Zero Day Malware Cleaning with the Sysinternals Tools

Mark Russinovich
Technical Fellow
Windows Azure
About Me

- Technical Fellow, Windows Azure, Microsoft
- Co-founder and chief software architect of Winternals Software
- Author of Windows Sysinternals tools
- Co-author of Windows Internals book series
  - With Dave Solomon
- Co-author of Sysinternals Administrator’s Reference
  - With Aaron Margosis
- Author of Zero Day: A Novel

Copyright © 2011 Mark Russinovich
About this Talk

- Learn about Sysinternals tools and techniques for analyzing and cleaning malware
  - Professional antimalware analysis requires
  - But even for professionals, Sysinternals tools can prove useful

Analyzing:
- Understanding the impact of malware
- Can be used to understand malware operation
- Generates road map for cleaning infestations

Cleaning:
- Removing an infestation of a compromised system
- Attempting a clean can also reveal more information about malware’s operation

Copyright © 2011 Mark Russinovich
Malware Cleaning Steps

- Disconnect from network
- Identify malicious processes and drivers
- Suspend and terminate identified processes
- Identify and delete malware autostarts
- Delete malware files
- Reboot and repeat
Identifying Malware Processes
What Are You Looking For?

Investigate processes that...

- ...have no icon
- ...have no description or company name
- ...unsigned Microsoft images
- ...live in Windows directory or user profile
- ...are packed
- ...include strange URLs in their strings
- ...have open TCP/IP endpoints
- ...host suspicious DLLs or services
What About Task Manager?

Task Manager provides little information about images that are running.
Process Explorer

- Process Explorer is “Super Task Manager”
- Has lots of general troubleshooting capabilities:
  - DLL versioning problems
  - Handle leaks and locked files
  - Performance troubleshooting
  - Hung processes
- We’re going to focus on its malware cleaning capabilities
Process Explorer 2010 Updates

Versions 12 and 14 included many enhancements, big and small:

- Network and disk activity
- Multi-tab system information
- Tree CPU usage
- Improved DLL scanning algorithm
- Command-lines in process tooltips
- Svchost information
- Service threads
- .NET assembly information
- Support for > 64
More precise CPU accounting

- Task Manager, Resource Monitor and older Process Explorer versions use time-slice accounting
  - Whatever thread is executing at a timer tick (typically 15.6ms) is charged for the entire time slice
  - Charge is kernel mode if thread is in kernel mode, user mode for user mode

- Process Explorer v14.1 uses cycle counts
  - Full cycle count usage on Win7/Server 2008 R2 because of new API
  - On Vista uses cycle counts to detect < time slice
  - On XP, uses context switches to detect < time slice

- Sub 0.01 usage is shown as < 0.01
Process Explorer v15: GPU Monitoring

- Captures GPU utilization and memory usage
  - System-wide
  - Per-Process
The Process View

- The process tree sort shows parent-child relationships
- Icon, description, and company name are pulled from image version information
  - Most malware doesn’t have version information
  - What about malware pretending to be from Microsoft?
    - We’ll deal with that shortly...
- Use the Window Finder (in the toolbar) to associate a window with its owning process
- Use the Search Online menu entry to lookup unknown processes
  - But malware often uses totally random or pseudo-random names

Copyright © 2011 Mark Russinovich
Refresh Highlighting

- Refresh highlighting highlights changes
  - Red: process exited
  - Green: new process
- Change duration (default 1 second) in Options
- Press space bar to pause and F5 to refresh
- Cause display to scroll to make new processes visible with Show New Processes option
- We’ll see how to spot short-lived processes later...
Process-type Highlights

- Blue processes are running in the same security context as Process Explorer
- Pink processes host Windows services (we’ll look at services shortly)
- Purple highlighting indicates an image is “packed”
  - Packed can mean compressed or encrypted
  - Malware commonly uses packing (e.g. UPX) to make antivirus signature matching more difficult
  - Packing and encryption also hides strings from view
- There are a few other colors, but they’re not important for malware hunting

Copyright © 2011 Mark Russinovich
Tooltips

- Process tooltips show the full path to the process image
- Malware more often hides behind Svchost, Rundll32 and Dllhost
  - Tooltip for Rundll32 processes shows hosted DLL
  - Dllhost tooltip shows hosted COM server
  - Tooltip for service processes shows hosted services
    - Services covered in detail shortly...
Detailed Process Information

- Double-click on a process to see detailed information
- Image tab:
  - Description, company name, version (from .EXE)
  - Full image path
  - Command line used to start process
  - Current directory
  - Parent process
  - User name
  - Start time
Windows Services

- Services can start when the system boots and run independently of the logged-on user
  - Examples include IIS, Themes, Server, Workstation, ...
  - Can run as their own process or as a service DLL inside a Svchost.exe

- The services tab shows detailed service information:
  - Registry name (HKLM\System\CurrentControlSet\Services\...)
  - Display name
  - Description (optional)
  - DLL path (for Svchost DLLs)
All (well, most) Microsoft code is digitally signed
- Hash of file is signed with Microsoft’s private key
- Signature is checked by decrypting signed hash with the public key

You can selectively check for signatures with the Verify button on the process image tab
- Select the Verify Image Signatures option to check all
- Add the Verified Signer column to see all

Note that verification will connect to the Internet to check Certificate Revocation List (CRL) servers

Copyright © 2011 Mark Russinovich
Sigcheck and ListDlls

- Scan the system for suspicious executable images
  
  ```
  sigcheck -e -u -s c:\
  ```

- Look for same characteristics as suspicious processes
  - Be especially wary of items in the \Windows directory
  - Investigate all unsigned images

- ListDlls will scan running processes for unisigned DLLs
  
  ```
  listdlls -u
  ```

Copyright © 2011 Mark Russinovich
Strings

- On-disk and in-memory process strings are visible on the Strings tab
  - There’s only a difference if the image is compressed or encrypted
- Strings can help provide clues about unknown processes
  - Look for URLs, names and debug strings
- You can also dump strings with the command-line Strings utility from Sysinternals

```
strings <file>
```
The DLL View

- Malware can hide as a DLL inside a legitimate process
  - We’ve already seen this with Rundll32 and Svchost
  - Typically loads via an autostart
  - Can load through “dll injection”
  - Packing highlight shows in DLL view as well

- Open the DLL view by clicking on the DLL icon in the toolbar
  - Shows more than just loaded DLLs
  - Includes .EXE and any “memory mapped files”

- Can search for a DLL with the Find dialog
- DLL strings are also viewable from the DLL menu

Copyright © 2011 Mark Russinovich
Loaded Drivers

- There are several tools for viewing configured drivers:
  - Start->Run->Msinfo32
  - Built-in SC command: sc query type= driver
  - Device Manager with View->Show Hidden Devices
- Process Explorer DLL view for the System process shows loaded drivers
  - Even drivers that delete their image files
  - Same path and version info as standard DLL view
- Usually they’re not stoppable
  - Delete their files and autostart settings later

Copyright © 2011 Mark Russinovich
Terminating Malicious Processes

- Don’t kill the processes
  - Malware processes are often restarted by watchdogs
- Instead, suspend them
  - Note that this might cause a system hang for Svchost processes
  - Record the full path to each malicious EXE and DLL
- After they are all asleep then kill them
  - Watch for restarts with new names...
Cleaning Autostarts
Investigating Autostarts

Windows XP Msconfig (Start->Run->Msconfig) falls short when it comes to identifying autostarting applications

- It knows about few locations
- It provides little information
Autoruns

- Shows standard system extension points
  - Standard Run keys and Startup folders
  - Shell, userinit
  - Services and drivers
  - Tasks
  - Winlogon notifications
  - Explorer and IE addins (toolbars, Browser Helper Objects, ...)
  - More and ever growing...

- Each startup category has its own tab and all items display on the Everything tab
  - Startup name, image description, company and path

Copyright © 2011 Mark Russinovich
How Autoruns Works

- Many different formats and rules for extension points
- Shared scan routine for common types
- Disabling moves an entry to a subkey or folder named AutorunsDisabled
Identifying Malware Autostarts

- Zoom-in on add-ons (including malware) by selecting these options:
  - Verify Code Signatures
  - Hide Microsoft Entries
- Select an item to see more in the lower window
  - Online search unknown images
  - Double-click on an item to look at where its configured in the Registry or file system
- Has other features:
  - Can display other profiles
  - Can also show empty locations (informational only)
  - Includes compare functionality
  - Includes equivalent command-line version, Autorunsc.exe

Copyright © 2011 Mark Russinovich
Analyzing Offline Systems

- Autoruns includes support for scanning offline systems
The Case of the Son’s Adware

- Web page automatically opened on logon after father got laptop back from son
- http://www.e-markettop.com/

- Tried running Malwarebytes, but it would immediately close

Copyright © 2011 Mark Russinovich
The Case of the Son’s Adware (Cont)

- Process Explorer showed one unsigned process, Crystal.exe:

  ![Process Explorer Screenshot]

- After suspending Crystal, Malwarebytes ran to completion
  - No malware reported
  - McAfee didn’t report any malware, either
The Case of the Son’s Adware (Cont)

- Ran Autoruns and found Crystal in the Run key:

<table>
<thead>
<tr>
<th>Key</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>HKEY_LOCAL_MACHINE\SOFTWARE\Wow6432Node\Microsoft\Active Setup\Installed Components</td>
<td>Windows Mail</td>
</tr>
<tr>
<td>Microsoft Corporation</td>
<td>c:\Program Files\Windows Mail\winmail.exe</td>
</tr>
<tr>
<td>HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\Run</td>
<td>Crystal.exe</td>
</tr>
<tr>
<td></td>
<td>c:\Users\Jeff\App Data\Roaming\Crystal.exe</td>
</tr>
<tr>
<td>HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\Run</td>
<td>nwiz</td>
</tr>
<tr>
<td></td>
<td>... c:\Program Files\NW\nwiz.exe</td>
</tr>
</tbody>
</table>

- Disabled it, rebooted and system operated normally: problem solved

- Web search revealed that it was the Bifrost trojan: http://comprolive.com/remove/trojan/bifrost/crystal.exe-usb.exe-cleaner.exe

Copyright © 2011 Mark Russinovich
The Case of the Scareware

A user’s father-in-law complained that there was an application that wouldn’t exit:
The Case of the Scareware (Cont)

When a user tries to run Task Manager or MsConfig, they get errors, even in Safe Mode:
The Case of the Scareware: Solved

- Ran Autoruns and one entry stood out as suspicious
  - No company name or description
  - Installed in user’s profile
  - Replaces shell

- Right-clicked and “jumped” to the Shell registry entry
  - Replaced malware name with “Explorer”
  - Rebooted

- Problem solved
The Case of the Unusable System


- Friend’s of a friend’s computer became infected with malware
- It was unable to run anything:
  - Could have used Safemode, but wanted to see if I could run something at logon before malware activated
    - Logged off and back on
    - Was able to run Process Explorer and Autoruns

Copyright © 2011 Mark Russinovich
The Case of the Unusable System: Solved

- Process Explorer had one unsigned, random-name, packed image in user's profile directory:

- Same one showed up twice in Autoruns:

- Used Sigcheck to look for other suspicious files: none
- Killed process, deleted autostarts: system clean

Copyright © 2011 Mark Russinovich
Deleting Autostarts

- Disable suspicious autostarts
  - First rule of troubleshooting: make reversible changes
- After you’re done do a full refresh
- If they come back, run Process Monitor to see who’s putting them back
  - You might have misidentified a malware process
  - It might be a hidden, system, or legitimate process
Tracing Malware Activity
Tracing Malware

- Tracing activity can reveal the system impact of malware
- Tracing shows initial infection, before cloaking is applied
- Can reveal the internals of "buddy system" and other infection-protection mechanisms
- Process Monitor makes tracing easy
  - A simple filter can identify all system modifications
  - Investigating stacks can distinguish legitimate activity from malicious activity
Process Monitor

- Process Monitor is a real-time file, registry, process and thread monitor
- It replaces Filemon and Regmon
  - More advanced filtering
  - Operation call stacks
  - Boot-time logging
  - Data mining views
  - Process tree to see short-lived processes

When in doubt, run Process Monitor!
- It will often show you the cause for error messages
- It many times tells you what is causing sluggish performance
How Process Monitor Works

- Process Monitor uses a device driver
  - Extracts the driver to \Windows\System32\Drivers
  - Installs the driver
  - Deletes the driver file
- Requires “Debug Programs” user right
  - First run requires the “Load Driver” user right

User Mode
Kernel Mode

Process Monitor UI

Process Monitor Driver

File System Filter
Registry Callback
Kernel Callouts
TCP/IP Driver ETW events
How Process Monitor Works (cont.)

- Registry monitoring:
  - On Windows 2000, Windows XP 32-bit: system-call hooking
  - On Windows XP 64-bit, Server 2003, and Vista: registry callback
- File system monitoring:
  - File system mini-filter driver (causes Windows version requirements)
- Image loads:
  - Memory-manager image load callback
- Process/Thread create/exit:
  - Kernel process/thread callback
- TCP/IP
  - ETW events emitted by TCPIP.sys
Event Classes

- File system (Filemon)
  - Includes I/O command input and output details
- Registry (Regmon)
  - Includes all data (First 16-bytes part of REG_BINARY and first 2048-bytes for other types)
- Process
  - Process create and exit
  - Thread create and exit
  - Image loads, including drivers
- Network
  - ETW network tracing
- Profiling
  - Toolhelp thread snapshots

Copyright © 2011 Mark Russinovich
Event Properties

- Event details
  - Duration, process, thread, details, etc.
- Process information
  - Command line
  - User
  - Session and logon session
  - Image information
  - Start time
- Thread stack at time of event

Copyright © 2011 Mark Russinovich
Filtering

- To filter on a value, right-click on the line and select the attribute from the Include, Exclude or Highlight submenus.
- You can select multiple values simultaneously.
- When you set a highlight filter you can move through highlighted event properties.
Demo: Watching a File Save

- Run Process Monitor, then:
  - Run Notepad
  - Type some text
  - Save the file as test.txt

- Find the real file save in the log file
  - Set a highlight filter on the saved file path
Symbols

- Download the latest Debugging Tools for Windows from Microsoft (free)

- Configure Process Monitor’s symbol engine:
  - Use dbghelp.dll from the Debugging Tools
  - Point at the Microsoft public symbol server

- To grab symbols for offline access, use Symchk, part of the Debugging Tools
Event Properties: Stack

- The stack tab shows the stack of the thread executing the operation
- Stack is function-call history
- Thread stacks can show root cause
- Uses symbol engine

```
Function 1
Function 2
Function 3
```

Stack Display

Copyright © 2011 Mark Russinovich
Analyzing a Stack

- Look at the function names and DLLs to identify the root cause
- Double-click on a line to see the DLL properties

Note: user stack capture isn’t supported on 64-bit versions of Windows XP/Server 2003
Analyzing Process Startup with Stacks

There are lots of file system I/Os and Registry operations during Notepad’s startup

Using the stack we can identify three phases:

- Prefetch
  - Reads in directories
  - Faults in DLLs
- DLL initialization
  - Loader walks import tables
- Application initialization
  - Initializes DLLs e.g. OLE
  - Reads global settings

Copyright © 2011 Mark Russinovich
Advanced Filters

- Multiple-filter behavior:
  - Values from different attributes are AND’ed
  - Values for the same attribute are OR’ed

- More complex filtering is available in the Filter dialog
- Outlook-style rule definition

- You can save and restore filters

- Filter for watching malware impact: “Category is Write”

Copyright © 2011 Mark Russinovich
The Process Tree

- Tools->Process Tree
  - Shows all processes that have been seen in the trace (including parents)
  - Can toggle on and off terminated processes

- The process tree provides an easy way to see process relationships
  - Short-lived processes
  - Command lines
  - User names
The Case of the Slow Project File Opens

- Customer reported that opens of Project files from a network were slow and 1 of 10 opens resulted in an error:
  
  ![Error Message]
  
  - There was a problem sending the command to the program.

- Microsoft support asked them to captured a Process Monitor trace.
The Case of the Slow Project File Opens (Cont)

- First observation: tremendous number of access to share because user-profiles stored there:

<table>
<thead>
<tr>
<th>Total Events</th>
<th>Path</th>
</tr>
</thead>
<tbody>
<tr>
<td>4,103</td>
<td>&lt;Total&gt;</td>
</tr>
<tr>
<td>2,038</td>
<td>\DBG.ADS.D.B.COM\LON-USER S-U\VFOS_USERS01\hossje-2\config\AppData\Microsoft\Office\Office12\WinNT transient files</td>
</tr>
</tbody>
</table>
Second observation: Symantec A/V prescans entire file:
The Case of the Slow Project File Opens: Solved

- Recommendation 1: Move user profile AppData folders to local system
- Recommendation 2: Disable local scanning of files on network share since the server also has antivirus
- After recommendations followed, no more issues: problem solved
The Case of the Sysinternals-Blocking Malware


- Friend asked user to take a look at system suspected of being infected with malware
  - Boot and logons took a long time
  - Microsoft Security Essentials (MSE) malware scan would never complete
  - Nothing jumped out in Task Manager
- Tried running Sysinternals tools, but all exited immediately after starting:
  - Autoruns
  - Process Monitor
  - Process Explorer
  - Even Notepad opening a text file named “Process Explorer” would also terminate

Copyright © 2011 Mark Russinovich
The Case of the Sysinternals-Blocking Malware (Cont)

- Looking through Sysinternals suite, noticed Desktops utility
  - Hoped malware might not be smart enough to monitor additional desktops

- Sure enough, was able to launch Process Monitor and other tools:
  - Malware probably looks for tools in window titles
  - Window enumeration only returns windows of current desktop

Copyright © 2011 Mark Russinovich
The Case of the Sysinternals-Blocking Malware (Cont)

- Nothing suspicious in Process Explorer
- Next, ran Process Monitor
  - Noticed a lot of Winlogon activity, so set a filter to include it
  - Could see a once-per-second check of a strange key:

![Image of Process Monitor]

- Saw name of random DLL in the key:

![Image of Registry Key]
The Case of the Sysinternals-Blocking Malware: Solved

- Tried deleting the key, but after refreshing it was back
- Went back to MSE and directed it to scan just the random DLL image file on disk:

![Microsoft Security Essentials Alert](image)

- After clean, was able to delete Registry key and system was back to normal: problem solved
The Case of the Malicious Autostart

- Microsoft Support got a report of Marioforever.exe malware spreading within a company
- Malware infected Winlogon:

```
C:\>listdlls winlogon -d nvrsmad1.dll
```

ListDLLs v2.25 - DLL lister for Win9x/NT
Copyright (C) 1997-2004 Mark Russinovich
Sysinternals - www.sysinternals.com

winlogon.exe pid: 416
Command line: winlogon.exe

<table>
<thead>
<tr>
<th>Base</th>
<th>Size</th>
<th>Version</th>
<th>Path</th>
</tr>
</thead>
<tbody>
<tr>
<td>0x10000000</td>
<td>0x34000</td>
<td></td>
<td>C:\WIN\DOW\S\ystem32\nvrsmad1.dll</td>
</tr>
</tbody>
</table>

- The malware didn’t show up in Autoruns, so how it loaded was a mystery

Copyright © 2011 Mark Russinovich
The Case of the Malicious Autostart (Cont)

- Captured a Process Monitor boot log and searched for nvrsm:

<table>
<thead>
<tr>
<th>winlogon.exe.exe</th>
<th>RegCloseKey</th>
<th>HKLM\SOFTWARE\Microsoft\Windows NT\CurrentVersion\CompatibilityWinlogon</th>
<th>SUCCESS</th>
</tr>
</thead>
<tbody>
<tr>
<td>winlogon.exe.exe</td>
<td>RegOpenKey</td>
<td>HKLM\Software\Microsoft\Windows\NT\CurrentVersion\Windows</td>
<td>SUCCESS</td>
</tr>
<tr>
<td>winlogon.exe.exe</td>
<td>RegQueryValue</td>
<td>HKLM\SOFTWARE\Microsoft\Windows\NT\CurrentVersion\Windows\bwpInit_DLLs</td>
<td>SUCCESS Type: REG_SZ Length: 14 Data: nvrsmar</td>
</tr>
<tr>
<td>winlogon.exe.exe</td>
<td>RegOpenKey</td>
<td>HKLM\System\CurrentControlSet\Control\Session Manager\SafeDllSearchMode</td>
<td>SUCCESS Desired Access: Read</td>
</tr>
</tbody>
</table>

- Reference is in “bwpInit_DLLs” key
  - That’s not a Windows autostart key
  - Similar to AppInit_DLLs, a legitimate key
  - Why was Winlogon loading DLLs referenced there?
Launched Autoruns again on infected system and User32.dll stood out:

Autorunsc confirmed that they were different:
The Case of the Malicious Autostart Solved

- Looked at DLL properties in Process Explorer
  - In-memory strings were the same
  - On-disk strings had one difference:

- Rebooted system into Microsoft Diagnostic and Repair Toolkit and replaced User32.dll with good version: Malware cleaned

Copyright © 2011 Mark Russinovich
Basic vs Advanced Mode

- Basic mode includes filters that exclude system activity
  - Process Monitor activity
  - Paging file
  - System process
  - NTFS metadata files

- Basic mode massages I/O names:
  - E.g. IRP_MJ_READ => ReadFile
Logging to a File

- By default, logging is page-file backed
  - Use Options->History Depth to avoid memory exhaustion
  - Or use Filter->Drop Filtered Events
- Use the File->Backing File dialog to specify logging to a file
  - Data is written in native as its captured
  - Log size is limited only by available disk space
  - Dialog also shows trace statistics

Copyright © 2011 Mark Russinovich
Running Process Monitor Before Logon

- Sometimes need to capture I/O or registry activity during boot, the logon or logoff process
  - Problem: when you logoff all your processes are terminated
- Solutions: Run Process Monitor in a different logon session
  - psexec -s -i -d
Boot Logging

- Process Monitor can capture all activity from very early in the boot process:
  - Options->Enable Boot Logging
- Capture continues through shutdown or you run Process Monitor
  - Log data is saved as raw data to %Windir%\Procmon.pmb
  - The next time you run Process Monitor it will offer to transform the data to a native PML log
- Enable Advanced Output to see all events
- Typical Windows Vista boot->Log in->Shutdown generates 1-2 million events
Stuxnet and Alureon
Analyzing a Stuxnet Infection


- Discovered June 2010 after it had spread for year
- Exploited 4 zero day Windows vulnerabilities
  - Print spooler for remote code execution
  - Shell link Explorer code execution from infected key
  - Win2K/Windows XP Win32k.sys privilege elevation
  - Windows 7 Task Scheduler privilege elevation
- Drivers signed by certificates stolen from RealTek and JMicron
- Rootkit code for Siemens Step 7 SCADA PLC for centrifuges
- Suspected to have targeted Iranian centrifuges used for Uranium enrichment at Natanz nuclear facility
  - Iran confirms in September 2010 that thousands were destroyed
  - Suspected to be created by Israel and US

Copyright © 2011 Mark Russinovich
Analyzing an Alureon Infection

- First malware to bypass 64-bit Windows Kernel-mode Code Integrity
  - Modifies MBR and boots system in “WinPE” mode – code integrity is off
  - Loads malicious unsigned Kdcom.dll from hidden sectors off end of volume
  - Modifies registry boot flags to hide WinPE mode so system continues to boot normally
  - Uses modified disk driver to mask modified MBR
- Took A/V several months to develop on-line cleaning
- Very little visible sign of infection

Copyright © 2011 Mark Russinovich
Summary and the Future
The Future of Malware

We’ve seen the trends:
- Malware that pretends to be from Microsoft or other legitimate companies
- Malware protected by sophisticated rootkits
- Malware that has stolen certificates

Cleaning is going to get much, much harder
- Targeted and polymorphic malware won’t get AV/AS signatures
- Malware can directly manipulate Windows structures to cause misdirection
- All standard tools will be directly attacked by malware
- There will be more un-cleanable malware
- Malware will adapt to a limited-user environment

You can’t know you’re infected unless you find a symptom
The bottom line is that prevention and containment is the best defense
Zero Day – A Novel

- A cyberthriller true to the science
- www.zerodaythebook.com
- Signing from 3-3:30 at the bookstore

http://www.youtube.com/watch?v=ucyMBYg9RWU
The Sysinternals Administrator’s Reference

- The official guide to the Sysinternals tools
  - Covers every tool, every feature, with tips
  - Written by markruss and aaronmar
  - Available in June

- Full chapters on the major tools:
  - Process Explorer
  - Process Monitor
  - Autoruns

- Other chapters by tool group
  - Security, process, AD, desktop, ...


Copyright © 2011 Mark Russinovich
References

- Sysinternals
  - Sysinternals Administrator’s Reference, by Mark Russinovich and Aaron Margosis, Microsoft Press
  - Mark’s Sysinternals Blog
  - Mark’s Webcasts
- Symantec Stuxnet Dossier
- Ars technica Stuxnet story
- Joe Johnson: Alureon: the first 64-bit Rootkit
- Windows Internals, by Mark Russinovich and David Solomon with Alex Ionescu, Microsoft Press

Copyright © 2011 Mark Russinovich