconds | 0.5 |

File Details file indicators

| | |
|--------------------------|--|
| File name | 7351eaaa39eb672c00c1dbe1e525a9e0 |
| File size | 303104 bytes |
| File type | PE32 executable (GUI) Intel 80386 Mono/.Net assembly, for MS Windows |
| CRC32 | D45DD4BC |
| MD5 | 7351eaaa39eb672c00c1dbe1e525a9e0 |
| SHA1 | f5f06f53f270f1fd044da1da9eea5b59794bc346 |
| SHA256 | 078ae46df0b431c7d423568495ee01caaf9d024aaf880061c739cfcb4dbf4490 |
| SHA512 | 950a5e85b4f161578660179eb2afe95798edaebf1b2998702c1250fea613c3b95b9143e643994ebad67e08702ddab47a6accb4b25c9f2d7a3d19fa3ca1b8cbf7 |
| Ssdeep | None |
| PEiD Signatures | None matched |
| Yara Signatures | |
| Antivirus Results | 25/46 (collapse) |

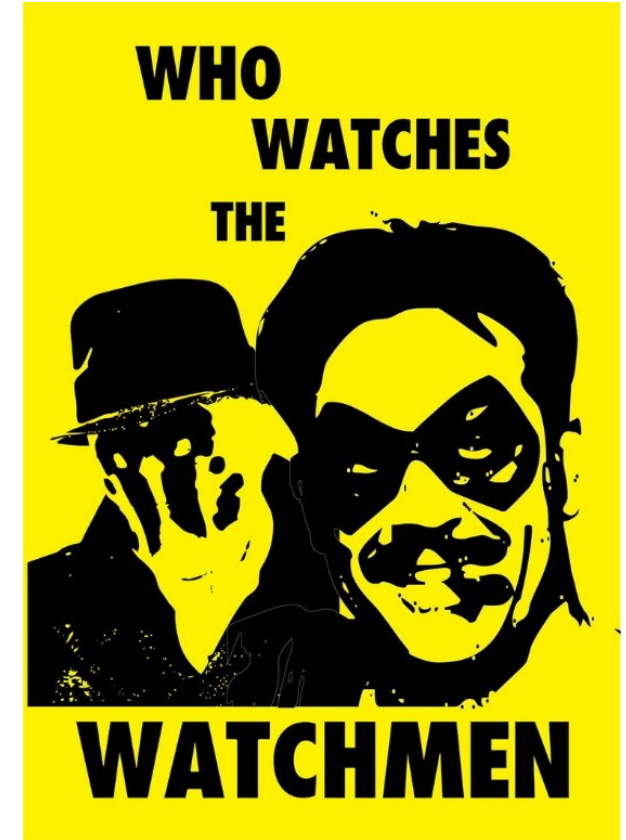
Internet sandbox services

- Public service
 - OpSEC issues
 - Huge comparison database
 - Exact match by hash
 - Similarity search by keywords
-
- Malwr.com (public Cuckoo sandbox)
 - VirusTotal.com
 - ThreatExpert.com



Operational security (OpSec)

- Advanced **attackers monitor** victim's actions
 - Unique indicators visible on Google?
 - Attacker host monitoring for incoming traffic
 - Keywords search in mails, PDFs...
 - VirusTotal uploads
- Basics of OpSec
 - “Think before you act” mentality
 - Limited information sharing
 - Trace removal



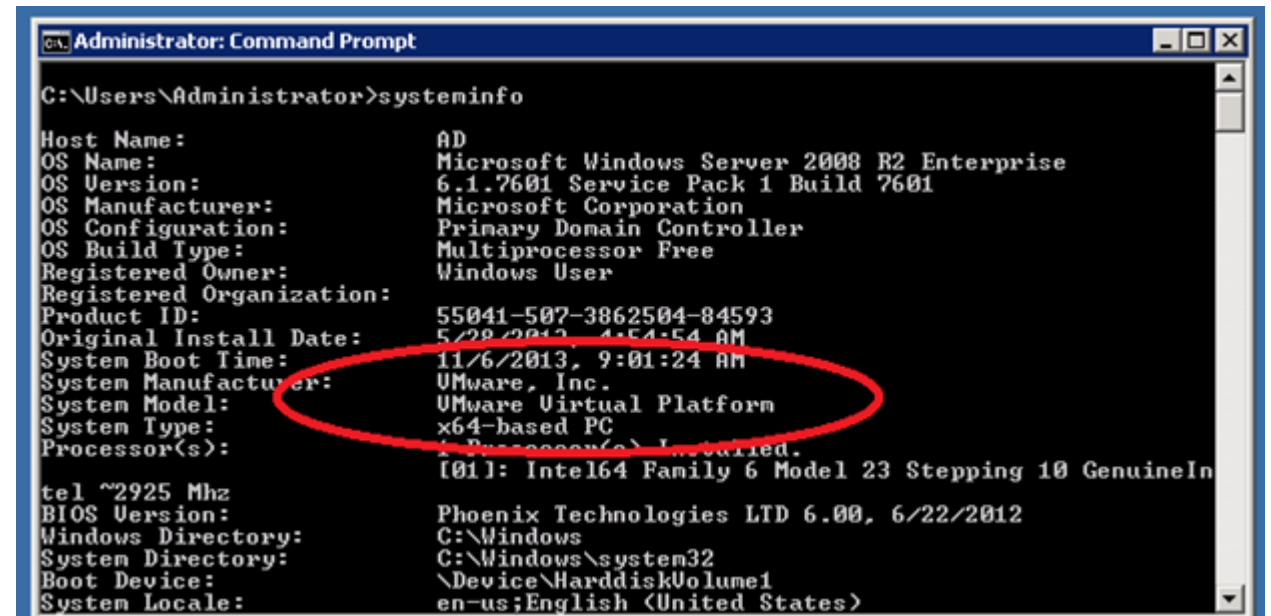
OpSec – Basic rules

- No ping
- No DNS lookup
- No accessing to suspicious domains
- No premature remediation steps (reboot, antivirus scan, OS reinstall)
- No upload of samples
- No indicator validation on external sources

- NOT EVEN through 3rd parties

Anti-sandbox techniques

- Continuous development – sandbox vs. anti-sandbox
- Malware inactive in analysis environment
- Tools presence detection (Wireshark, etc.)
- Virtualization detection
 - Registry (key existence, key value)
 - File system (file existence, drivers)
 - Processes (syscall response)
- Human presence detection
 - Mouse movement
 - Keyboard activity
 - File artefacts



```
Administrator: Command Prompt
C:\Users\Administrator>systeminfo
Host Name:                AD
OS Name:                  Microsoft Windows Server 2008 R2 Enterprise
OS Version:               6.1.7601 Service Pack 1 Build 7601
OS Manufacturer:        Microsoft Corporation
OS Configuration:        Primary Domain Controller
OS Build Type:             Multiprocessor Free
Registered Owner:         Windows User
Registered Organization:
Product ID:                55041-507-3862504-84593
Original Install Date:    5/28/2013, 4:54:54 AM
System Boot Time:         11/6/2013, 9:01:24 AM
System Manufacturer:      VMware, Inc.
System Model:              VMware Virtual Platform
System Type:               x64-based PC
Processor(s):              1 Processor(s) Installed.
                          [01]: Intel64 Family 6 Model 23 Stepping 10 GenuineIn
tel ~2925 Mhz
BIOS Version:              Phoenix Technologies LTD 6.00, 6/22/2012
Windows Directory:        C:\Windows
System Directory:          C:\Windows\system32
Boot Device:                \Device\HarddiskVolume1
System Locale:              en-us;English (United States)
```


Document analysis – Quick insight

- File metadata
- EXIF information on pictures
 - Creator, creation time, photo source, photo GPS
- exiftool
- Document sandboxing possible
- Document interpretation ambiguity
- Practical examples
 - Double extensions, different content in different viewers, code block obfuscation & hiding

Lab exercise

Lab – Overview

- Hands-on experience of manual black-box analysis
- Guided analysis of selected malware samples
- Tools
 - Wireshark – Network activity
 - Process Monitor – File system activity, process creation
 - Autoruns – Persistence
 - Process explorer – Process map

Lab – Samples

- 2-3 samples from different malware families
 - Commodity malware – Zeus, ZeroAccess, Generic Trojans,...
- Students will execute samples in virtual environment
 - Provided simple analysis virtual machine (Windows)
 - Indicators collected – network, files, persistence
 - Discussion about interpretation of facts
- Homework
 - Samples for analysis independently
 - Write a cohesive report and present key information to the reader