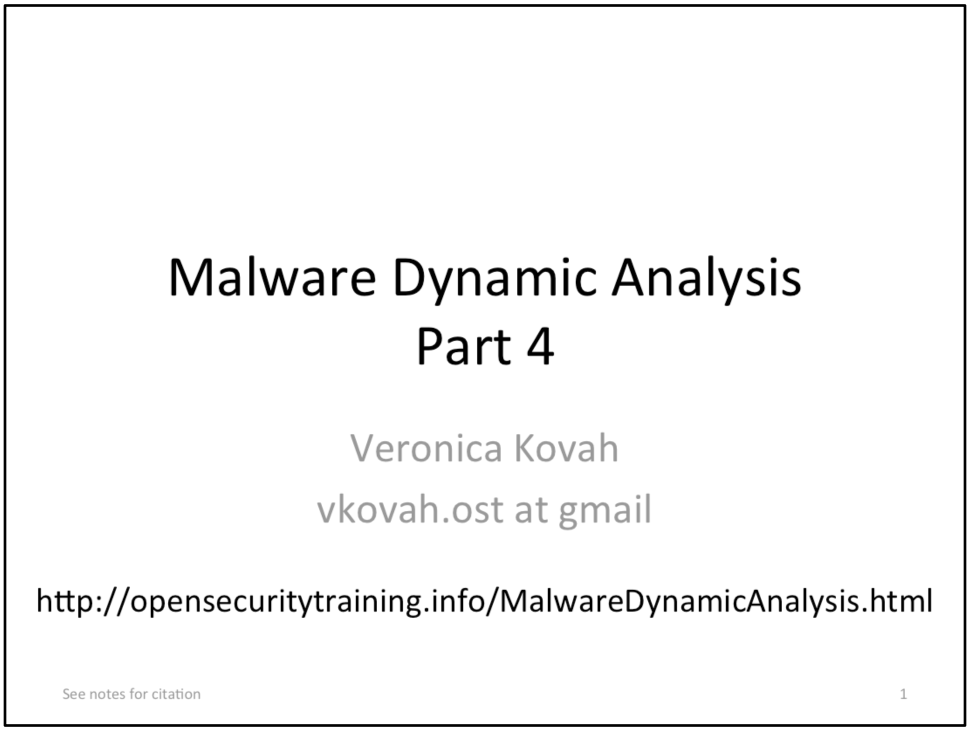


Malware Dynamic Analysis
Part 4

Veronica Kovah

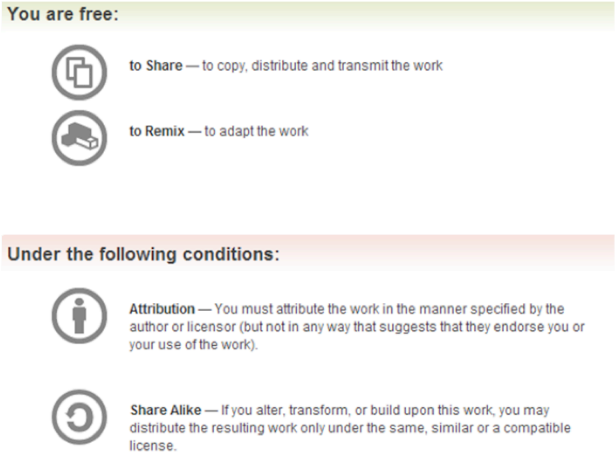
vkovah.ost at gmail

http://opensecuritytraining.info/MalwareDynamicAnalysis.html

See notes for citation

All materials is licensed under a Creative Commons “Share Alike” license

http://creativecommons.org/licenses/by-sa/3.0/



You are free:

to Share — to copy, distribute and transmit the work

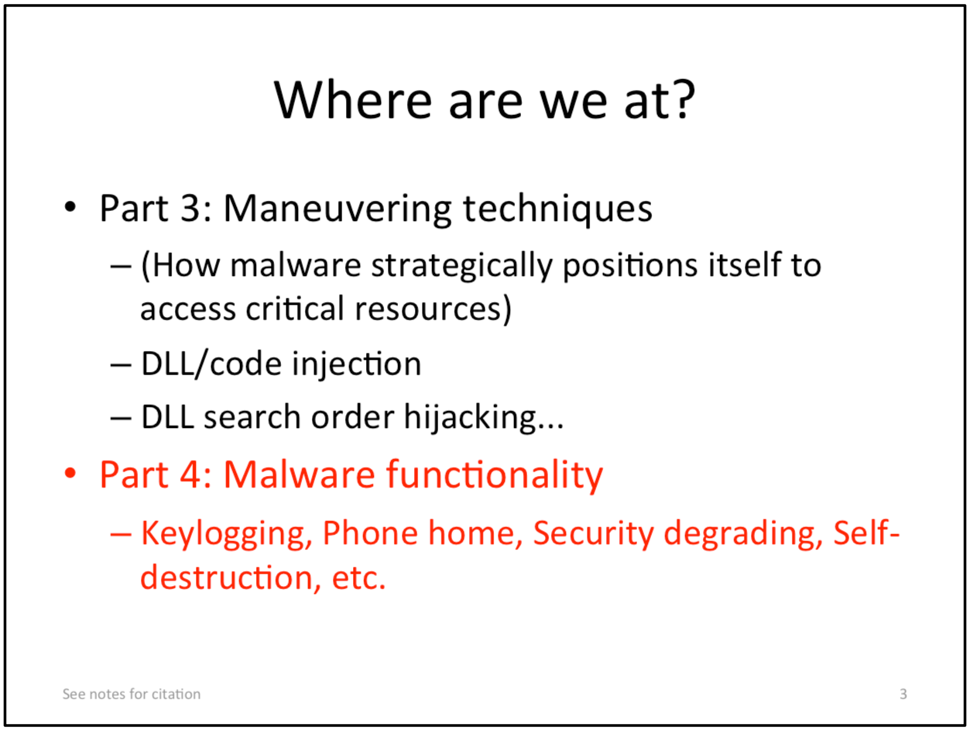
to Remix — to adapt the work

Under the following conditions:

Attribution — You must attribute the work in the manner specified by the author or licensor (but not in any way that suggests that they endorse you or your use of the work).

Share Alike — If you alter, transform, or build upon this work, you may distribute the resulting work only under the same, similar or a compatible license.

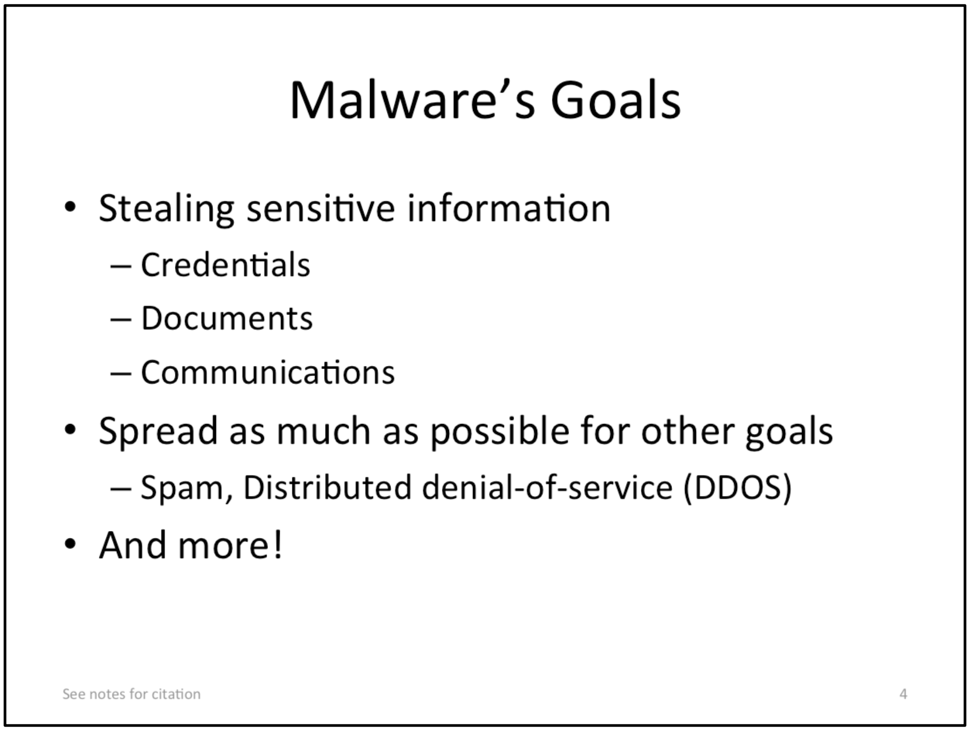
See notes for citation



Where are we at?

- Part 3: Maneuvering techniques
  - (How malware strategically positions itself to access critical resources)
  - DLL/code injection
  - DLL search order hijacking...
- Part 4: Malware functionality
  - Keylogging, Phone home, Security degrading, Self-destruction, etc.

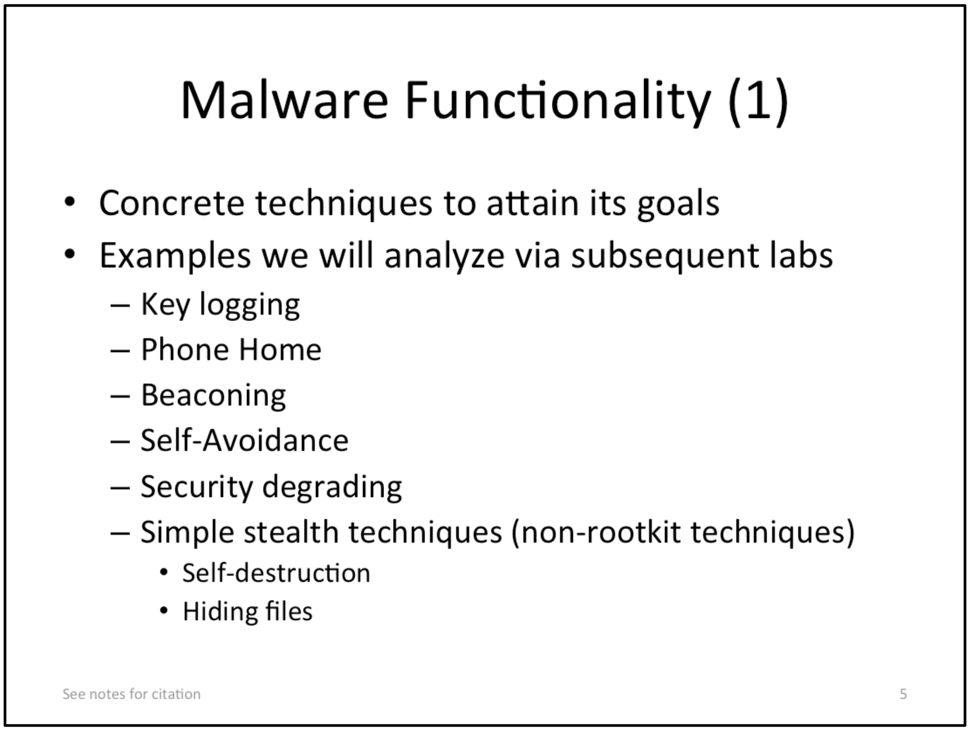
See notes for citation



Malware's Goals

- Stealing sensitive information
  - Credentials
  - Documents
  - Communications
- Spread as much as possible for other goals
  - Spam, Distributed denial-of-service (DDOS)
- And more!

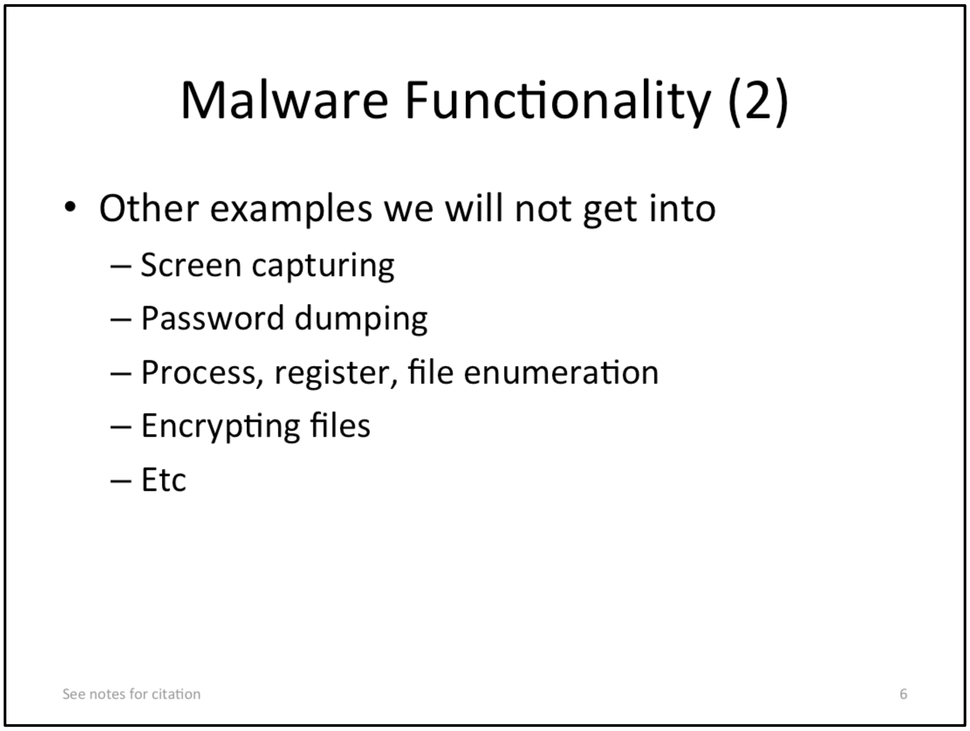
See notes for citation



Malware Functionality (1)

- Concrete techniques to attain its goals
- Examples we will analyze via subsequent labs
  - – Key logging
  - – Phone Home
  - – Beaconing
  - – Self-Avoidance
  - – Security degrading
  - – Simple stealth techniques (non-rootkit techniques)
    - Self-destruction
    - Hiding files

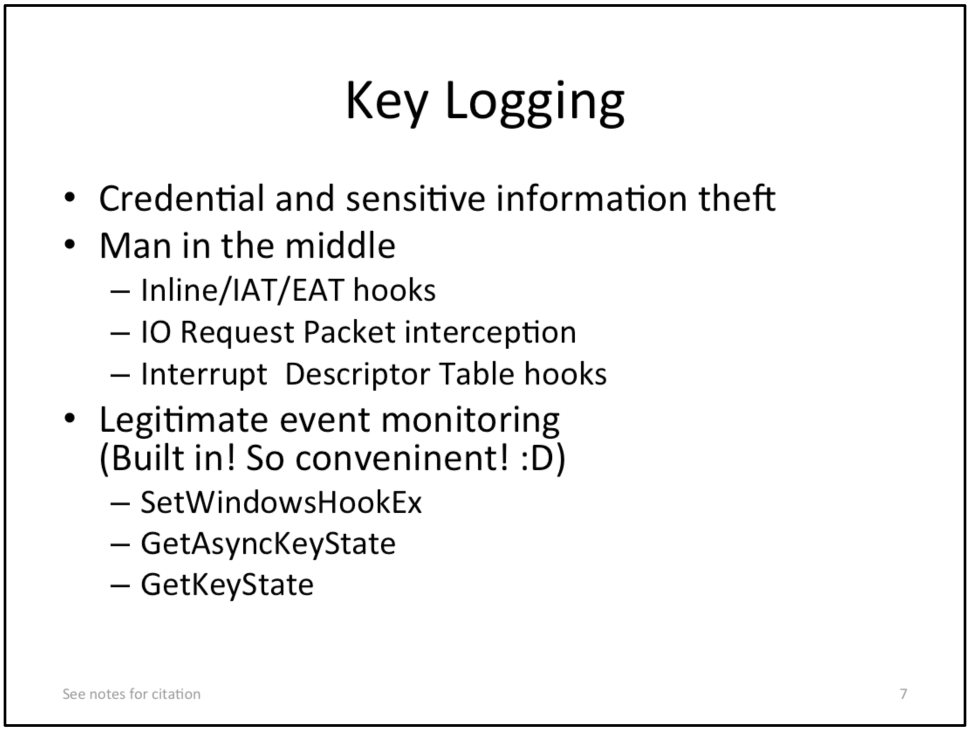
See notes for citation



Malware Functionality (2)

- Other examples we will not get into
  - Screen capturing
  - Password dumping
  - Process, register, file enumeration
  - Encrypting files
  - Etc

See notes for citation

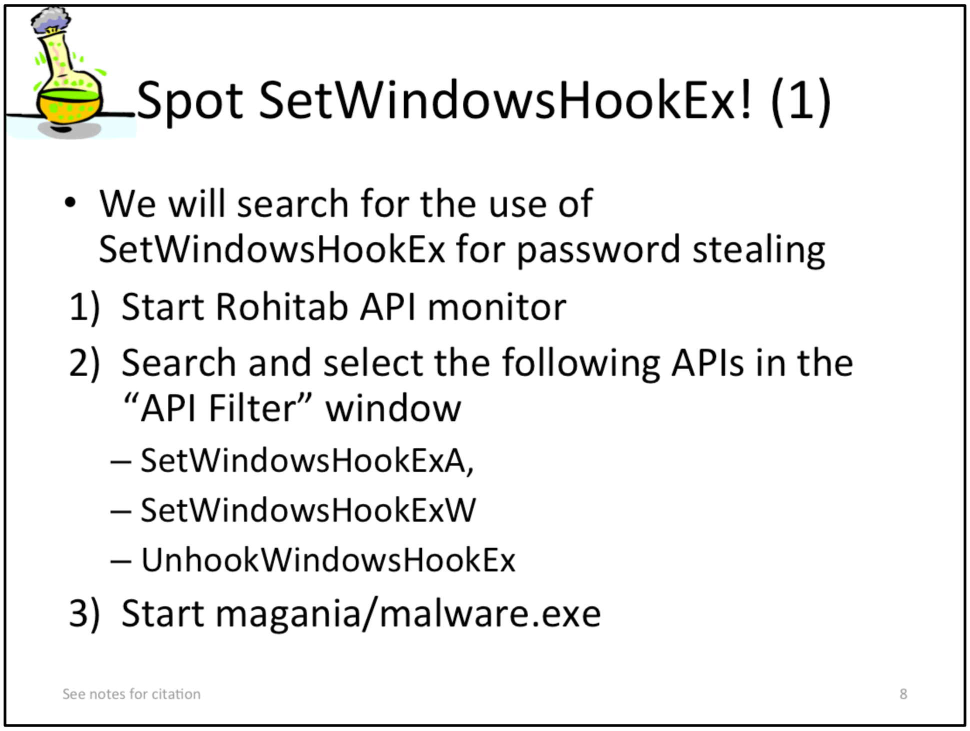


Key Logging

- Credential and sensitive information theft
- Man in the middle
  - Inline/IAT/EAT hooks
  - IO Request Packet interception
  - Interrupt Descriptor Table hooks
- Legitimate event monitoring (Built in! So conveninent! :D)
  - SetWindowsHookEx
  - GetAsyncKeyState
  - GetKeyState

[References]

- Emre TINAZTEPE, The Adventures of a Keystroke, http://opensecuritytraining.info/Keylogging.html
- Michael Sikorski et al., Chapter 11. Malware Behavior, Practical Malware Analysis
- Greg Hoglund et al., Chapter 8. Hardware Manipulation, Rootkits
- Bill Blunden, Chapter 8. Deploying Filter Drivers, The Rootkit Arsenal: Escape and Evasion

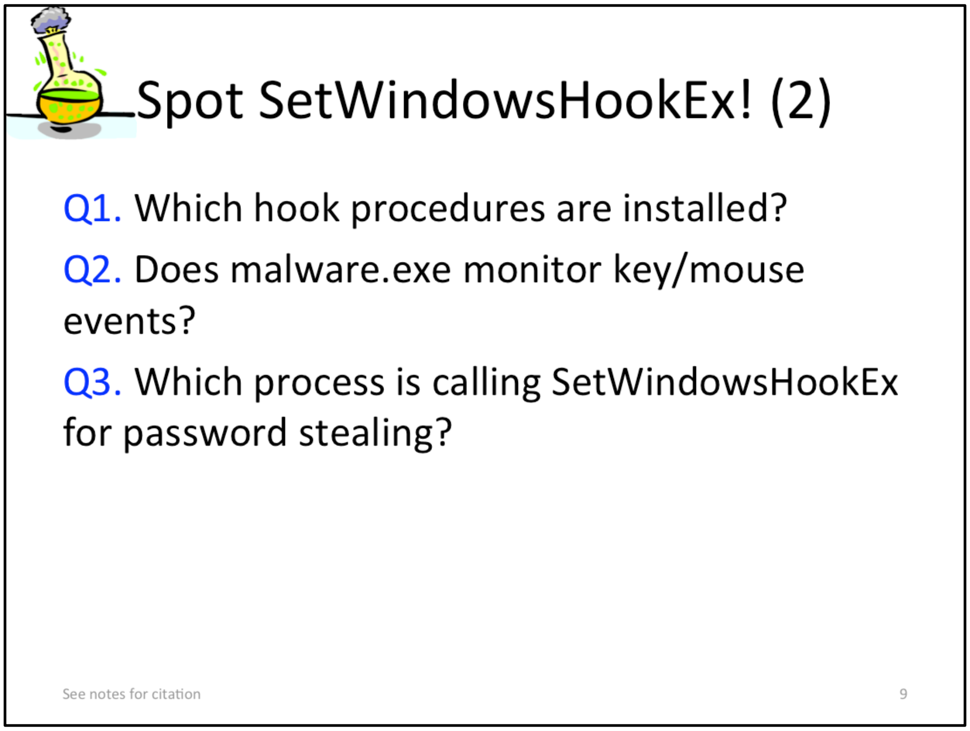


Spot SetWindowsHookEx! (1)

- We will search for the use of SetWindowsHookEx for password stealing

1) Start Rohitab API monitor
2) Search and select the following APIs in the "API Filter" window
   - SetWindowsHookExA,
   - SetWindowsHookExW
   - UnhookWindowsHookEx
3) Start magania/malware.exe

See notes for citation



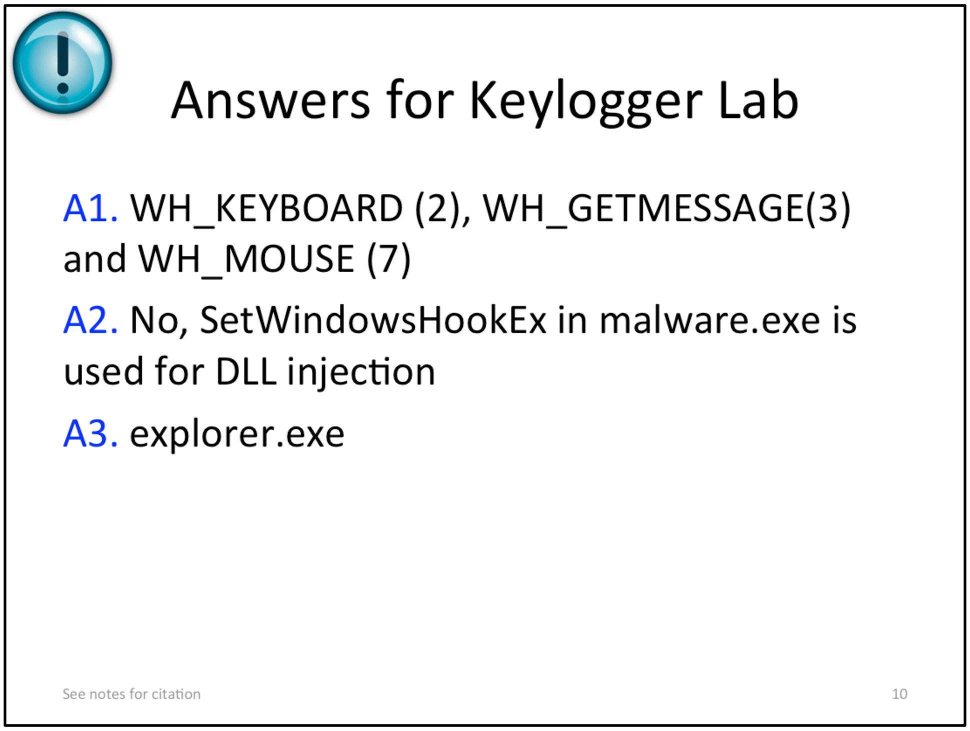
Spot SetWindowsHookEx! (2)

Q1. Which hook procedures are installed?

Q2. Does malware.exe monitor key/mouse events?

Q3. Which process is calling SetWindowsHookEx for password stealing?

See notes for citation



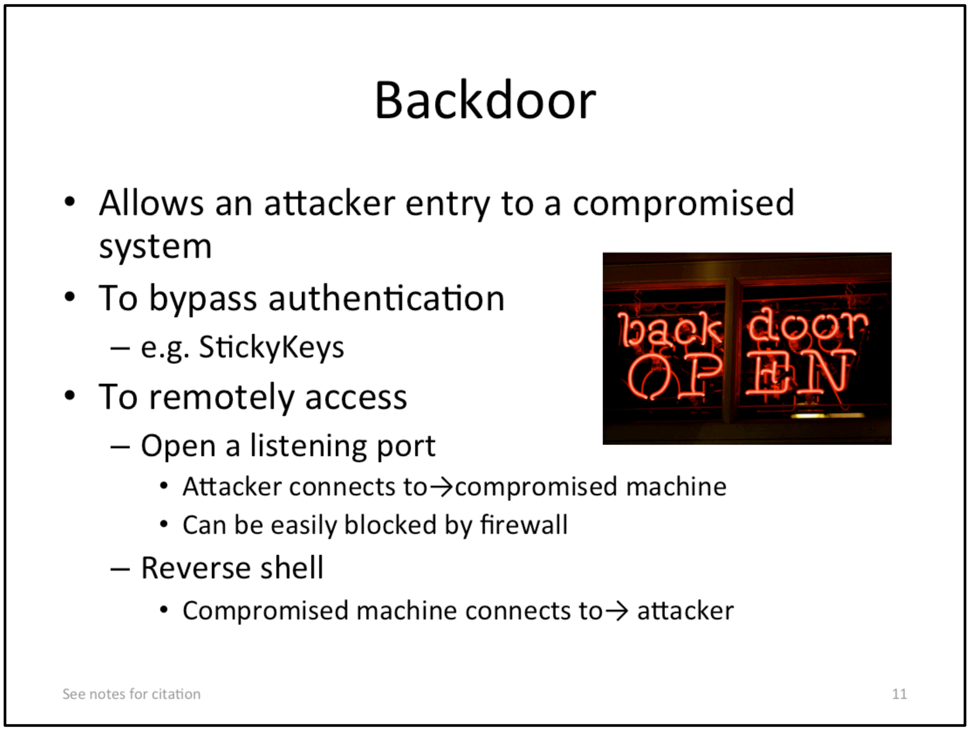
Answers for Keylogger Lab

A1. WH_KEYBOARD (2), WH_GETMESSAGE(3) and WH_MOUSE (7)

A2. No, SetWindowsHookEx in malware.exe is used for DLL injection

A3. explorer.exe

See notes for citation



Backdoor

- Allows an attacker entry to a compromised system
- To bypass authentication
  - e.g. StickyKeys
- To remotely access
  - Open a listening port
    - Attacker connects to→compromised machine
    - Can be easily blocked by firewall
  - Reverse shell
    - Compromised machine connects to→ attacker

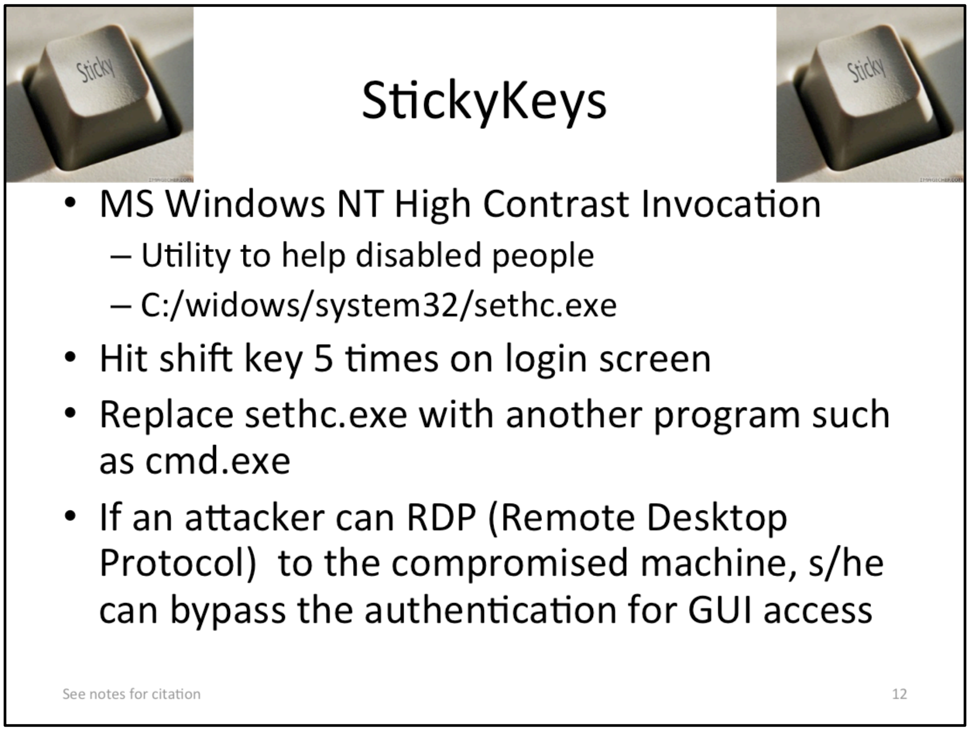
back door
OPEN

See notes for citation

11

[Image Sources]

- http://media.ascendworks.com/wp-content/uploads/backdoor.jpeg



StickyKeys

- MS Windows NT High Contrast Invocation
  - Utility to help disabled people
  - C:/widows/system32/sethc.exe
- Hit shift key 5 times on login screen
- Replace sethc.exe with another program such as cmd.exe
- If an attacker can RDP (Remote Desktop Protocol) to the compromised machine, s/he can bypass the authentication for GUI access

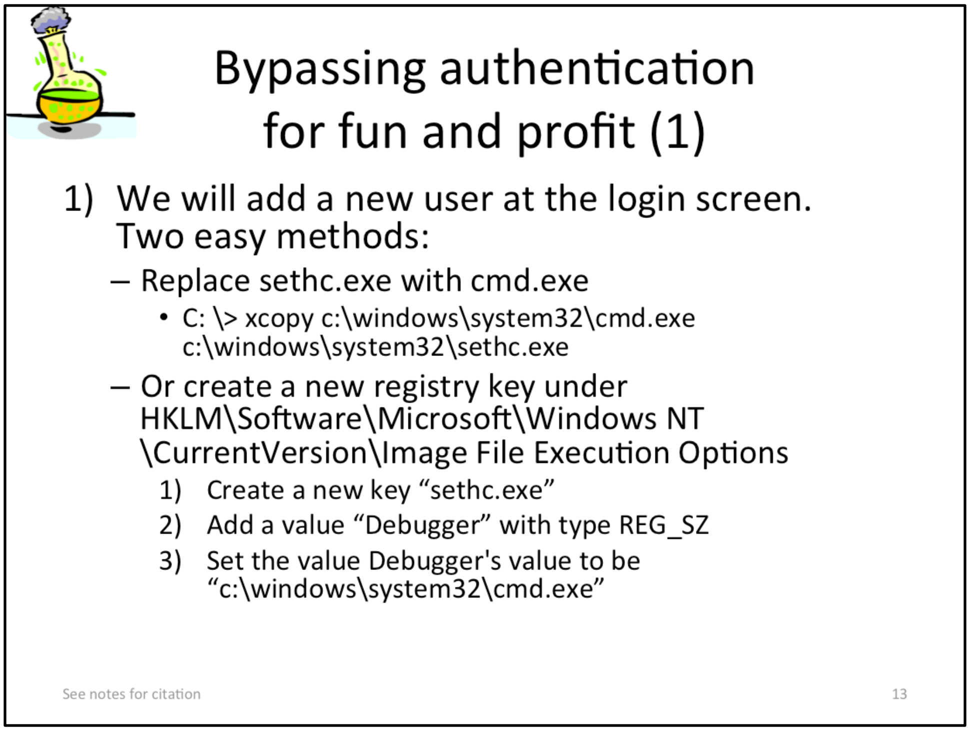
See notes for citation

[References]

- Windows Vista Vulnerable to StickyKeys Backdoor, http://blogs.mcafee.com/mcafee-labs/windows-vista-vulnerable-to-stickykeys-backdoor
- Ryan Kazanciyan, The “Hikit” Rootkit: Advanced and Persistent Attack Techniques (Part 1), https://blog.mandiant.com/archives/3155
- OmnipotentEntity, sethc.exe and Getting a SYSTEM Level Prompt Outside of Login, http://www.nerdparadise.com/tech/windows/sethcsystemlevelprompt/

[Image Sources]

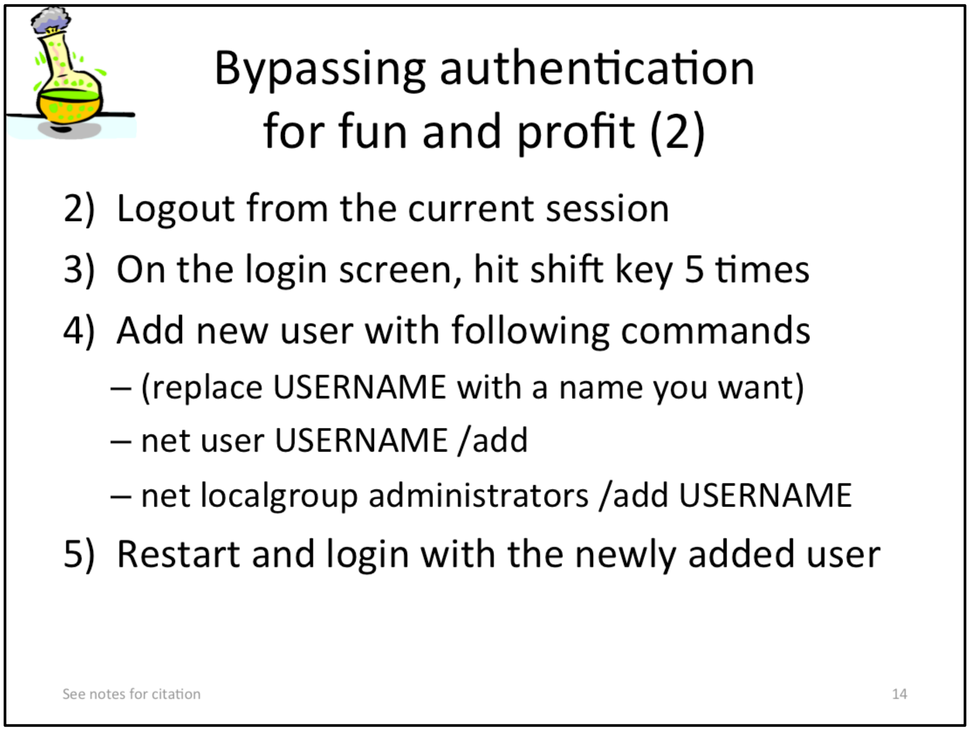
http://astoriedcareer.com/sticky_key.jpg



Bypassing authentication for fun and profit (1)

1) We will add a new user at the login screen. Two easy methods:
   - Replace sethc.exe with cmd.exe
     - C: \> xcopy c:\windows\system32\cmd.exe c:\windows\system32\sethc.exe
   - Or create a new registry key under HKLM\Software\Microsoft\Windows NT \CurrentVersion\Image File Execution Options
     1) Create a new key “sethc.exe”
     2) Add a value “Debugger” with type REG_SZ
     3) Set the value Debugger's value to be “c:\windows\system32\cmd.exe”

See notes for citation



Bypassing authentication for fun and profit (2)

2) Logout from the current session

3) On the login screen, hit shift key 5 times

4) Add new user with following commands

   – (replace USERNAME with a name you want)

   – net user USERNAME /add

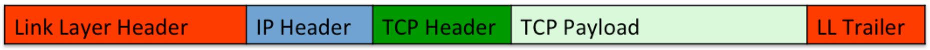
   – net localgroup administrators /add USERNAME

5) Restart and login with the newly added user

See notes for citation

Network Recap

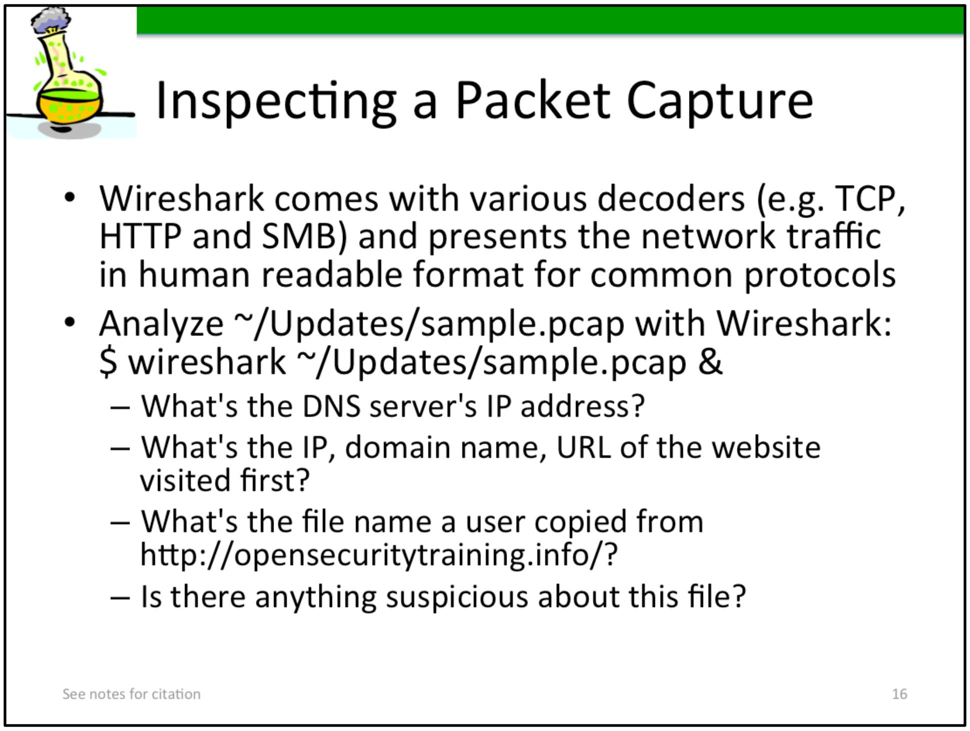
- Layered architecture



Link Layer Header | IP Header | TCP Header | TCP Payload | LL Trailer

- Common port list
  - HTTP (80), HTTPS (443), DNS (53), SMB (445)
  - http://www.iana.org/assignments/service-names-port-numbers/service-names-port-numbers.xml
- Connection initiator's port is usually randomly picked between 1024 and 2^{16} - 1
- Common open ports not blocked by firewall
  - DNS (UDP 53): inbound and outbound
  - HTTP (TCP 80, 8080): outbound

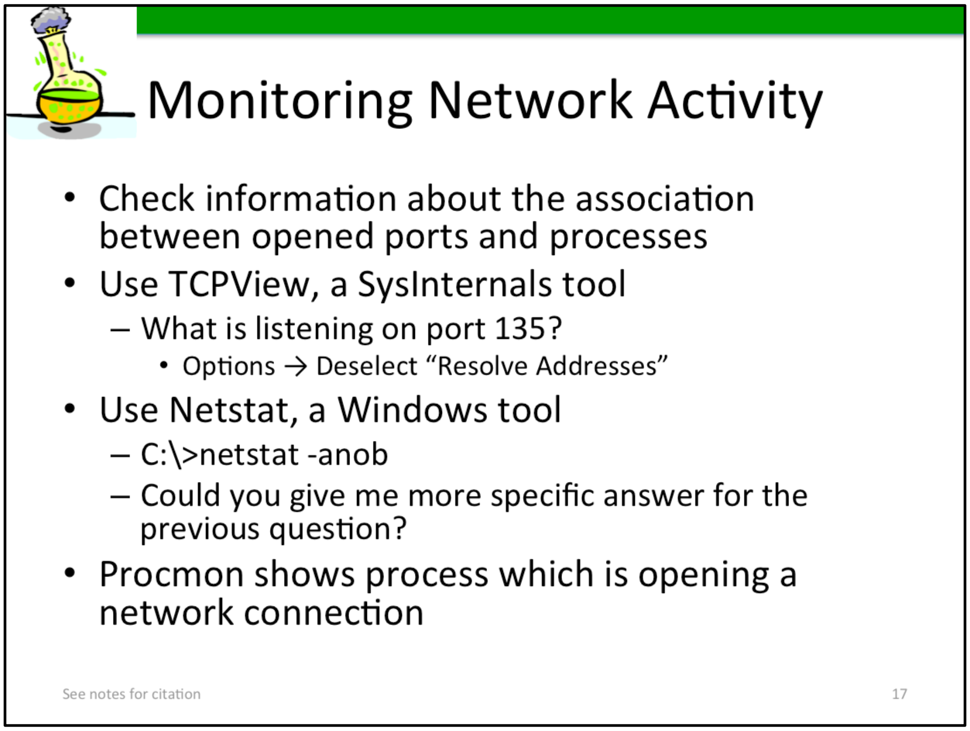
See notes for citation



Inspecting a Packet Capture

- Wireshark comes with various decoders (e.g. TCP, HTTP and SMB) and presents the network traffic in human readable format for common protocols
- Analyze ~/Updates/sample.pcap with Wireshark:
  ```
  $ wireshark ~/Updates/sample.pcap &
  ```
  - What's the DNS server's IP address?
  - What's the IP, domain name, URL of the website visited first?
  - What's the file name a user copied from http://opensecuritytraining.info/?
  - Is there anything suspicious about this file?

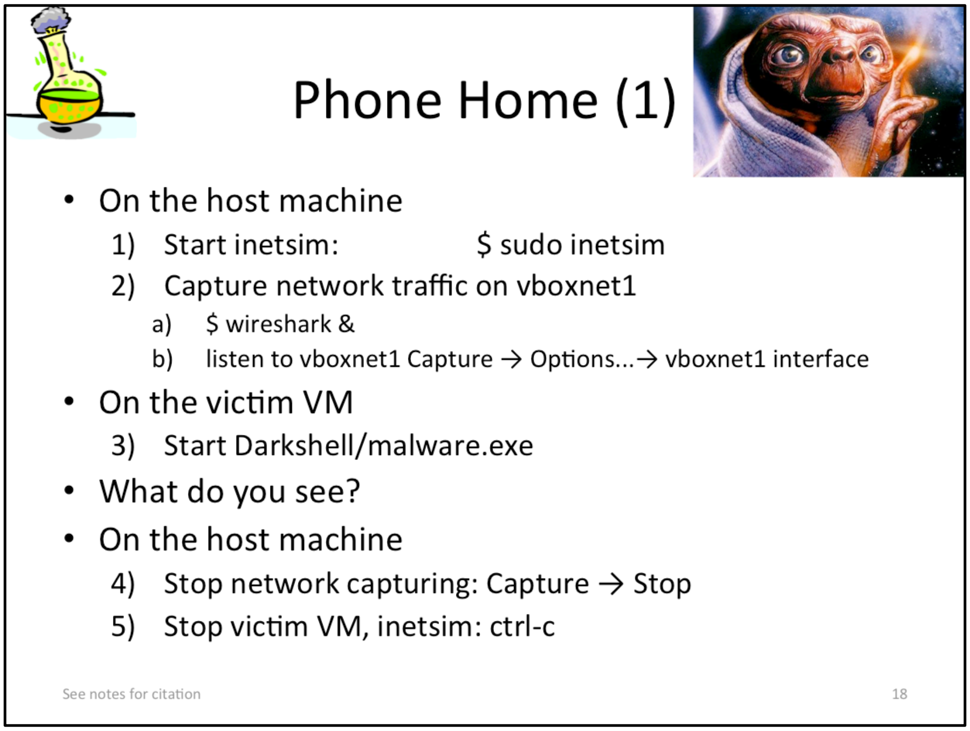
See notes for citation



Monitoring Network Activity

- Check information about the association between opened ports and processes
- Use TCPView, a SysInternals tool
  - What is listening on port 135?
    - Options → Deselect “Resolve Addresses”
- Use Netstat, a Windows tool
  - C:\>netstat -anob
  - Could you give me more specific answer for the previous question?
- Procmon shows process which is opening a network connection

See notes for citation



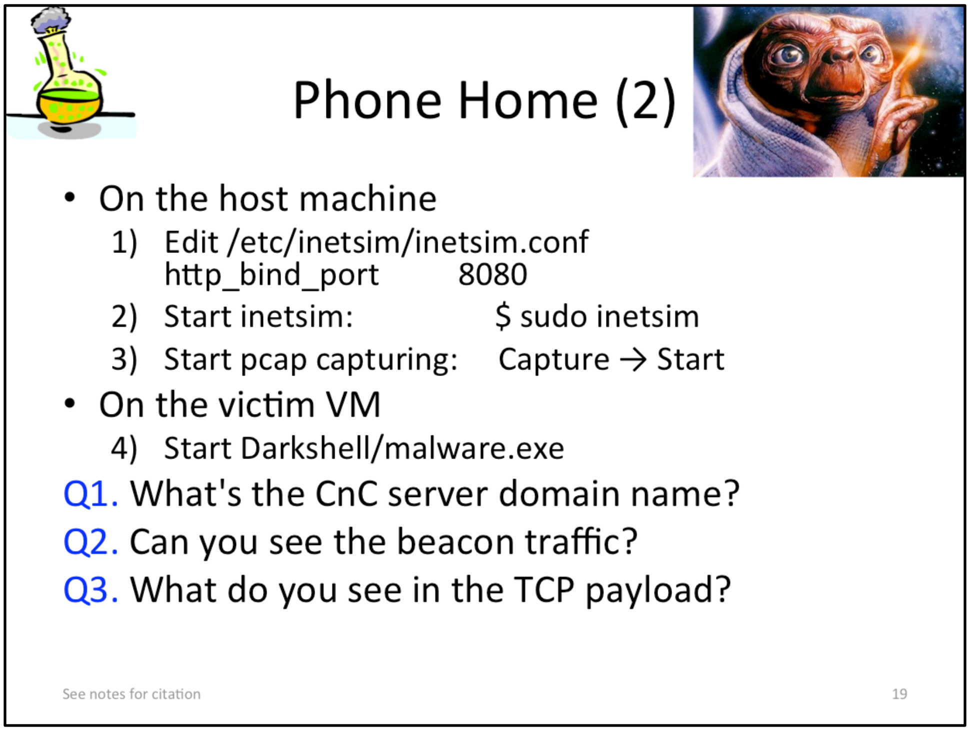
Phone Home (1)

- On the host machine
  1) Start inetsim: $ sudo inetsim
  2) Capture network traffic on vboxnet1
     a) $ wireshark &
     b) listen to vboxnet1 Capture → Options...→ vboxnet1 interface
- On the victim VM
  3) Start Darkshell/malware.exe
- What do you see?
- On the host machine
  4) Stop network capturing: Capture → Stop
  5) Stop victim VM, inetsim: ctrl-c

See notes for citation 18

[References]

- Jeff Edwards, Darkshell: A DDoS bot targetting vendors of industrial food-processing equipment, http://ddos.arbornetworks.com/2011/01/darkshell-a-ddos-bot-targetting-vendors-of-industrial-food-processing-equipment/



Phone Home (2)

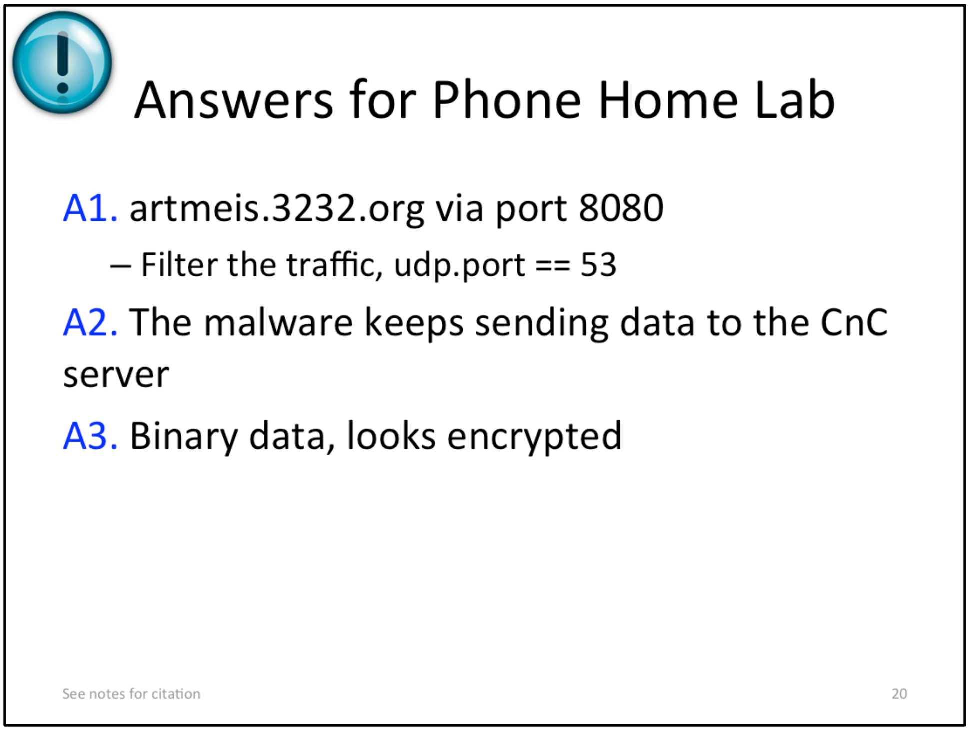
- On the host machine
  1) Edit /etc/inetsim/inetsim.conf
     http_bind_port 8080
  2) Start inetsim: $ sudo inetsim
  3) Start pcap capturing: Capture → Start
- On the victim VM
  4) Start Darkshell/malware.exe

Q1. What's the CnC server domain name?

Q2. Can you see the beacon traffic?

Q3. What do you see in the TCP payload?

See notes for citation



Answers for Phone Home Lab

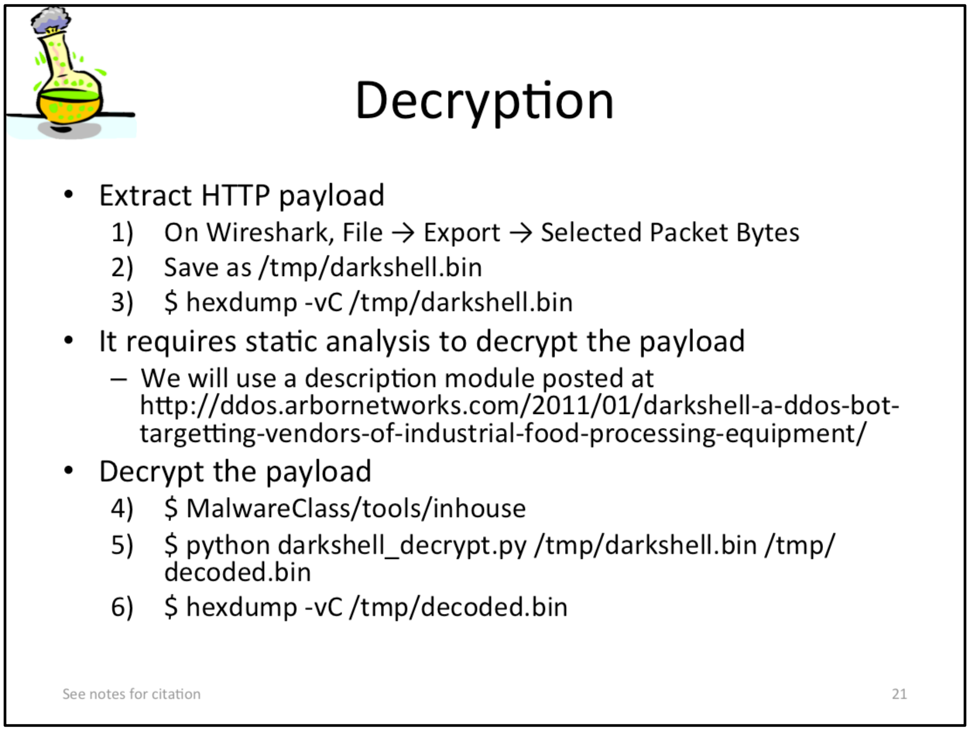
A1. artmeis.3232.org via port 8080

- Filter the traffic, udp.port == 53

A2. The malware keeps sending data to the CnC server

A3. Binary data, looks encrypted

See notes for citation



Decryption

- Extract HTTP payload
  1) On Wireshark, File → Export → Selected Packet Bytes
  2) Save as /tmp/darkshell.bin
  3) $ hexdump -vC /tmp/darkshell.bin
- It requires static analysis to decrypt the payload
  - We will use a description module posted at http://ddos.arbornetworks.com/2011/01/darkshell-a-ddos-bot-targetting-vendors-of-industrial-food-processing-equipment/
- Decrypt the payload
  4) $ MalwareClass/tools/inhouse
  5) $ python darkshell_decrypt.py /tmp/darkshell.bin /tmp/decoded.bin
  6) $ hexdump -vC /tmp/decoded.bin

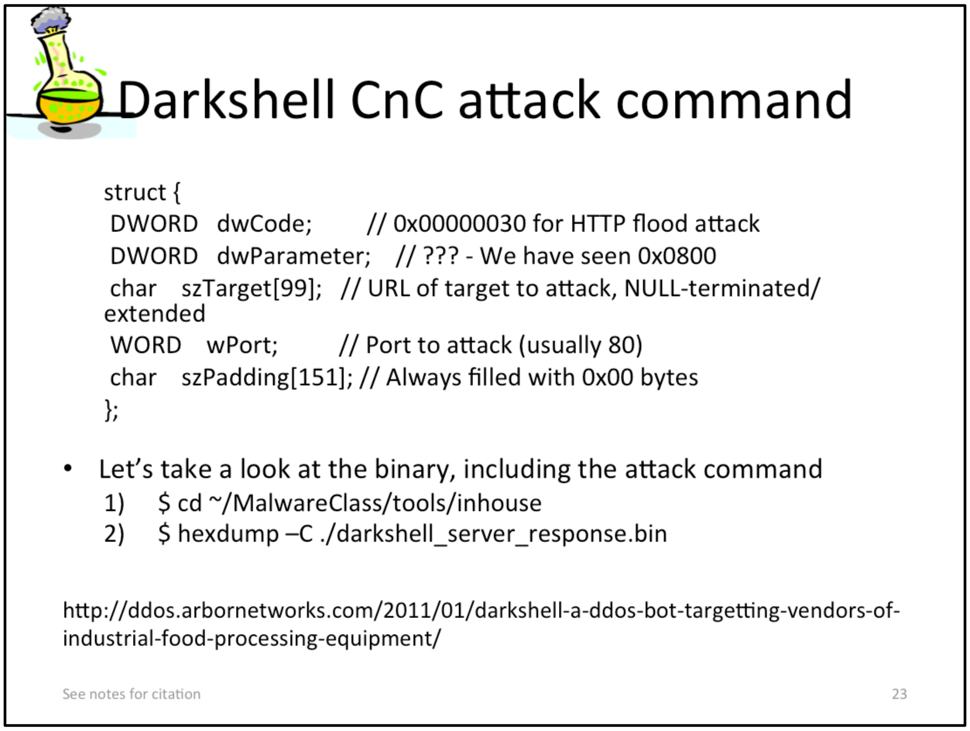
See notes for citation

Phone Home Phormat

```
// Darkshell bot-to-CnC comms
struct {
  // Header:
  DWORD   dwMagic;    // always 0x00000010 for Darkshell
  // Obfuscated section:
  char    szComputerName[64]; // Name of infected host, NULL-terminated/extended
  char    szMemory[32];    // Amount of memory in infected host; format "%dMB"; NULL-terminated/extended
  char    szWindowsVersion[32];   // Specifies version of Windows; one of: Windows98, Windows95,
                               // WindowsNT, Windows2000, WindowsXP, Windows2003, or Win Vista;
                               // NULL-terminated/extended
  char    szBotVersion[32];   // Specifies version of bot; NULL-terminated/extended;
  DWORD   szUnknown1[4];     // ??? - Always NULL-terminated 'n'
  // Binary section:
  char    szPadding1[32];  // Filled with 0x00 bytes
  WORD    wUnknown2;  // ??? - We have seen 0x00A0, 0x00B0, and 0x00C0
  WORD    wUnknown3;  // ??? - Always 0xFD7F
  char    szPadding2[20];  // Filled with 0x00 bytes
  WORD    wUnknown4;  // ??? - Always 0xB0FC
  BYTE    cUnknown5;  // ??? - We have seen 0xD6, 0xD7, 0xE6, 0xE7, and 0xF1
  BYTE    cZero;      // Always 0x00
  DWORD   dwSignature[8]; // Always 0x00000000, 0xFFFFFFFF, 0x18EE907C, 0x008E917C,
                       //       0xFFFFFFFF, 0xFA8D91&C, 0x25D6907C, 0xCFEA907C
};
```

http://ddos.arbornetworks.com/2011/01/darkshell-a-ddos-bot-targetting-vendors-of-industrial-food-processing-equipment/

See notes for citation



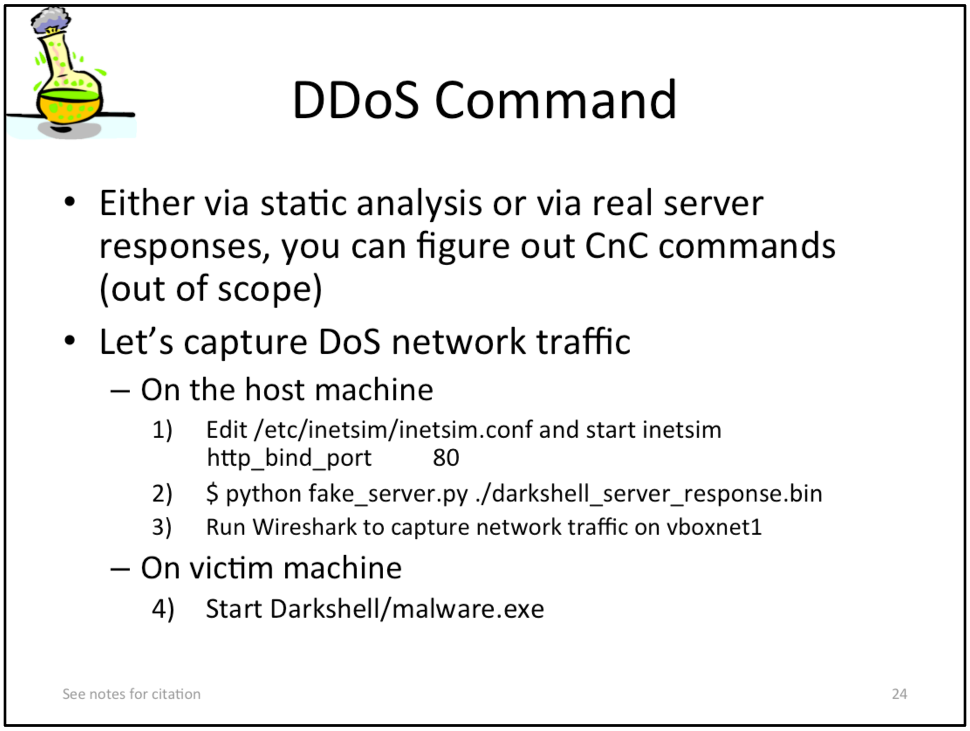
Darkshell CnC attack command

```
struct {
 DWORD   dwCode;        // 0x00000030 for HTTP flood attack
 DWORD   dwParameter;   // ??? - We have seen 0x0800
 char    szTarget[99];  // URL of target to attack, NULL-terminated/
extended
 WORD    wPort;         // Port to attack (usually 80)
 char    szPadding[151]; // Always filled with 0x00 bytes
};
```

- Let’s take a look at the binary, including the attack command
  1) $ cd ~/MalwareClass/tools/inhouse
  2) $ hexdump –C ./darkshell_server_response.bin

http://ddos.arbornetworks.com/2011/01/darkshell-a-ddos-bot-targetting-vendors-of-industrial-food-processing-equipment/

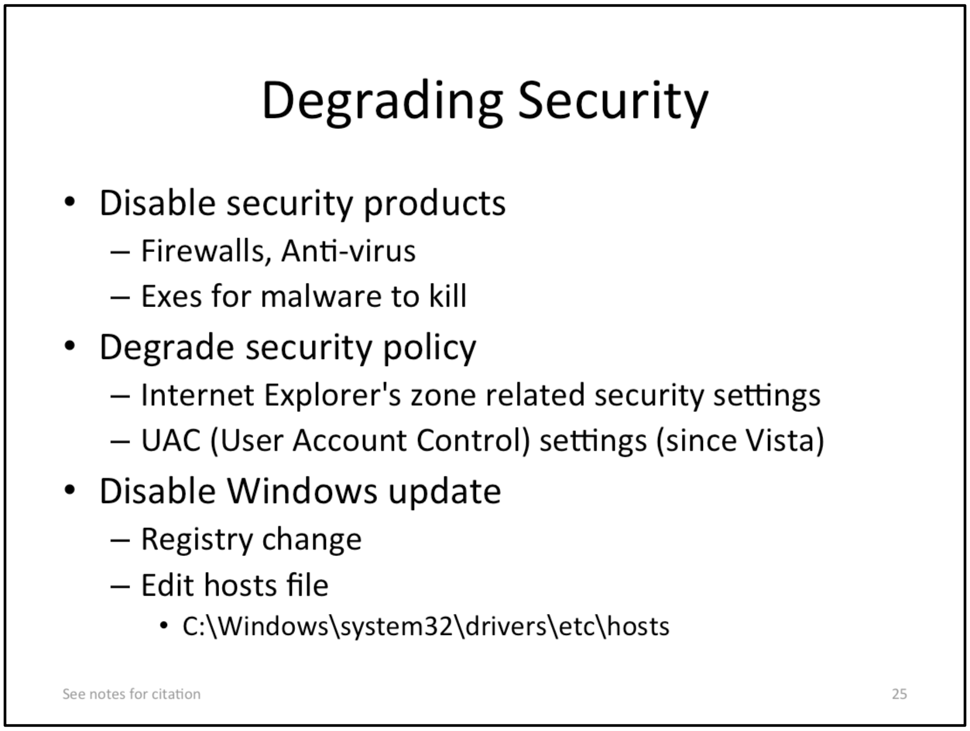
See notes for citation



DDoS Command

- Either via static analysis or via real server responses, you can figure out CnC commands (out of scope)
- Let's capture DoS network traffic
  - On the host machine
    1) Edit /etc/inetsim/inetsim.conf and start inetsim
       http_bind_port 80
    2) $ python fake_server.py ./darkshell_server_response.bin
    3) Run Wireshark to capture network traffic on vboxnet1
  - On victim machine
    4) Start Darkshell/malware.exe

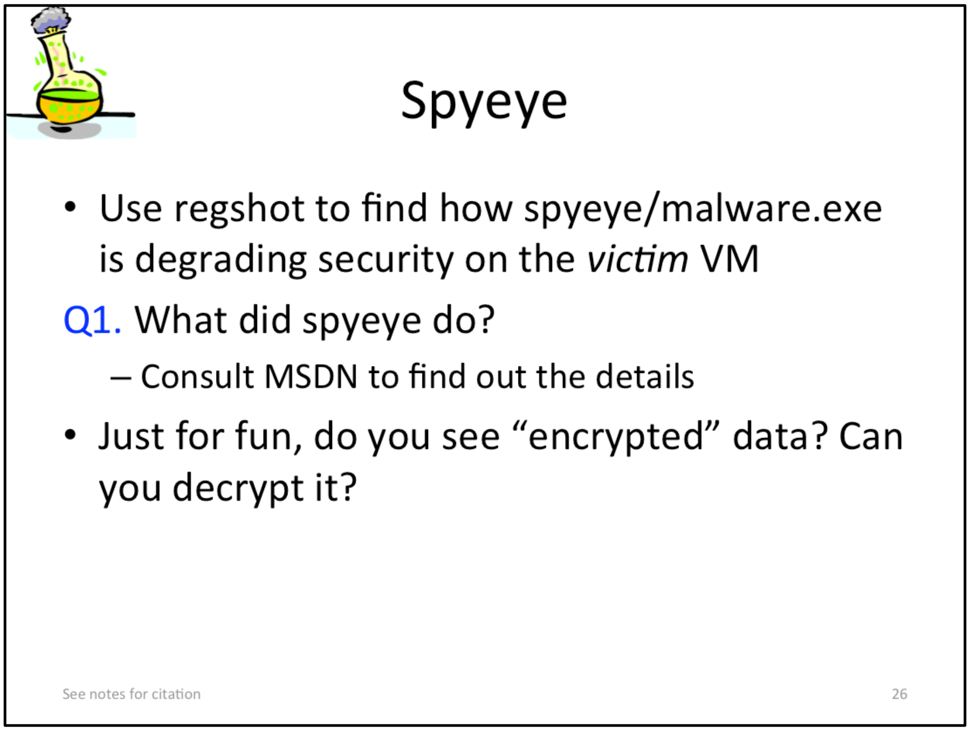
See notes for citation



Degrading Security

- Disable security products
  - – Firewalls, Anti-virus
  - – Exes for malware to kill
- Degrade security policy
  - – Internet Explorer's zone related security settings
  - – UAC (User Account Control) settings (since Vista)
- Disable Windows update
  - – Registry change
  - – Edit hosts file
    - C:\Windows\system32\drivers\etc\hosts

See notes for citation



Spyeye

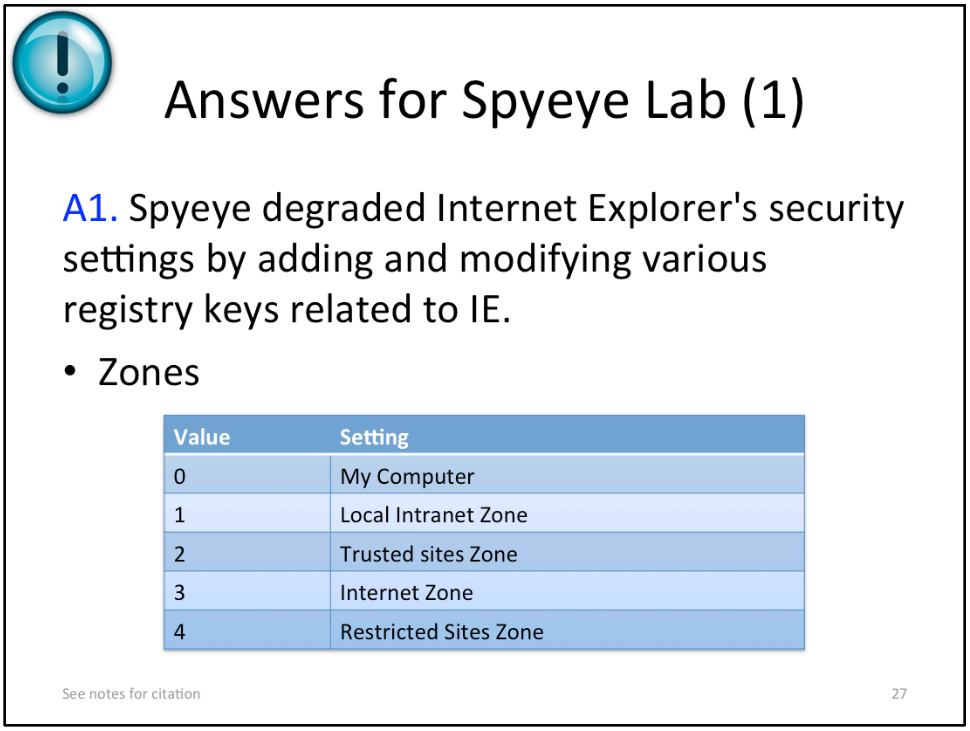
- Use regshot to find how spyeye/malware.exe is degrading security on the *victim* VM

Q1. What did spyeye do?

- Consult MSDN to find out the details

- Just for fun, do you see “encrypted” data? Can you decrypt it?

See notes for citation



Answers for Spyeye Lab (1)

A1. Spyeye degraded Internet Explorer's security settings by adding and modifying various registry keys related to IE.

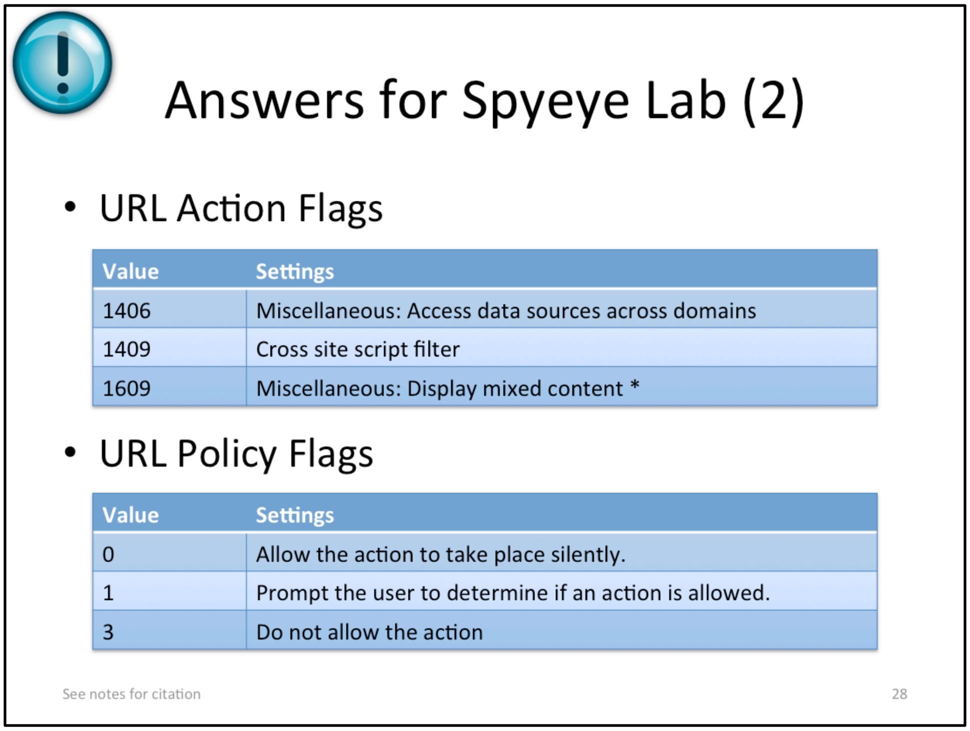
- Zones

Value	Setting
0	My Computer
1	Local Intranet Zone
2	Trusted sites Zone
3	Internet Zone
4	Restricted Sites Zone

See notes for citation

[References]

- MMPC Threat Report – EyeStye, http://www.microsoft.com/en-us/download/details.aspx?id=30399
- Internet Explorer security zones registry entries for advanced users, http://support.microsoft.com/kb/182569



Answers for Spyeye Lab (2)

- URL Action Flags

Value	Settings
1406	Miscellaneous: Access data sources across domains
1409	Cross site script filter
1609	Miscellaneous: Display mixed content *

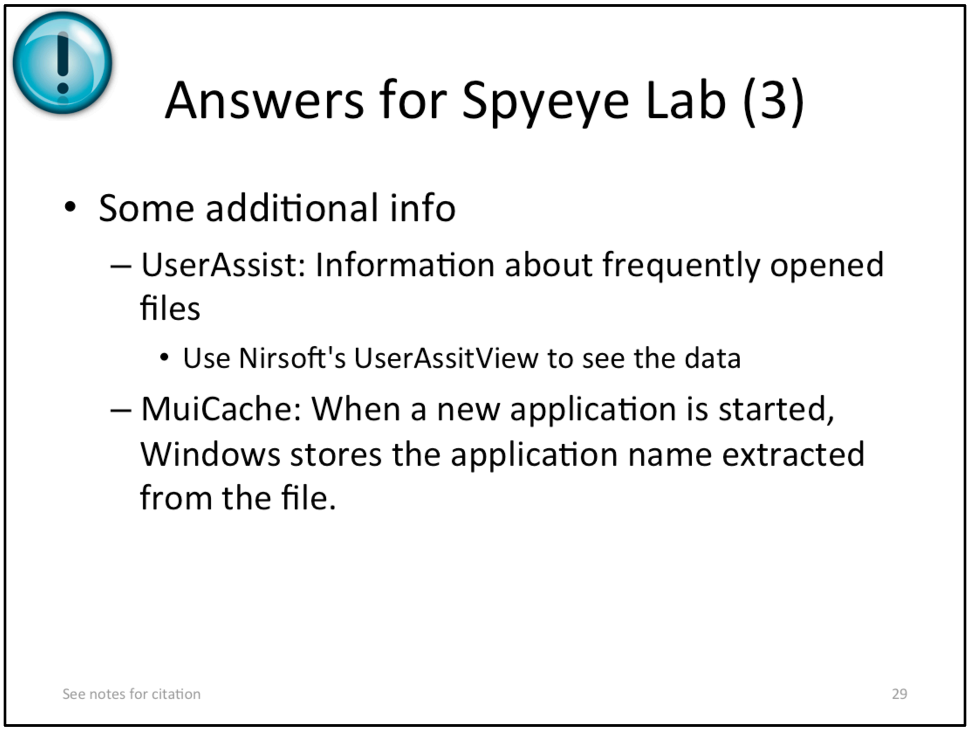
- URL Policy Flags

Value	Settings
0	Allow the action to take place silently.
1	Prompt the user to determine if an action is allowed.
3	Do not allow the action

See notes for citation

[References]

- URL Action Flags, http://msdn.microsoft.com/en-us/library/ms537178(v=vs.85).aspx
- URL Policy Flags, http://msdn.microsoft.com/en-us/library/ms537179(v=vs.85).aspx



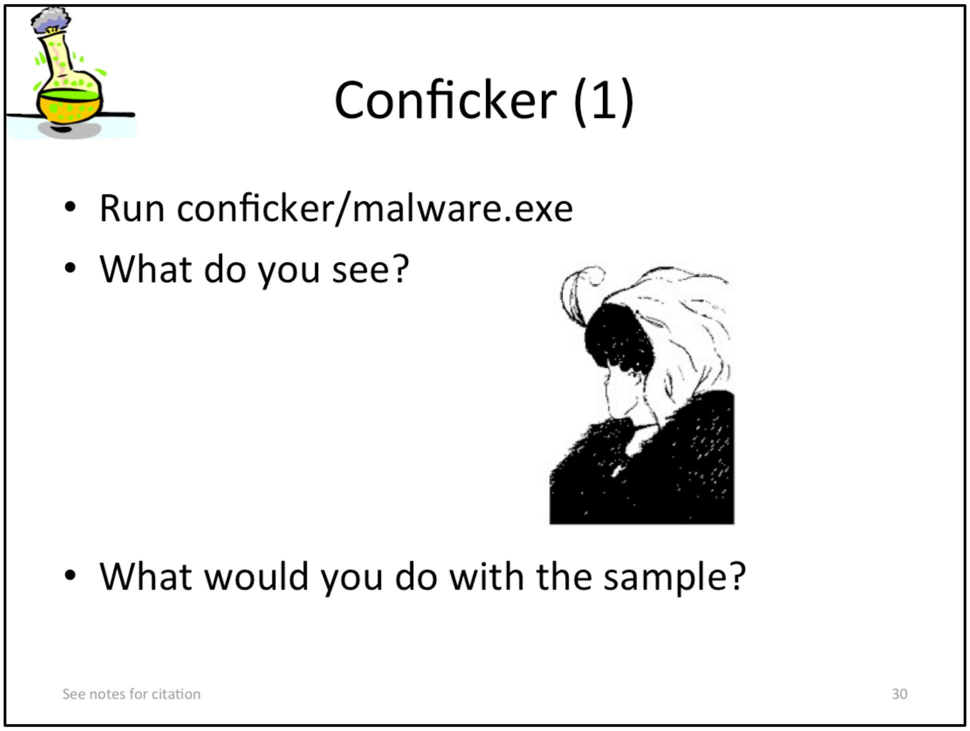
Answers for Spyeye Lab (3)

- Some additional info
  - UserAssist: Information about frequently opened files
    - Use Nirsoft's UserAssitView to see the data
  - MuiCache: When a new application is started, Windows stores the application name extracted from the file.

See notes for citation

[References]

- UserAssistView v1.02, http://www.nirsoft.net/utils/userassist_view.html
- MUICacheView v1.01, http://www.nirsoft.net/utils/muicache_view.html



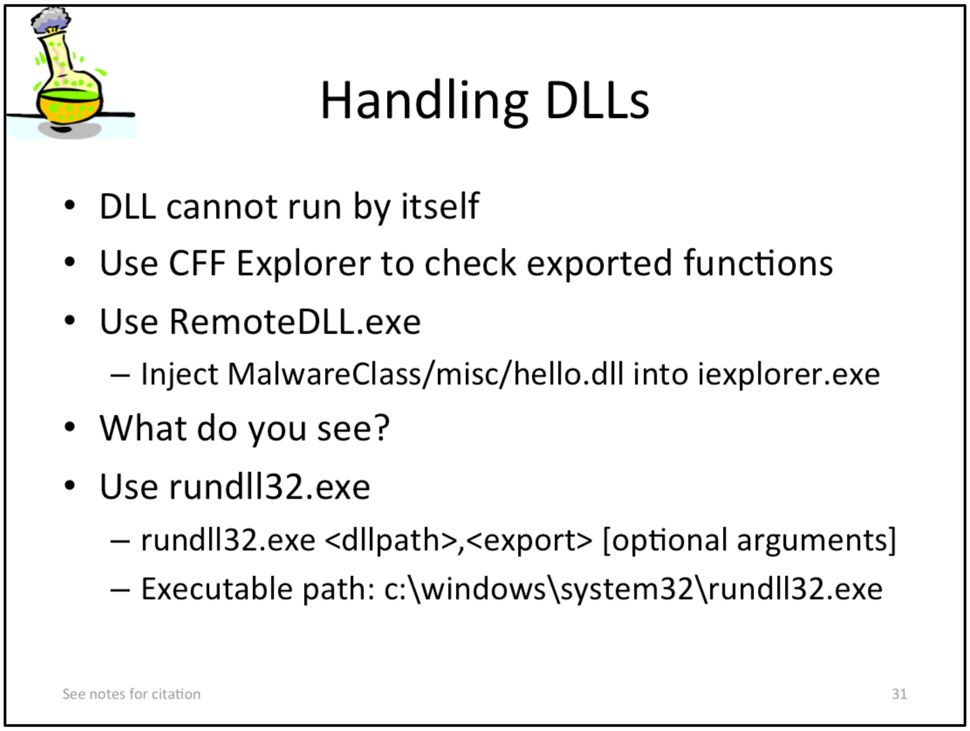
Conficker (1)

- Run conficker/malware.exe
- What do you see?
- What would you do with the sample?

See notes for citation

[Image Sources]

- http://mathworld.wolfram.com/images/gifs/young3.jpg



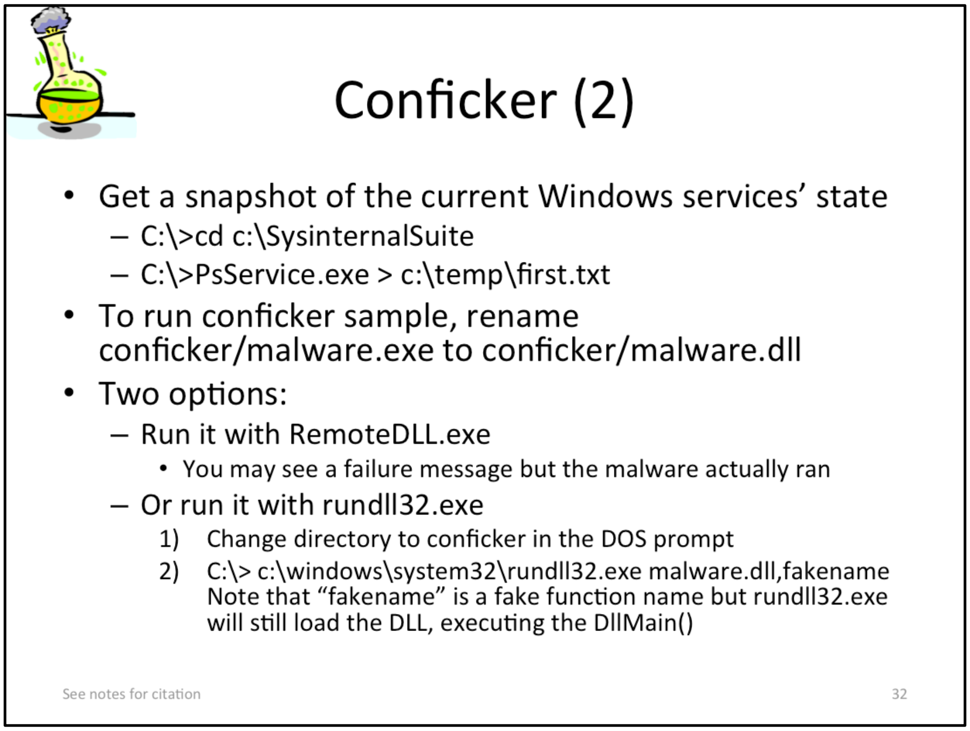
Handling DLLs

- DLL cannot run by itself
- Use CFF Explorer to check exported functions
- Use RemoteDLL.exe
  - Inject MalwareClass/misc/hello.dll into iexplorer.exe
- What do you see?
- Use rundll32.exe
  - rundll32.exe <dllpath>,<export> [optional arguments]
  - Executable path: c:\windows\system32\rundll32.exe

See notes for citation

[References]

- Michael Ligh et al., Chapter 13. Working with DDLs, Malware Analysts's Cookbook and DVD
- RemoteDLL, http://securityxploded.com/remotedll.php



Conficker (2)

- Get a snapshot of the current Windows services' state
  - C:\>cd c:\SysinternalSuite
  - C:\>PsService.exe > c:\temp\first.txt
- To run conficker sample, rename conficker/malware.exe to conficker/malware.dll
- Two options:
  - Run it with RemoteDLL.exe
    - You may see a failure message but the malware actually ran
  - Or run it with rundll32.exe
    1) Change directory to conficker in the DOS prompt
    2) C:\> c:\windows\system32\rundll32.exe malware.dll,fakename
       Note that "fakename" is a fake function name but rundll32.exe will still load the DLL, executing the DllMain()

See notes for citation

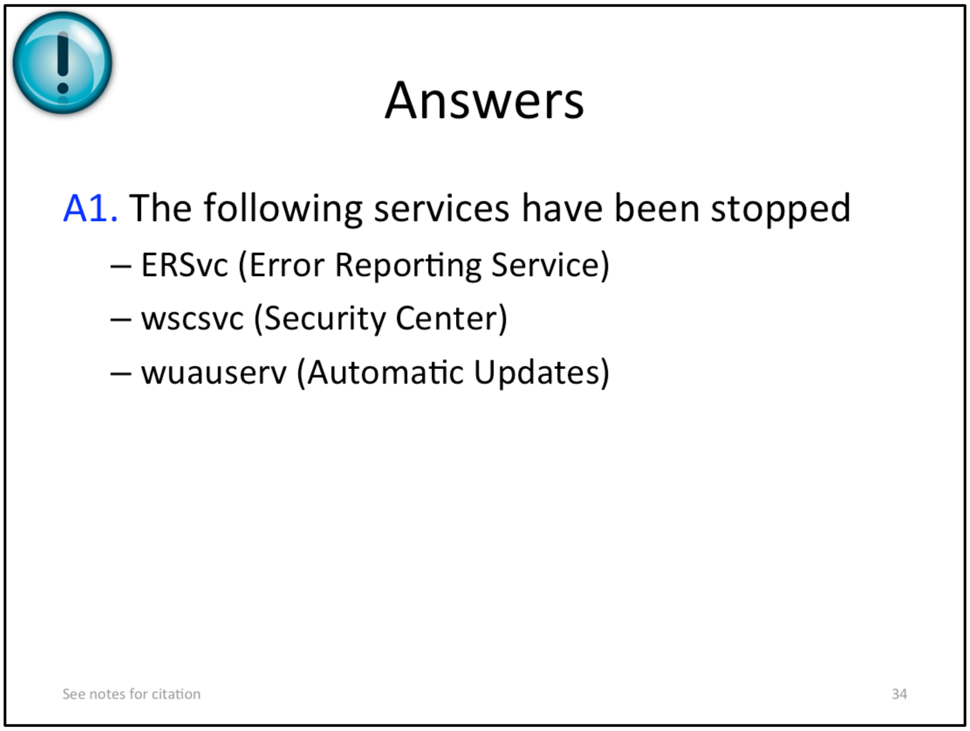
[References]

- Michael Ligh et al., Chapter 13. Working with DLLs, Malware Analyst's Cookbook and DVD

Conficker (3)

- Get the second snapshot of the current Windows services' state
  - C:\>PsServices.exe > c:\temp\second.txt
- Diff the two files
  - Use PSPad.exe (or any other GUI text editor)
    1. a. Open c:\temp\first.txt
    2. b. Tools → Text Differences → Text Diff with This Files... → select c:\temp\second.txt

Q1. How did conficker degrade security?



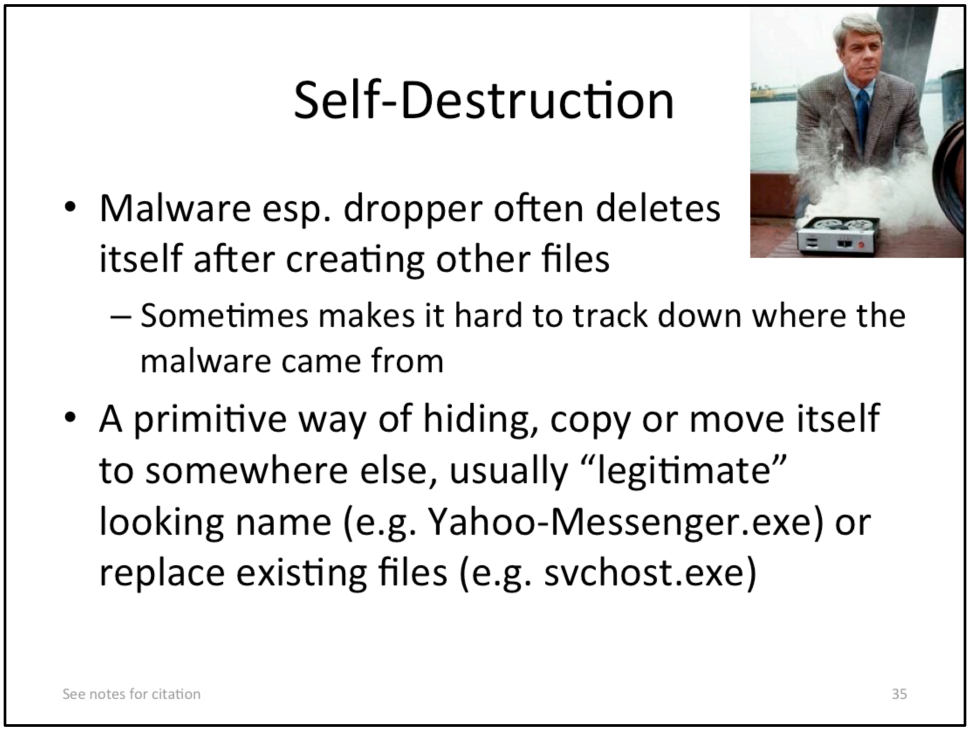
Answers

A1. The following services have been stopped

- ERSvc (Error Reporting Service)
- wscsvc (Security Center)
- wuauserv (Automatic Updates)

See notes for citation

34



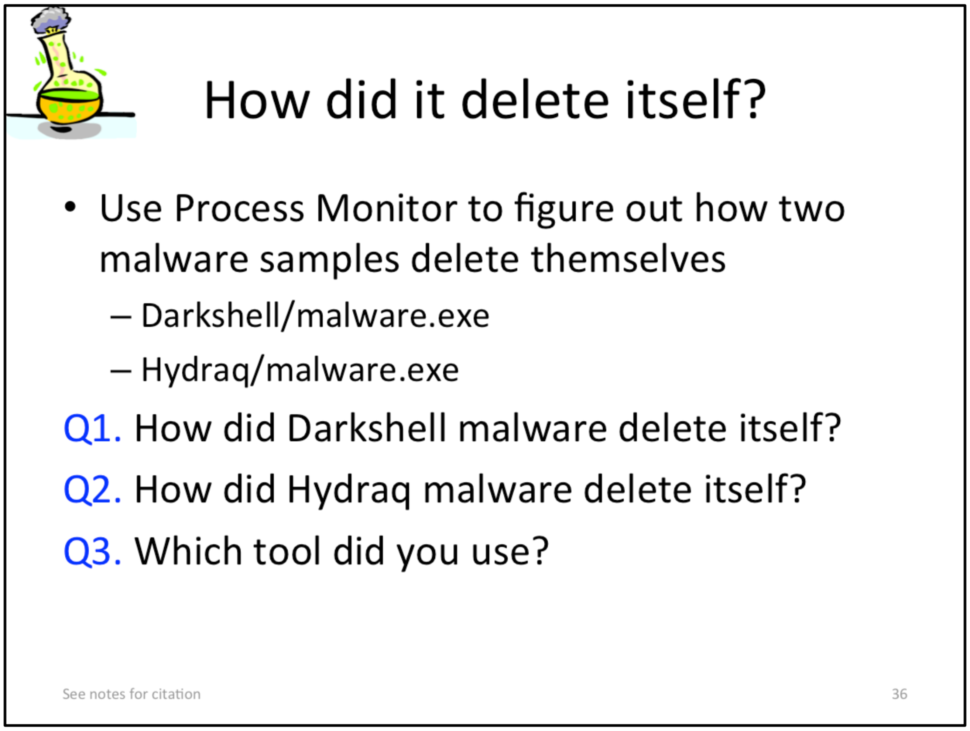
Self-Destruction

- Malware esp. dropper often deletes itself after creating other files
  - Sometimes makes it hard to track down where the malware came from
- A primitive way of hiding, copy or move itself to somewhere else, usually “legitimate” looking name (e.g. Yahoo-Messenger.exe) or replace existing files (e.g. svchost.exe)

See notes for citation

[Image Sources]

- http://www.techweekeurope.co.uk/wp-content/uploads/2012/05/phelpstape.jpg



How did it delete itself?

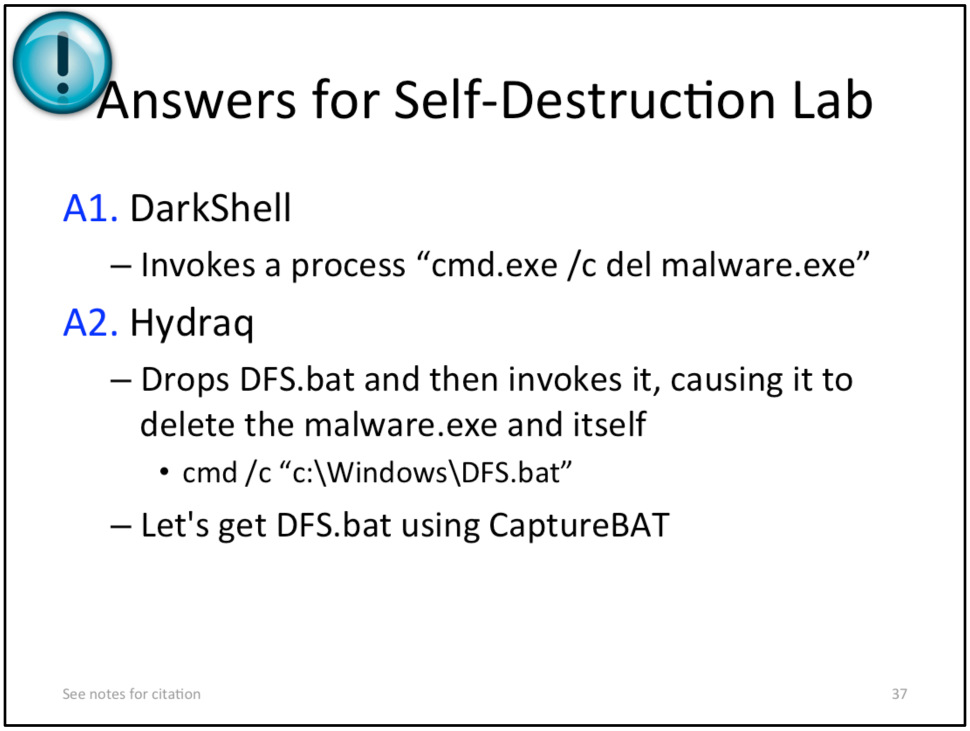
- Use Process Monitor to figure out how two malware samples delete themselves
  - Darkshell/malware.exe
  - Hydraq/malware.exe

Q1. How did Darkshell malware delete itself?

Q2. How did Hydraq malware delete itself?

Q3. Which tool did you use?

See notes for citation



Answers for Self-Destruction Lab

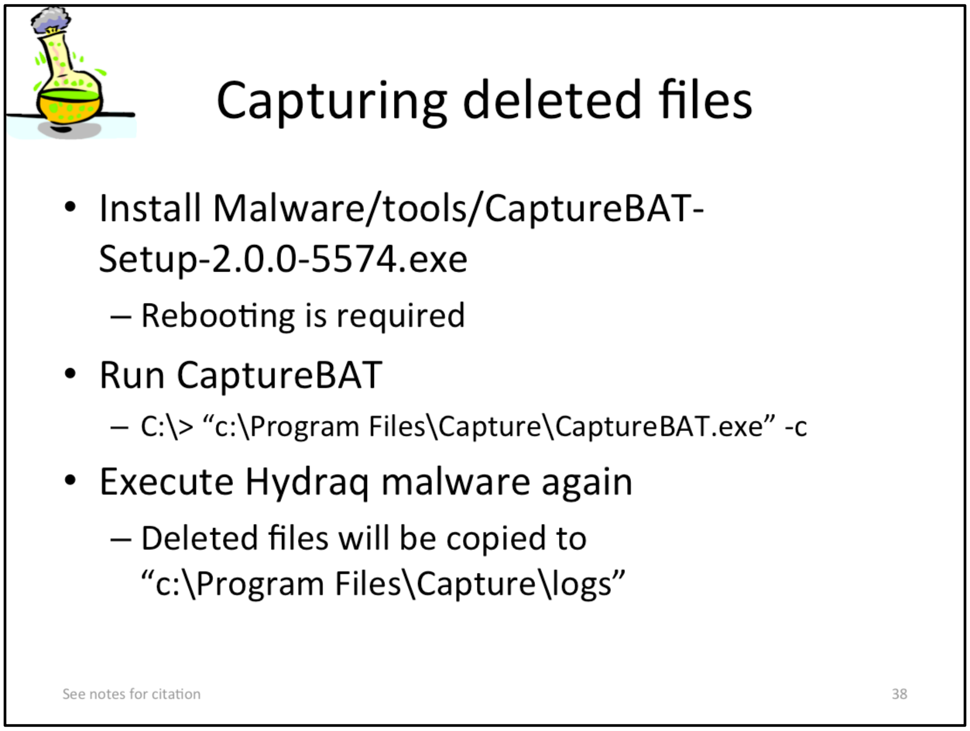
A1. DarkShell

- Invokes a process “cmd.exe /c del malware.exe”

A2. Hydraq

- Drops DFS.bat and then invokes it, causing it to delete the malware.exe and itself
  - cmd /c “c:\Windows\DFS.bat”
- Let's get DFS.bat using CaptureBAT

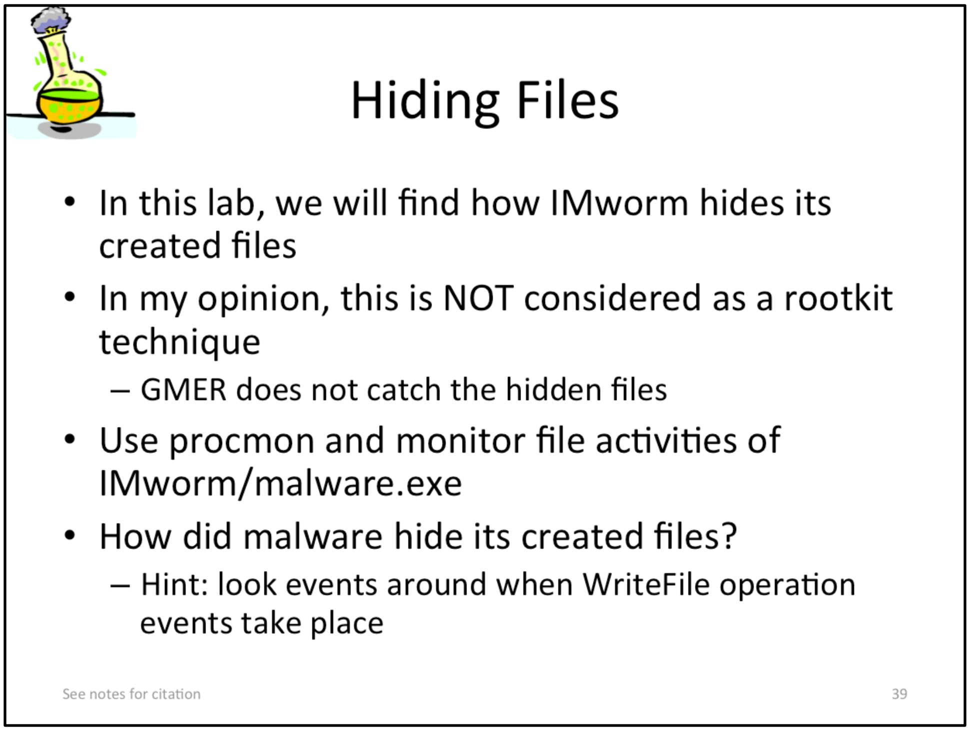
See notes for citation



Capturing deleted files

- Install Malware/tools/CaptureBAT-Setup-2.0.0-5574.exe
  - Rebooting is required
- Run CaptureBAT
  - C:\> “c:\Program Files\Capture\CaptureBAT.exe” -c
- Execute Hydraq malware again
  - Deleted files will be copied to “c:\Program Files\Capture\logs”

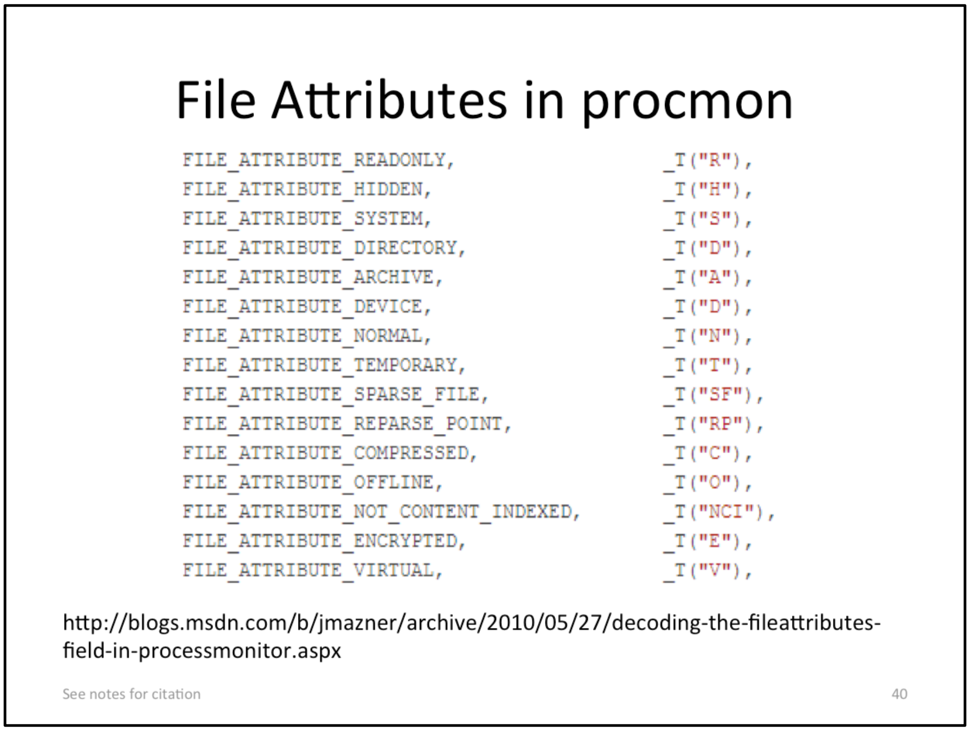
See notes for citation



Hiding Files

- In this lab, we will find how IMworm hides its created files
- In my opinion, this is NOT considered as a rootkit technique
  - GMER does not catch the hidden files
- Use procmon and monitor file activities of IMworm/malware.exe
- How did malware hide its created files?
  - Hint: look events around when WriteFile operation events take place

See notes for citation



File Attributes in procmon

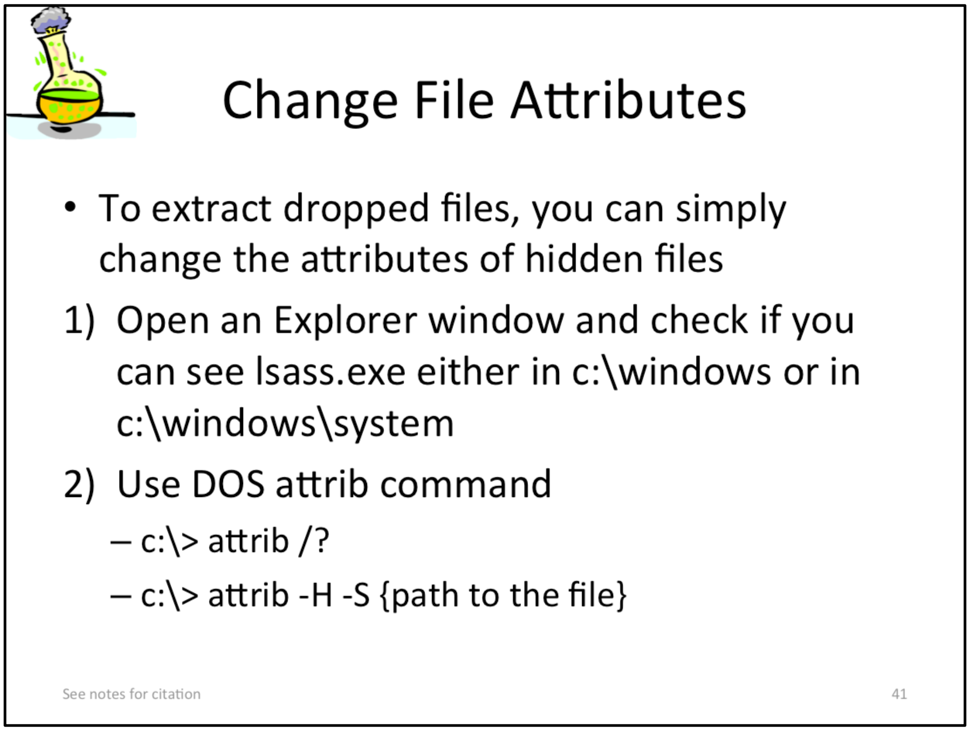
```
FILE_ATTRIBUTE_READONLY,                  _T("R"),
FILE_ATTRIBUTE_HIDDEN,                    _T("H"),
FILE_ATTRIBUTE_SYSTEM,                    _T("S"),
FILE_ATTRIBUTE_DIRECTORY,                 _T("D"),
FILE_ATTRIBUTE_ARCHIVE,                   _T("A"),
FILE_ATTRIBUTE_DEVICE,                    _T("D"),
FILE_ATTRIBUTE_NORMAL,                    _T("N"),
FILE_ATTRIBUTE_TEMPORARY,                 _T("T"),
FILE_ATTRIBUTE_SPARSE_FILE,               _T("SF"),
FILE_ATTRIBUTE_REPARSE_POINT,             _T("RP"),
FILE_ATTRIBUTE_COMPRESSED,                _T("C"),
FILE_ATTRIBUTE_OFFLINE,                   _T("O"),
FILE_ATTRIBUTE_NOT_CONTENT_INDEXED,       _T("NCI"),
FILE_ATTRIBUTE_ENCRYPTED,                 _T("E"),
FILE_ATTRIBUTE_VIRTUAL,                   _T("V"),
```

http://blogs.msdn.com/b/jmazner/archive/2010/05/27/decoding-the-fileattributes-field-in-processmonitor.aspx

See notes for citation

[References]

- Jeremy M, Decoding the FileAttributes field in ProcessMonitor, http://blogs.msdn.com/b/jmazner/archive/2010/05/27/decoding-the-fileattributes-field-in-processmonitor.aspx



Change File Attributes

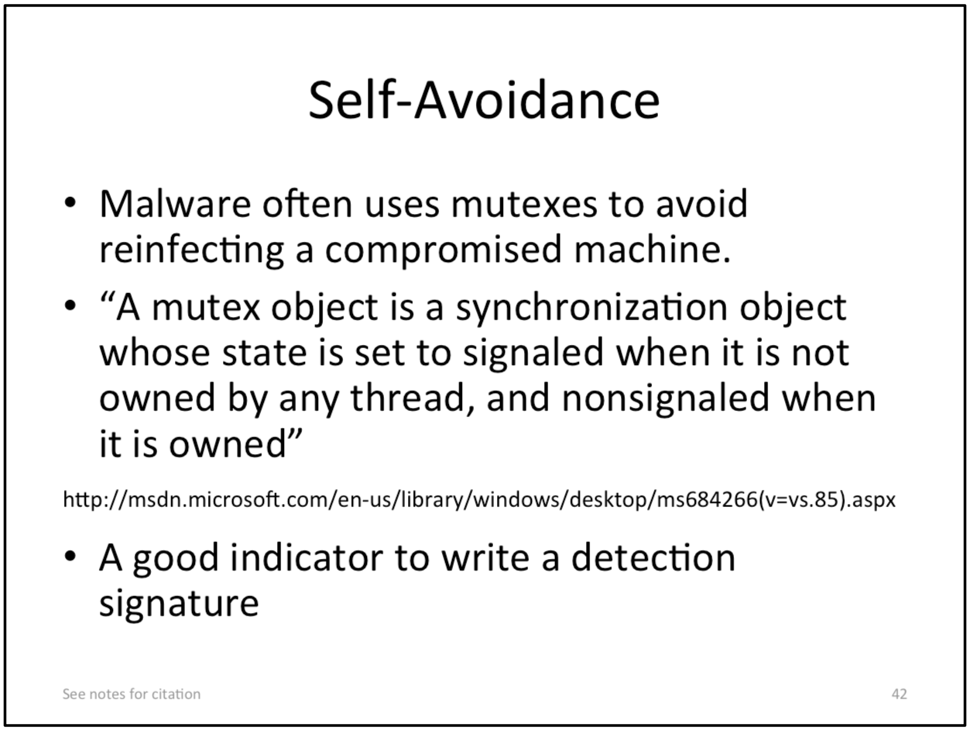
- To extract dropped files, you can simply change the attributes of hidden files

1) Open an Explorer window and check if you can see lsass.exe either in c:\windows or in c:\windows\system
2) Use DOS attrib command
   - c:\> attrib /?
   - c:\> attrib -H -S {path to the file}

See notes for citation 41

[References]

- Microsoft DOS attrib command, http://www.computerhope.com/attribhl.htm



Self-Avoidance

- Malware often uses mutexes to avoid reinfecting a compromised machine.
- “A mutex object is a synchronization object whose state is set to signaled when it is not owned by any thread, and nonsignaled when it is owned”

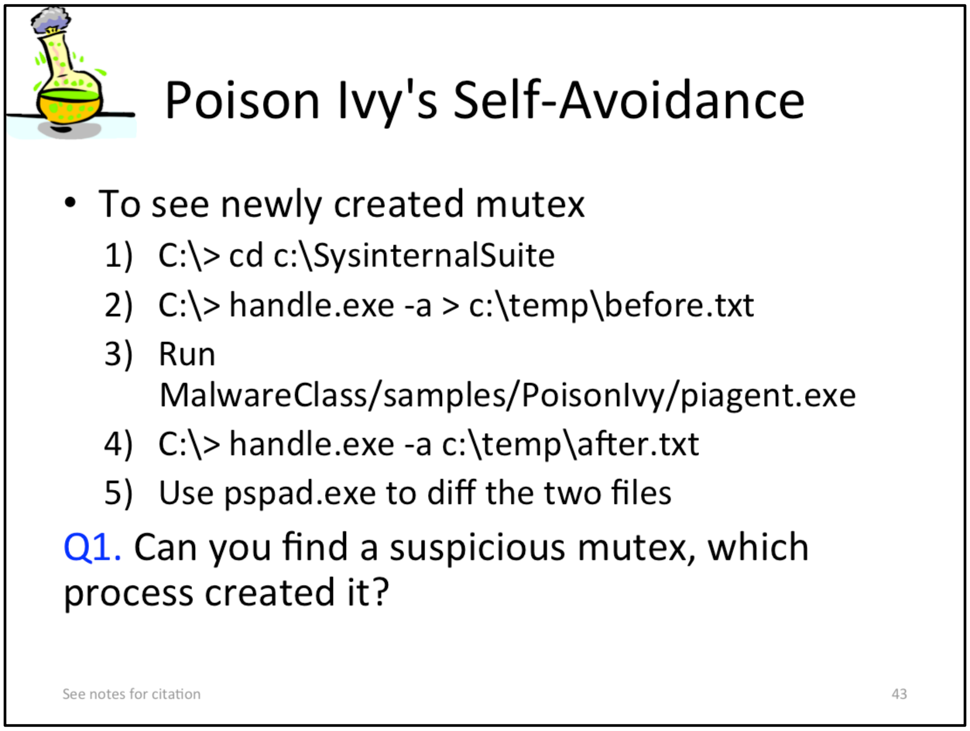
http://msdn.microsoft.com/en-us/library/windows/desktop/ms684266(v=vs.85).aspx

- A good indicator to write a detection signature

See notes for citation

[References]

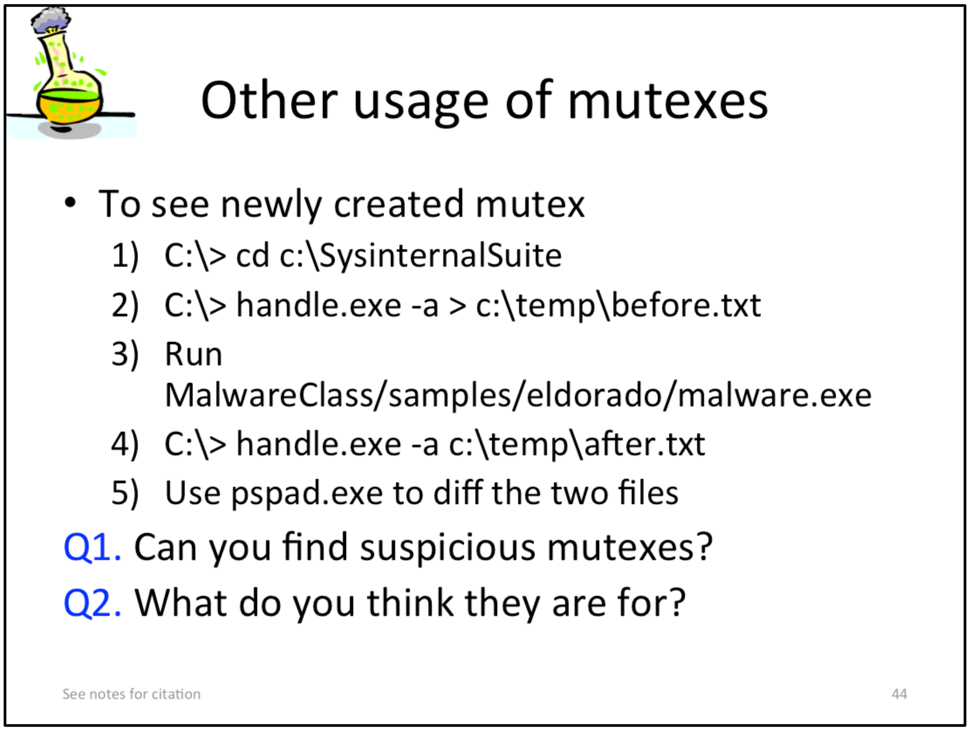
- Mutex Objects (Windows), http://msdn.microsoft.com/en-us/library/windows/desktop/ms684266(v=vs.85).aspx



Poison Ivy's Self-Avoidance

- To see newly created mutex
  1) C:\> cd c:\SysinternalSuite
  2) C:\> handle.exe -a > c:\temp\before.txt
  3) Run MalwareClass/samples/PoisonIvy/piagent.exe
  4) C:\> handle.exe -a c:\temp\after.txt
  5) Use pspad.exe to diff the two files

Q1. Can you find a suspicious mutex, which process created it?

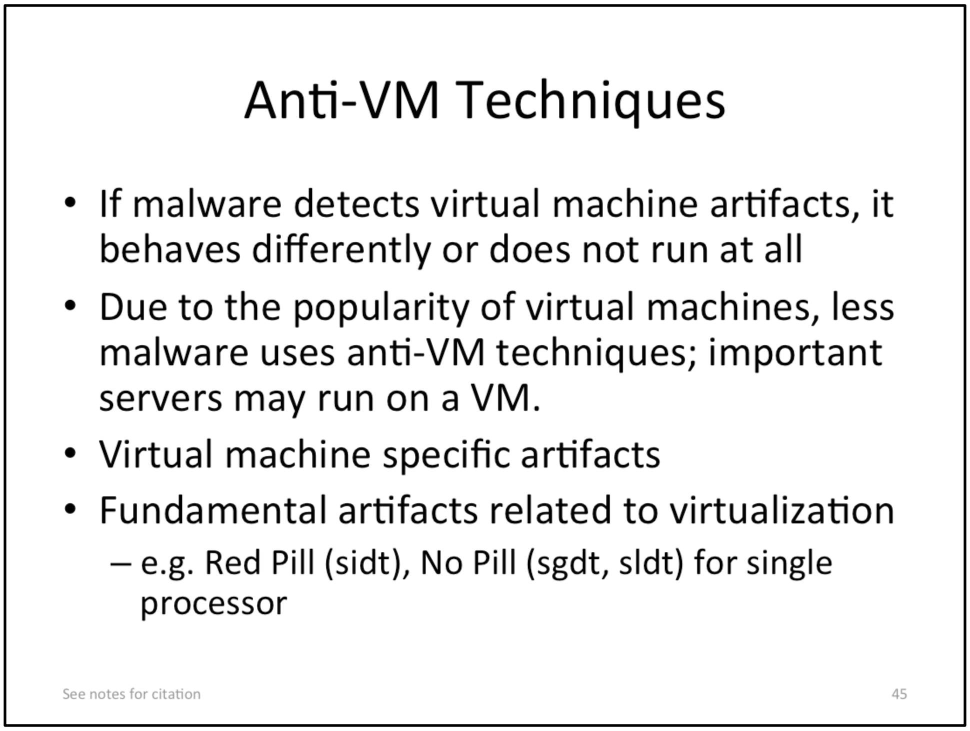


Other usage of mutexes

- To see newly created mutex
  1) C:\> cd c:\SysinternalSuite
  2) C:\> handle.exe -a > c:\temp\before.txt
  3) Run MalwareClass/samples/eldorado/malware.exe
  4) C:\> handle.exe -a c:\temp\after.txt
  5) Use pspad.exe to diff the two files

Q1. Can you find suspicious mutexes?

Q2. What do you think they are for?



Anti-VM Techniques

- If malware detects virtual machine artifacts, it behaves differently or does not run at all
- Due to the popularity of virtual machines, less malware uses anti-VM techniques; important servers may run on a VM.
- Virtual machine specific artifacts
- Fundamental artifacts related to virtualization
  - e.g. Red Pill (sidt), No Pill (sgdt, sldt) for single processor

See notes for citation

[References]

- Joanna Rutkowska, http://www.ouah.org/Red_%20Pill.html
- Danny Quist et al., http://www.offensivecomputing.net/files/active/0/vm.pdf
- Mikael, prowling - NSM foo, http://blog.prowling.nu/2012/08/modifying-virtualbox-settings-for.html