

Malware Analysis

Basic Analysis

By Z-Lab team

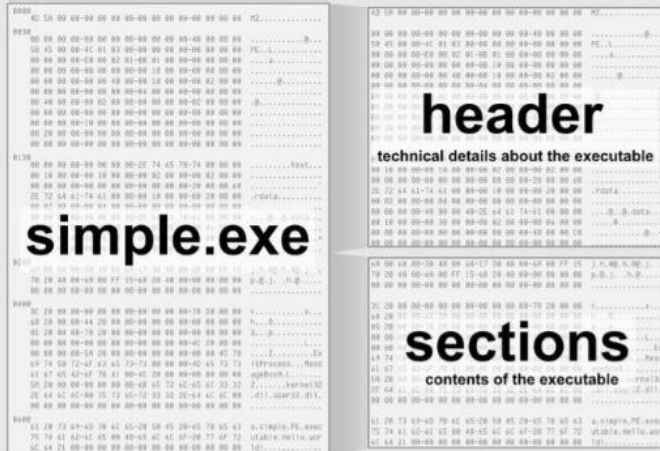


ISWATLab

PE file format

- The Portable Executable (PE) format is a file format for executables and DLLs used in 32-bit and 64-bit versions of Windows operating system.
- The term «portable» refers to the format's versatility in numerous environments of operating system software architecture.

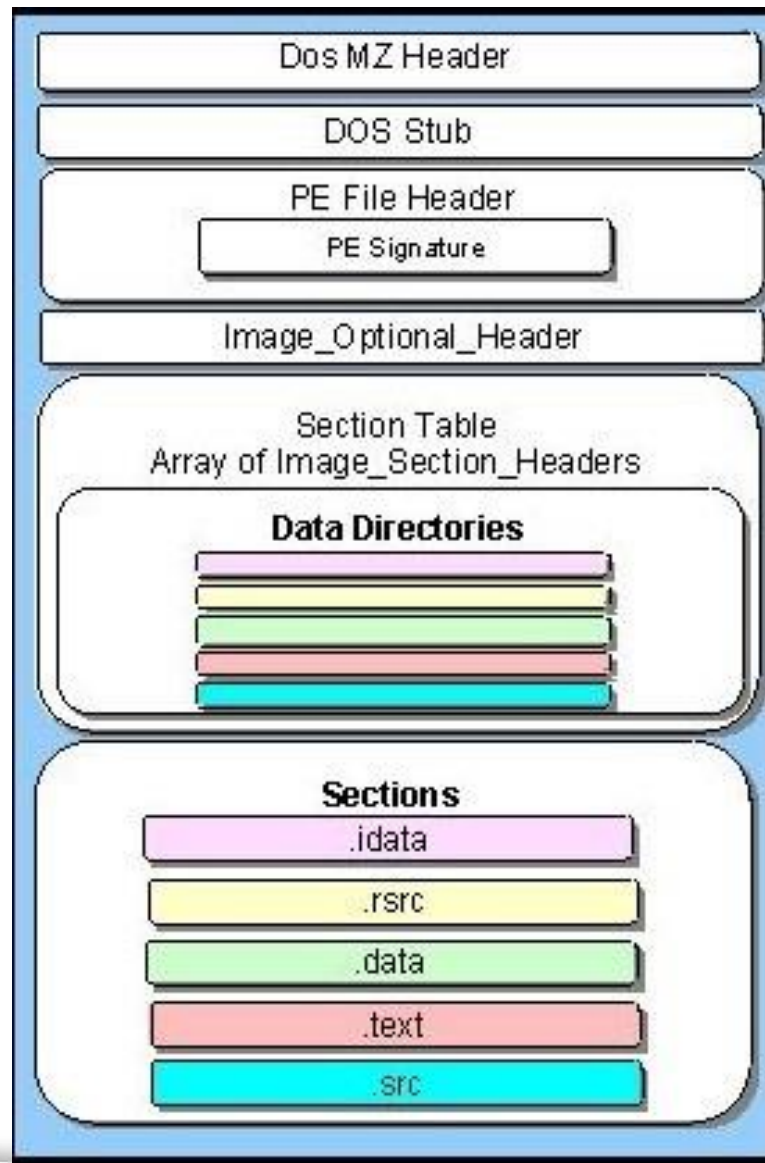
Dissected PE



```
61 20 73 69-8D 70 41      data          a.simple.PE.e
75 74 61 62-8C 65 61     ?         utable.Hello.i
6C 64 21 00-00 00 00     information used by the code  |d|
```

```
Offset: 0x00000000
61 20 73 69-60 78 6C 65-20 58 45 20-65 78 65 63 a.simple.PE.exec
75 74 61 62-6C 65 00 48-65 6C 6C 6F-20 77 6F 72 utable.Hello.world
6C 64 21 00
```

PE layout

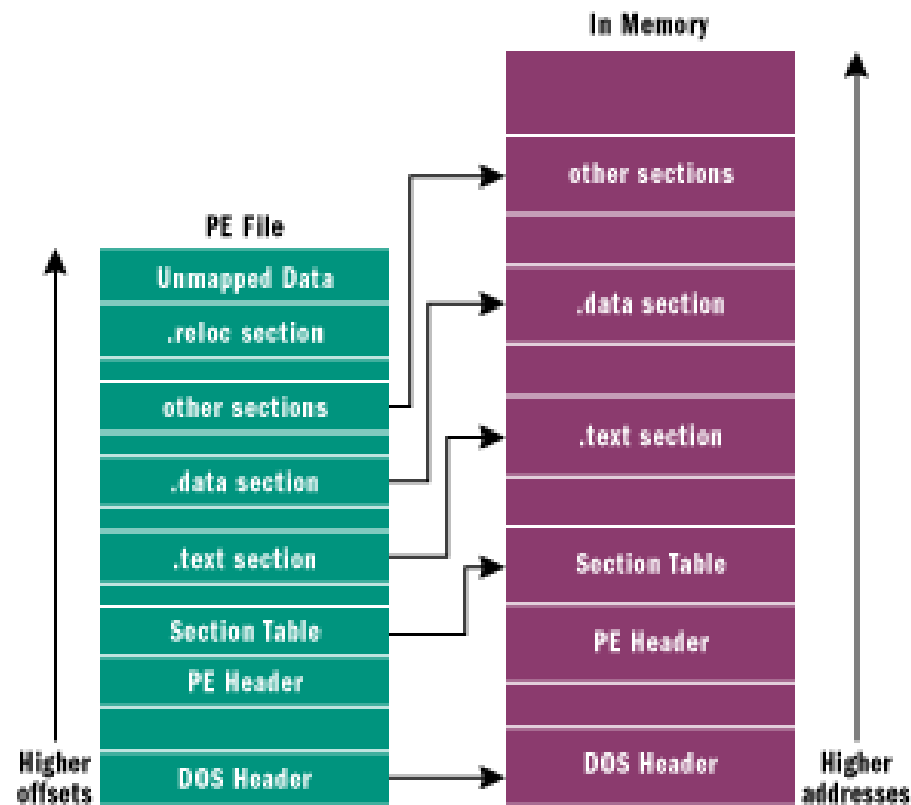


Some sections

- `.text`
 - Contains the executable code
- `.data`
 - Contains initialized data
- `.reloc`
 - Contains relocation information
- `.rsrc`
 - Contains resource info of a module
- `.idata`
 - Contains import data

Memory mapping

- Direct mapping in memory



DLL

- Dynamic-link Library

- Shared library between many processes
- It is a PE file with the IMAGE_FILE_DLL flag set
- It exports some functions

- Linking a DLL:

- Dynamic Linking: the OS loads the DLLs in memory using IAT
- Runtime Linking: when needs the DLL, the process uses

```
dllHandle = LoadLibrary ( filename );  
           |  
           v  
funcAddress = GetProcAddress ( dllHandle, functionName);  
           |  
           v  
call funcAddress;
```


General Rules for Malware Analysis

- Don't Get Caught in Details
 - You don't need to understand 100% of the code
 - Focus on key features
- Try Several Tools
 - If one tool fails, try another
 - Don't get stuck on a hard issue, move along
- Malware authors are constantly raising the bar

Basic Analysis

- Basic static analysis
 - View malware without looking at instructions
 - Tools: VirusTotal, strings
 - Quick and easy but fails for advanced malware and can miss important behavior
- Basic dynamic analysis
 - Easy but requires a safe test environment
 - Not effective on all malware

Only a First Step

- Malware can easily change its signature and fool the antivirus
- VirusTotal is convenient, but using it may alert attackers that they've been caught



VirusTotal is a free service that **analyzes suspicious files and URLs** and facilitates the quick detection of viruses, worms, trojans, and all kinds of malware.

No file selected

Choose File

Maximum file size: 64MB

Hashes

- MD5, SHA-1, SHA-256, SHA 512
- Condenses a file of any size down to a fixed-length fingerprint
- Uniquely identifies a file well in practice
 - There are MD5 collisions but they are not common
 - Collision: two different files with the same hash
- Label a malware file
- Share the hash with other analysts to identify malware
- Search the hash online to see if someone else has already identified the file

HashCalc

The screenshot shows the HashCalc application window. The title bar is light blue with a standard Windows icon and the text 'HashCalc'. The main area has a light gray background. At the top, there are two sections: 'Data Format:' with a dropdown menu set to 'File', and 'Data:' with a text box containing 'C:\Users\student\Desktop\p3.pcap' and a browse button (...). Below these are 'Key Format:' with a dropdown set to 'Text string' and an empty 'Key:' text box. A horizontal line separates the input section from the output section. The output section lists four hash types: MD5, MD4, SHA1, and SHA256. MD5 and SHA1 are checked, and their corresponding hash values are displayed in text boxes to the right. MD4 and SHA256 are unchecked, and their text boxes are empty.

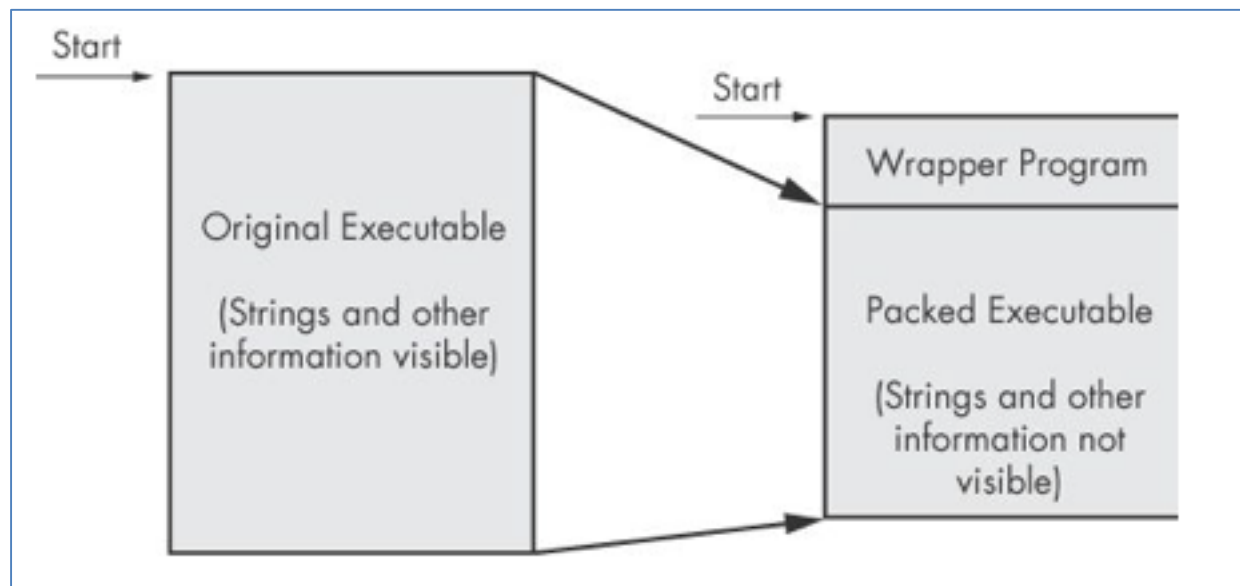
Hash Type	Hash Value
<input checked="" type="checkbox"/> MD5	52583b5e2c99d19c046915181fd7b29b
<input type="checkbox"/> MD4	
<input checked="" type="checkbox"/> SHA1	991d4e880832dd6aaebadb8040798a6b9f163194
<input type="checkbox"/> SHA256	

Strings

- Any sequence of printable characters is a **string**
- Strings are terminated by a **null** (0x00)
- ASCII characters are 8 bits long
 - Now called ANSI
- Unicode characters are 16 bits long
 - Microsoft calls them "wide characters"

Packing Files

- The code is compressed, like a Zip file
- This makes the strings and instructions unreadable
- All you'll see is the **wrapper** – small code that unpacks the file when it is run



Detecting Packers with PEiD

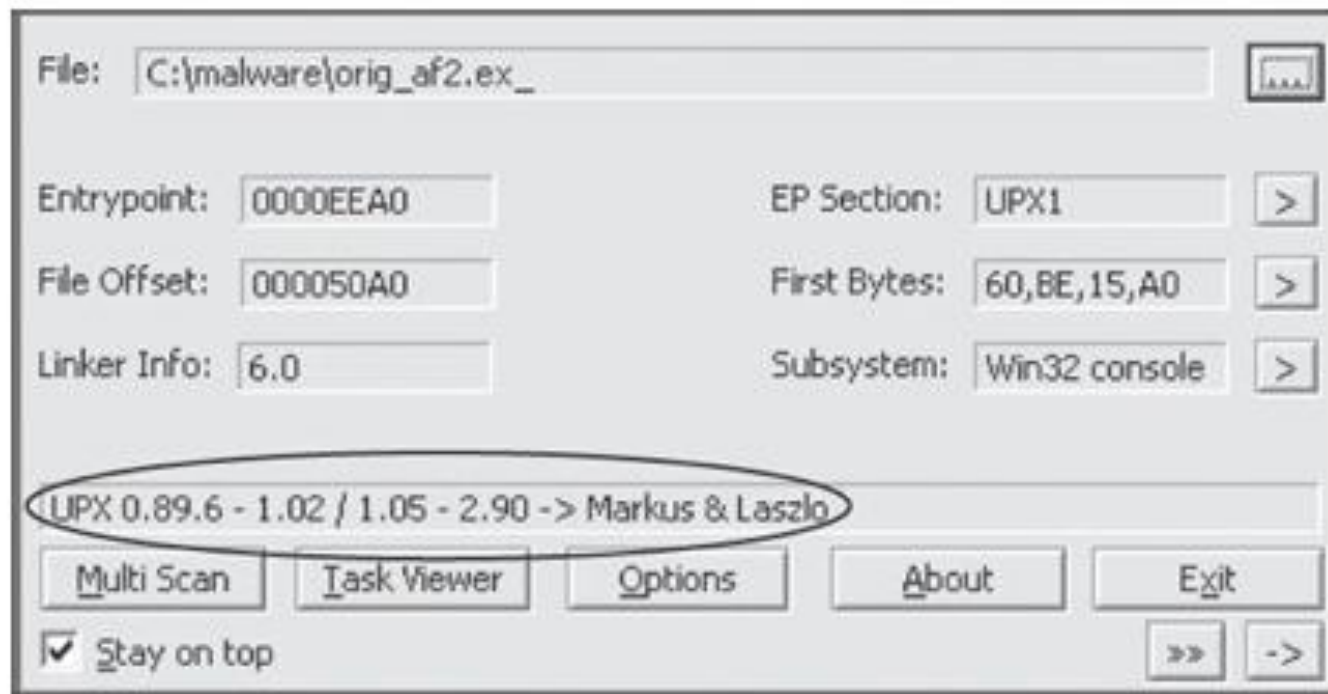
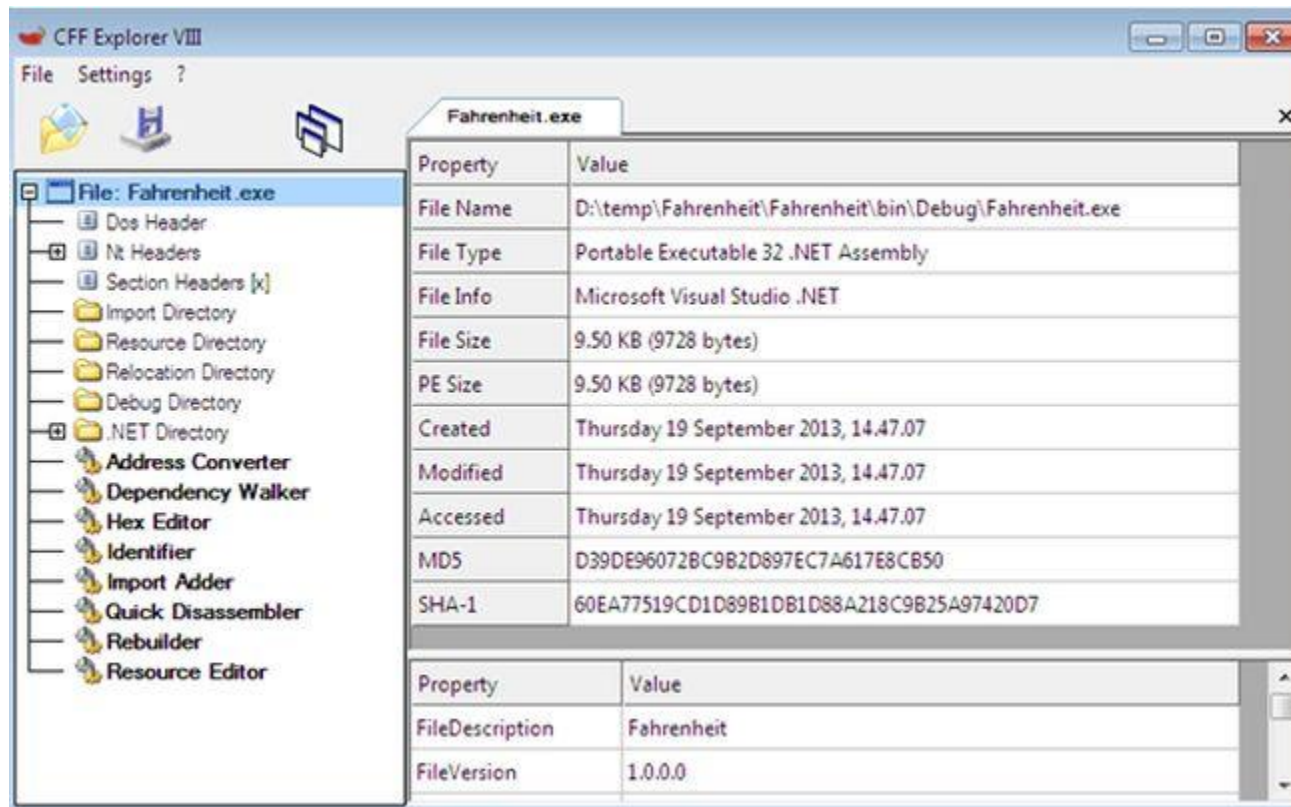


Figure 2-5. The PEiD program

CFF Explorer



Dynamic Analysis

- Static analysis can reach a dead-end, due to
 - Obfuscation
 - Packing
 - Examiner has exhausted the available static analysis techniques
- Dynamic analysis is efficient and will show you exactly what the malware does

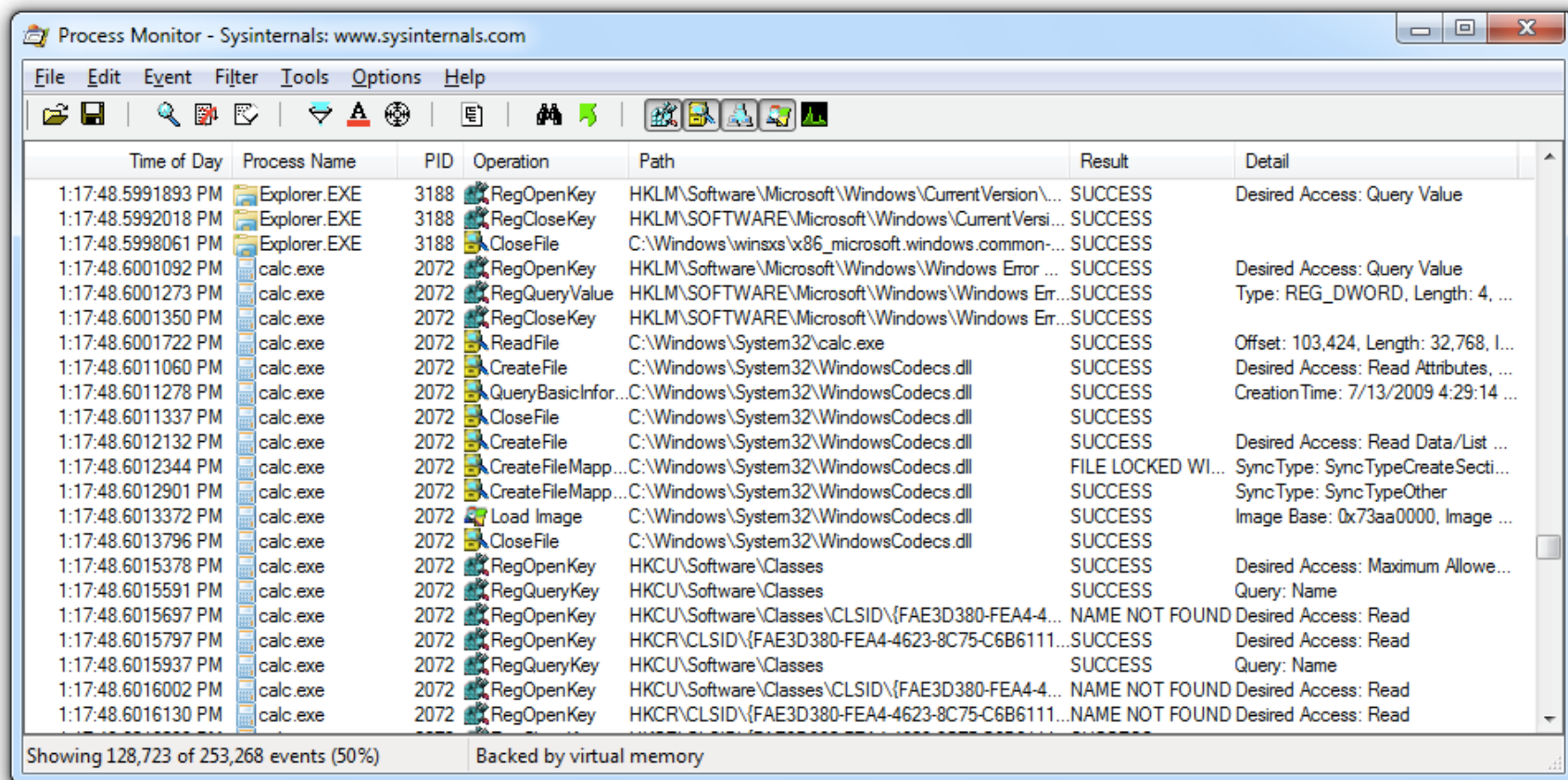
Sandbox

- All-in-one software for basic dynamic analysis
- Virtualized environment that simulates network services
- Examples: Joe Sandbox, ThreatExpert, BitBlaze, Cuckoo Sandbox, Hybrid Analysis
- They produce a report of results

Process Monitor

- Monitors registry, file system, network, process, and thread activity
- All recorded events are kept, but you can filter the display to make it easier to find items of interest
- Don't run it too long or it will fill up all RAM and crash the machine

Procmon



Time of Day	Process Name	PID	Operation	Path	Result	Detail
1:17:48.5991893 PM	Explorer.EXE	3188	RegOpenKey	HKLM\Software\Microsoft\Windows\CurrentVersion\...	SUCCESS	Desired Access: Query Value
1:17:48.5992018 PM	Explorer.EXE	3188	RegCloseKey	HKLM\SOFTWARE\Microsoft\Windows\CurrentVersi...	SUCCESS	
1:17:48.5998061 PM	Explorer.EXE	3188	CloseFile	C:\Windows\winsxs\x86_microsoft.windows.common-...	SUCCESS	
1:17:48.6001092 PM	calc.exe	2072	RegOpenKey	HKLM\Software\Microsoft\Windows\Windows Error ...	SUCCESS	Desired Access: Query Value
1:17:48.6001273 PM	calc.exe	2072	RegQueryValue	HKLM\SOFTWARE\Microsoft\Windows\Windows Err...	SUCCESS	Type: REG_DWORD, Length: 4, ...
1:17:48.6001350 PM	calc.exe	2072	RegCloseKey	HKLM\SOFTWARE\Microsoft\Windows\Windows Err...	SUCCESS	
1:17:48.6001722 PM	calc.exe	2072	ReadFile	C:\Windows\System32\calc.exe	SUCCESS	Offset: 103,424, Length: 32,768, l...
1:17:48.6011060 PM	calc.exe	2072	CreateFile	C:\Windows\System32\WindowsCodecs.dll	SUCCESS	Desired Access: Read Attributes, ...
1:17:48.6011278 PM	calc.exe	2072	QueryBasicInfor...	C:\Windows\System32\WindowsCodecs.dll	SUCCESS	CreationTime: 7/13/2009 4:29:14 ...
1:17:48.6011337 PM	calc.exe	2072	CloseFile	C:\Windows\System32\WindowsCodecs.dll	SUCCESS	
1:17:48.6012132 PM	calc.exe	2072	CreateFile	C:\Windows\System32\WindowsCodecs.dll	SUCCESS	Desired Access: Read Data/List ...
1:17:48.6012344 PM	calc.exe	2072	CreateFileMapp...	C:\Windows\System32\WindowsCodecs.dll	FILE LOCKED WI...	SyncType: SyncTypeCreateSecti...
1:17:48.6012901 PM	calc.exe	2072	CreateFileMapp...	C:\Windows\System32\WindowsCodecs.dll	SUCCESS	SyncType: SyncTypeOther
1:17:48.6013372 PM	calc.exe	2072	Load Image	C:\Windows\System32\WindowsCodecs.dll	SUCCESS	Image Base: 0x73aa0000, Image ...
1:17:48.6013796 PM	calc.exe	2072	CloseFile	C:\Windows\System32\WindowsCodecs.dll	SUCCESS	
1:17:48.6015378 PM	calc.exe	2072	RegOpenKey	HKCU\Software\Classes	SUCCESS	Desired Access: Maximum Allowe...
1:17:48.6015591 PM	calc.exe	2072	RegQueryValue	HKCU\Software\Classes	SUCCESS	Query: Name
1:17:48.6015697 PM	calc.exe	2072	RegOpenKey	HKCU\Software\Classes\CLSID\{FAE3D380-FEA4-4...	NAME NOT FOUND	Desired Access: Read
1:17:48.6015797 PM	calc.exe	2072	RegOpenKey	HKCR\CLSID\{FAE3D380-FEA4-4623-8C75-C6B6111...	SUCCESS	Desired Access: Read
1:17:48.6015937 PM	calc.exe	2072	RegQueryKey	HKCU\Software\Classes	SUCCESS	Query: Name
1:17:48.6016002 PM	calc.exe	2072	RegOpenKey	HKCU\Software\Classes\CLSID\{FAE3D380-FEA4-4...	NAME NOT FOUND	Desired Access: Read
1:17:48.6016130 PM	calc.exe	2072	RegOpenKey	HKCR\CLSID\{FAE3D380-FEA4-4623-8C75-C6B6111...	NAME NOT FOUND	Desired Access: Read

Showing 128,723 of 253,268 events (50%) Backed by virtual memory

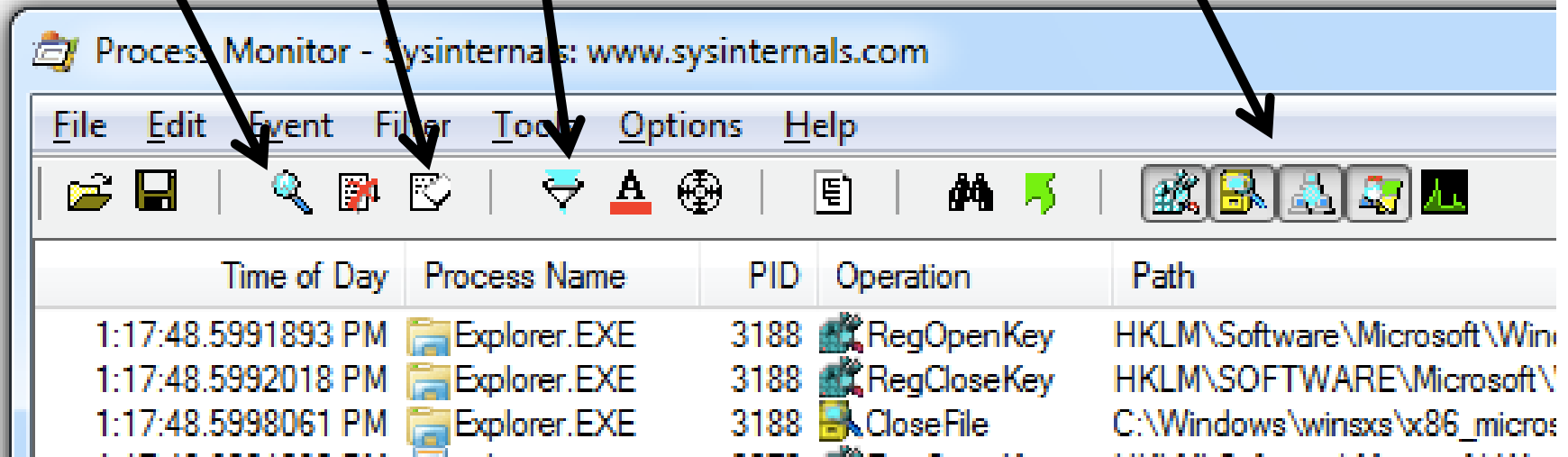
Process Monitor Toolbar

Start/Stop
Capture

Erase

Filter

Default Filters
Registry, File system, Network,
Processes



Process Explorer - Sysinternals: www.sysinternals.com [W7\student]

File Options View Process Find Users Help

Process	PID	CPU	Private Bytes	Working Set	Description	Company Name
System Idle Process	0	96.81	0 K	24 K		
System	4	0.09	48 K	560 K		
Interrupts	n/a	0.88	0 K	0 K	Hardware Interrupts and DPCs	
smss.exe	260		224 K	748 K	Windows Session Manager	Microsoft Corporation
csrss.exe	348	< 0.01	1,252 K	3,164 K	Client Server Runtime Process	Microsoft Corporation
wininit.exe	400		892 K	3,084 K	Windows Start-Up Application	Microsoft Corporation
services.exe	504	0.01	3,972 K	6,640 K	Services and Controller app	Microsoft Corporation
svchost.exe	652		2,700 K	6,024 K	Host Process for Windows S...	Microsoft Corporation
dllhost.exe	1716		6,176 K	4,804 K	COM Surrogate	Microsoft Corporation
WmiPrvSE.exe	740		1,804 K	4,736 K	WMI Provider Host	Microsoft Corporation
svchost.exe	724	< 0.01	2,972 K	6,012 K	Host Process for Windows S...	Microsoft Corporation
svchost.exe	772		13,776 K	11,760 K	Host Process for Windows S...	Microsoft Corporation
audiodg.exe	3200		14,960 K	13,972 K	Windows Audio Device Grap...	Microsoft Corporation
svchost.exe	912		37,940 K	42,292 K	Host Process for Windows S...	Microsoft Corporation
dwm.exe	3248	0.74	61,892 K	27,976 K	Desktop Window Manager	Microsoft Corporation
svchost.exe	936	0.02	20,836 K	29,900 K	Host Process for Windows S...	Microsoft Corporation
svchost.exe	1116	0.03	5,136 K	8,340 K	Host Process for Windows S...	Microsoft Corporation
svchost.exe	1260	0.06	10,840 K	11,960 K	Host Process for Windows S...	Microsoft Corporation
spoolsv.exe	1352		5,392 K	7,436 K	Spooler SubSystem App	Microsoft Corporation
svchost.exe	1388		6,752 K	8,720 K	Host Process for Windows S...	Microsoft Corporation
svchost.exe	1500		2,472 K	4,712 K	Host Process for Windows S...	Microsoft Corporation
gogoc.exe	1592	< 0.01	1,216 K	3,920 K	gogoCLIENT	gogo6, Inc.
vmtoolsd.exe	1728	0.07	7,260 K	10,368 K	VMware Tools Core Service	VMware, Inc.
svchost.exe						

CPU Usage: 3.19% Commit Charge: 21.92% Processes: 57 Physical Usage: 30.24%

Wireshark

The image shows the Wireshark 1.6.5 interface. The top menu bar includes File, Edit, View, Go, Capture, Analyze, Statistics, Telephony, Tools, Internals, and Help. Below the menu is a toolbar with various icons. A filter bar contains a text input field and buttons for Expression..., Clear, and Apply. The main packet list pane displays a table of captured packets. The selected packet (No. 10684) is highlighted in blue. To the right of the packet list is the packet details pane, which shows the hierarchical structure of the selected packet. At the bottom is the packet bytes pane, which displays the raw data of the selected packet in hexadecimal and ASCII. The status bar at the very bottom shows the file path, the number of packets displayed, marked, and dropped, and the current profile.

No.	Time	Source	Destination	Protocol	Length	Info
10684	11:05:51.330276	192.168.10.126	192.168.10.130	UDP	210	Source port: 4723 Destination port: 60729
10685	11:05:51.338450	192.168.1.33	192.168.1.4	TCP	196	xgrid > 55061 [PSH, ACK] Seq=331003 Ack=1 win=8706 Len=142
10686	11:05:51.338644	192.168.1.4	192.168.1.33	TCP	54	55061 > xgrid [ACK] Seq=1 Ack=331145 win=16070 Len=0
10687	11:05:51.338874	192.168.1.33	192.168.1.4	TCP	196	xgrid > 55061 [PSH, ACK] Seq=331145 Ack=1 win=8706 Len=142
10688	11:05:51.339258	192.168.1.33	224.0.1.0	UDP	184	Source port: 41475 Destination port: 4770
10689	11:05:51.346775	192.168.10.126	192.168.10.130	UDP	210	Source port: 4723 Destination port: 60729
10690	11:05:51.353886	192.168.1.33	224.0.1.0	UDP	184	Source port: 41475 Destination port: 4770
10691	11:05:51.357375	192.168.1.33	192.168.1.4	TCP	196	xgrid > 55061 [PSH, ACK] Seq=331287 Ack=1 win=8706 Len=142
10692	11:05:51.357573	192.168.1.4	192.168.1.33	TCP	54	55061 > xgrid [ACK] Seq=1 Ack=331429 win=16425 Len=0
10693	11:05:51.361889	192.168.1.34	224.0.1.0	UDP	148	Source port: 61227 Destination port: 10126
10694	11:05:51.362733	192.168.1.35	224.0.1.0	UDP	390	Source port: 60632 Destination port: 10127
10695	11:05:51.363542	192.168.10.126	192.168.10.130	UDP	210	Source port: 4723 Destination port: 60729
10696	11:05:51.369624	192.168.1.33	224.0.1.0	UDP	184	Source port: 41475 Destination port: 4770
10697	11:05:51.380278	192.168.10.126	192.168.10.130	UDP	210	Source port: 4723 Destination port: 60729
10698	11:05:51.382020	192.168.1.33	192.168.1.4	TCP	196	xgrid > 55061 [PSH, ACK] Seq=331429 Ack=1 win=8706 Len=142
10699	11:05:51.386170	192.168.1.33	224.0.1.0	UDP	184	Source port: 41475 Destination port: 4770
10700	11:05:51.395149	192.168.1.34	224.0.1.0	UDP	148	Source port: 61227 Destination port: 10126
10701	11:05:51.396154	192.168.1.35	224.0.1.0	UDP	390	Source port: 60632 Destination port: 10127
10702	11:05:51.396898	192.168.10.126	192.168.10.130	UDP	210	Source port: 4723 Destination port: 60729
10703	11:05:51.402645	192.168.1.33	224.0.1.0	UDP	184	Source port: 41475 Destination port: 4770
10704	11:05:51.406757	192.168.1.33	192.168.1.4	TCP	196	xgrid > 55061 [PSH, ACK] Seq=331571 Ack=1 win=8706 Len=142
10705	11:05:51.407067	192.168.1.4	192.168.1.33	TCP	54	55061 > xgrid [ACK] Seq=1 Ack=331713 win=16354 Len=0
10706	11:05:51.407257	192.168.1.33	192.168.1.4	TCP	196	xgrid > 55061 [PSH, ACK] Seq=331713 Ack=1 win=8706 Len=142
10707	11:05:51.413467	192.168.10.126	192.168.10.130	UDP	210	Source port: 4723 Destination port: 60729

Frame 1: 210 bytes on wire (1680 bits), 210 bytes captured (1680 bits)

Ethernet II, Src: Ge_41:3d:4b (00:a0:f4:41:3d:4b), Dst: Schweitz_02:b7:33 (00:30:a7:02:b7:33)

Internet Protocol Version 4, Src: 192.168.10.126 (192.168.10.126), Dst: 192.168.10.130 (192.168.10.130)

User Datagram Protocol, Src Port: 4723 (4723), Dst Port: 60729 (60729)

Data (168 bytes)

0000 00 30 a7 02 b7 33 00 a0 f4 41 3d 4b 08 00 45 00 .0...3... .A=K..E.
0010 00 c4 33 d1 00 00 1e 11 d2 07 c0 a8 0a 7e c0 a8 ..3.....
0020 0a 82 12 73 ed 39 00 b0 7f d4 aa 01 00 a8 00 01 ...S.9.....
0030 51 48 9a c8 00 06 dd d0 40 00 00 00 00 00 00 00 QH.....@.....
0040 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0050 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0060 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00

File: "C:\Users\jeffotto\AppData\Local\Temp..." Packets: 38680 Displayed: 38680 Marked: 0 Dropped: 0 Profile: Default

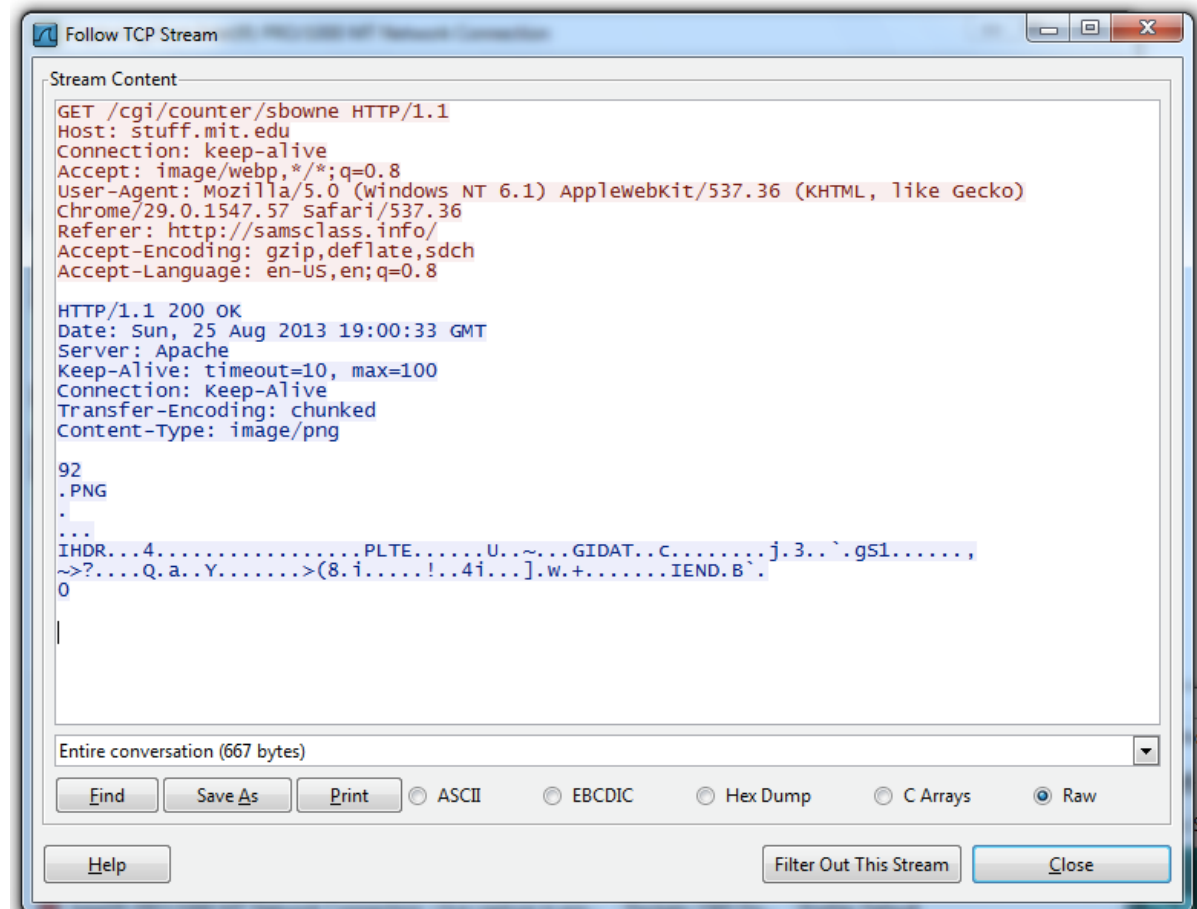
Packet
capture

Packet
detail

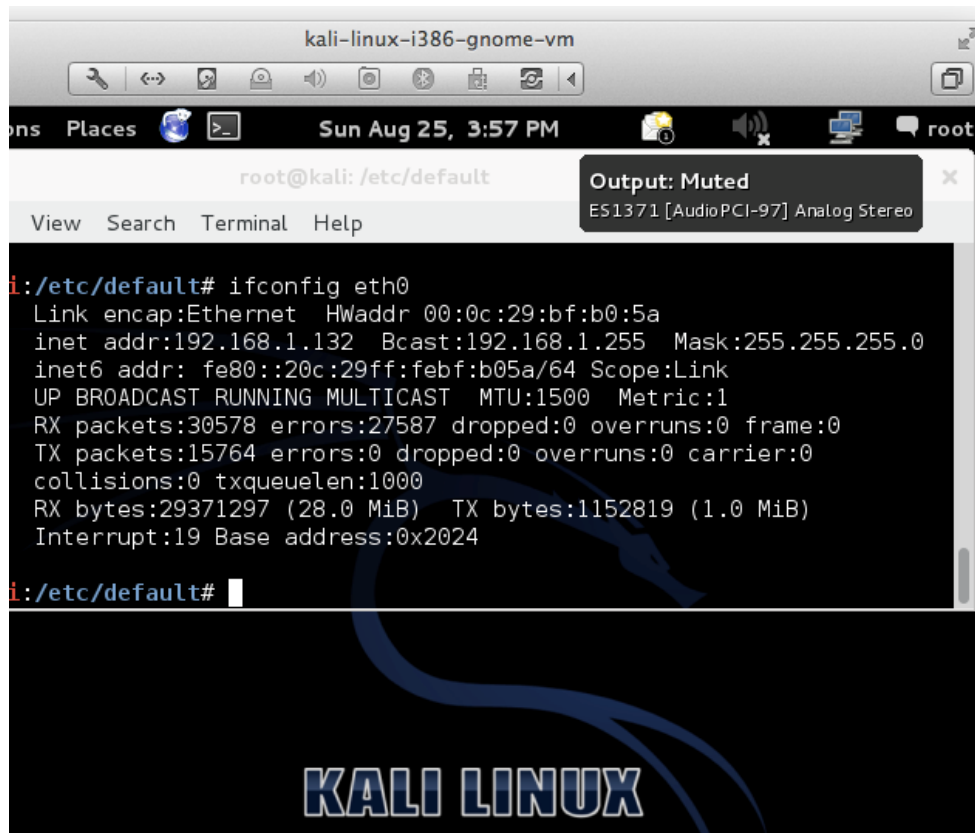
Raw data

Follow TCP Stream

- Can save files from streams here too



inetsim

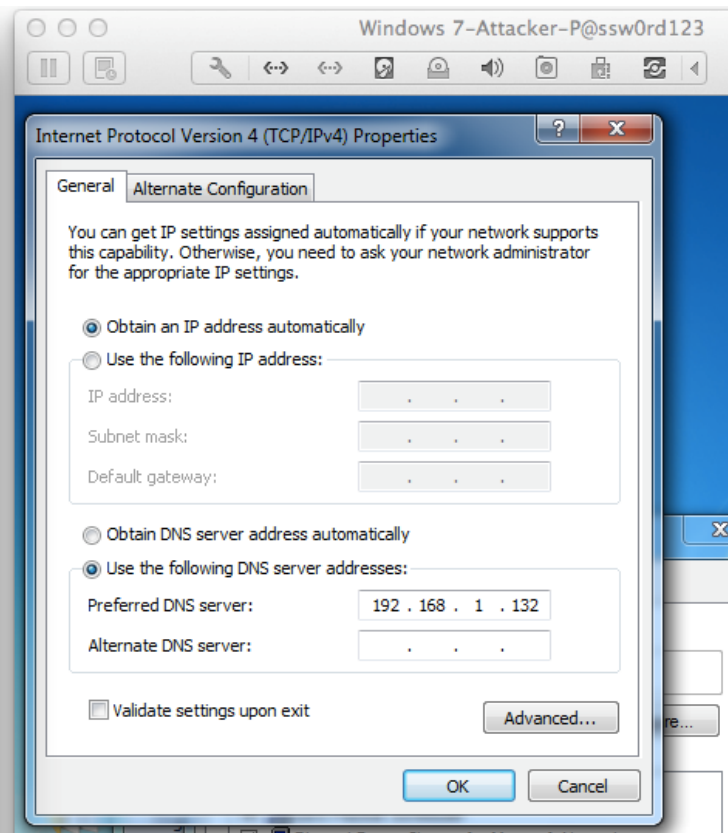


The screenshot shows a Kali Linux terminal window titled 'kali-linux-i386-gnome-vm'. The terminal prompt is 'root@kali: /etc/default'. The command 'ifconfig eth0' has been executed, and the output is displayed. The output shows the network configuration for the 'eth0' interface, including the link type (Ethernet), hardware address (00:0c:29:bf:b0:5a), IP address (192.168.1.132), broadcast address (192.168.1.255), netmask (255.255.255.0), and various statistics like RX and TX packets, errors, and bytes. The terminal also shows a 'KALI LINUX' logo at the bottom.

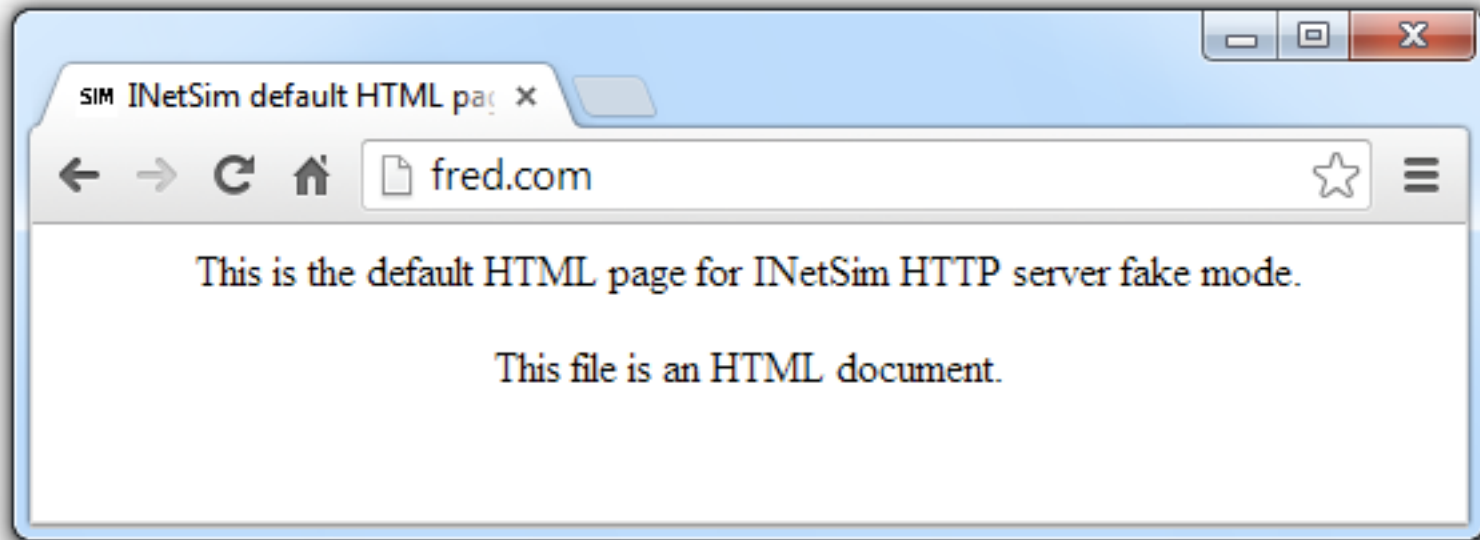
```
root@kali: /etc/default
View Search Terminal Help

i:/etc/default# ifconfig eth0
Link encap:Ethernet HWaddr 00:0c:29:bf:b0:5a
inet addr:192.168.1.132 Bcast:192.168.1.255 Mask:255.255.255.0
inet6 addr: fe80::20c:29ff:febf:b05a/64 Scope:Link
UP BROADCAST RUNNING MULTICAST MTU:1500 Metric:1
RX packets:30578 errors:27587 dropped:0 overruns:0 frame:0
TX packets:15764 errors:0 dropped:0 overruns:0 carrier:0
collisions:0 txqueuelen:1000
RX bytes:29371297 (28.0 MiB) TX bytes:1152819 (1.0 MiB)
Interrupt:19 Base address:0x2024

i:/etc/default#
```



INetSim Fools a Browser



Using the Tools

- Procmon
 - Filter on the malware executable name and clear all events just before running it
- Process Explorer
- Regshot
- Virtual Network with INetSim
- Wireshark

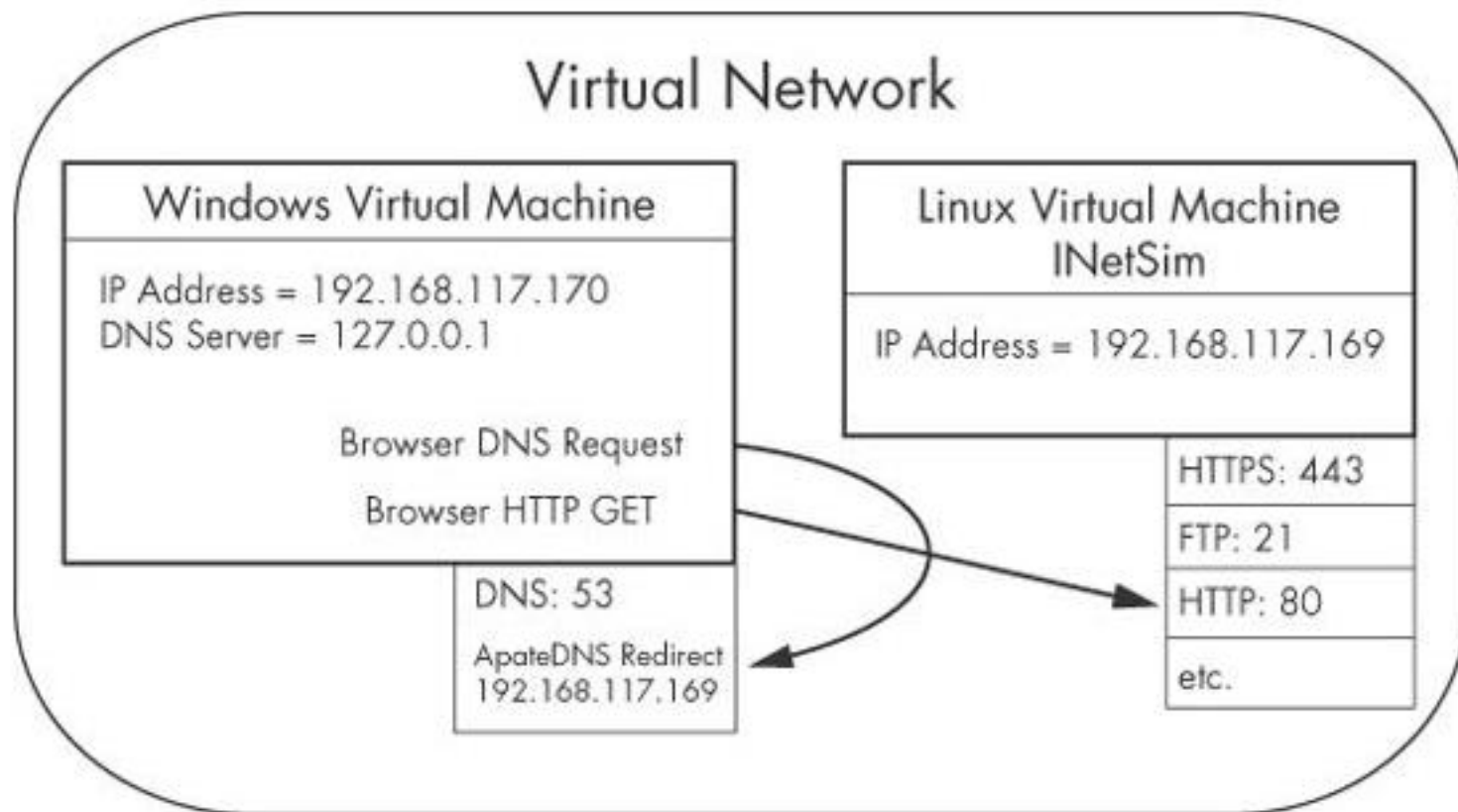


Figure 4-12. Example of a virtual network