OXFORD BROOKES UNIVERSITY

COMP6016
– Introduction to Malware

Dr. MUHAMMAD HILMI KAMARUDIN

OXFORD BROOKES UNIVERSITY

Lecture Learning Outcomes

By the end of today's lecture and practical you should

- Understand the common types of malware
- Have an appreciation of the history of malware
- Understand the common behaviour and attack vectors used in malware
- Be aware of the process of malware analysis and the stages involved

OXFORD BROOKES UNIVERSITY

What is malware?

- Malware (MALicious softWARE) is software used to **maliciously** compromise a computer or network
- Termed coined in 1990 by Yisrael Radai (prior to that, most things were just called **computer viruses**)
- Exactly what is malware will depend on context, but includes viruses, worms, trojans, rootkit, spyware, ransomware, and scareware. It may also include adware and PUPs (Potentially Unwanted Programmes)
- A programme in one context may be malware, but in another, may be a useful piece of software (e.g. mobile phone rooter)

The jailbreak of apple devices exposed a port and using the default root password we could just access the terminal… We need to know what jailbreaking does to do it safely

Rooting (Android) and jailbreaking (iOS) can be rooted using programs we consider malware! Makers don't want us to unlock terminals as that makes support harder…

OXFORD BROOKES UNIVERSITY

Modern malware

- Modern malware - is often highly sophisticated is often used commercially – it is big business for organised crime
- May be used by nation states to further national goals or to ensure national security (e.g. Stuxnet)

Targeted Iranian nuclear PLCs to break them! It was the first malware capable of crippling hardware and was believed to be a joint effort between the NSA, CIA and MOSSAD

- Often relies on a complex **command and control** system

Malware classification

OXFORD BROOKES UNIVERSITY

There are multiple ways of classifying malware, common ways are by behaviour, target platform, attack directive, or a combination of the others

One common classification is -

- Infectors (basic computer virus)
- Network worms
- Trojan horse
- Backdoors
- Remote access Trojans
- Information stealers
- Ransomware
- Scareware
- Mobile malware (special class by platform, not by behaviour)
- Greyware (PUPs, adware, etc.)

OXFORD BROOKES UNIVERSITY

Computer Virus classification

- Computer Viruses are normally classified based on their infection strategy.
- The most common classification is -
- File Infectors – infected the file on disk immediately (direct infector) or waiting in memory to infect a particular programme (memory-resident)
- Boot-Sector virus – infected the boot part of the disk
- Multipartite virus – a virus that is both a file infector and a Boot-Sector virus, or a virus that can infect multiple platforms

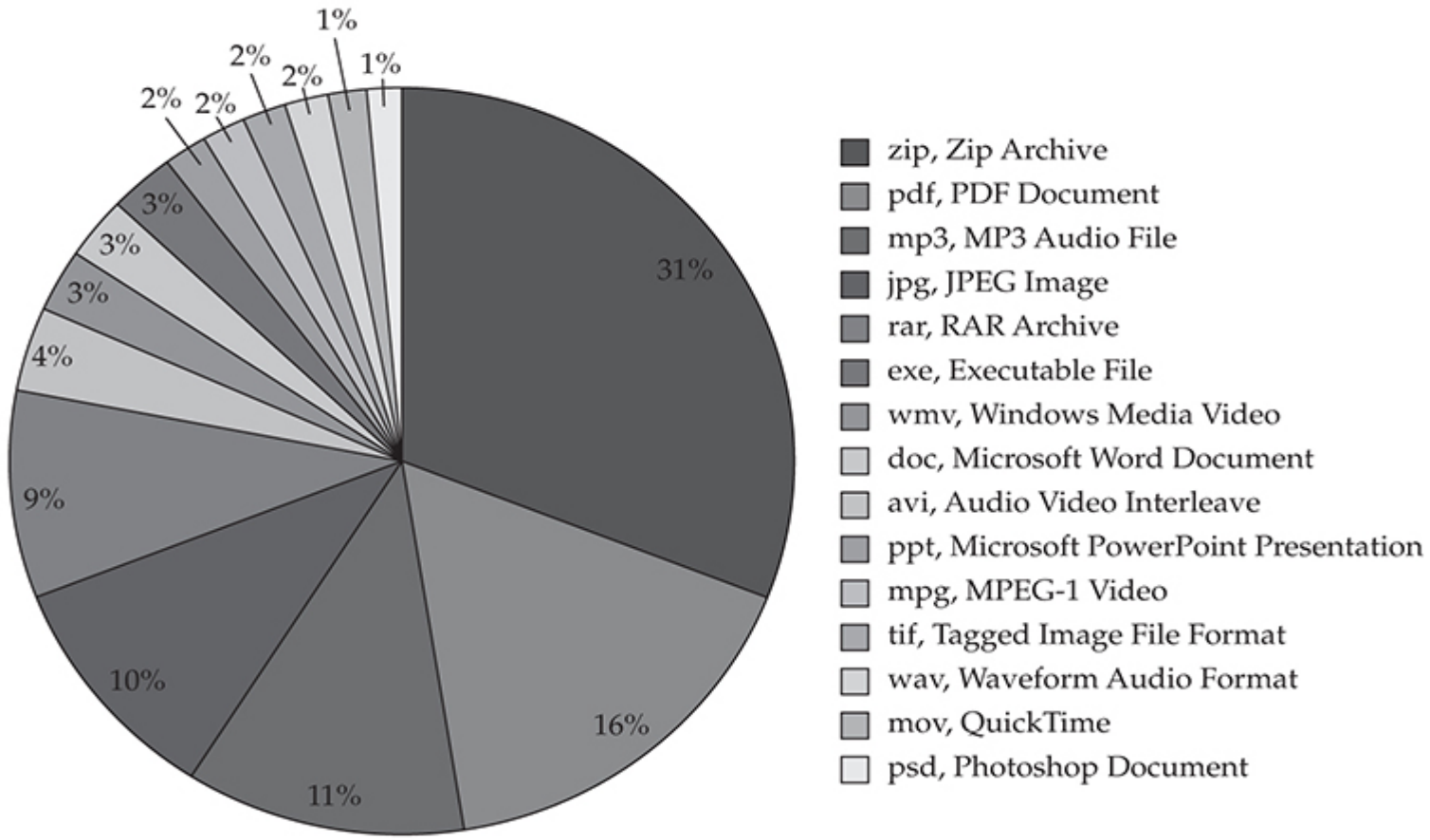
Computer Virus classification – direct infectors

OXFORD BROOKES UNIVERSITY

- Direct infector viruses actively search for a particular file(s) to infect
- These viruses are simple to write and simple to detect
- Infection occurs by one of -
- Overwriting – simply replace target executable with virus. Trivial to detect and often causes significant damage
- Companion – takes advantage of the order of execution priority on Microsoft OSes (COM, EXE, BAT) and adds itself as the highest priority (moving the original file if needed). Virus is normally made a hidden file
- Parasitic – the virus adds itself to the disk image of the original file and changes the execution start point

Computer Virus classification – direct infectors

OXFORD BROOKES UNIVERSITY



Common malware file types for Windows from - Sean M. Bodmer; Michael A. Davis; Christopher C. Elisan; Aaron LeMasters (2016) *Hacking Exposed Malware & Rootkits: Security Secrets and Solutions, Second Edition, 2nd Edition*

OXFORD BROOKES UNIVERSITY

Computer Virus classification – Memory and Boot sector viruses

- Memory-resident viruses – similar to direct infectors (they use the same infection strategy) except that they do not infect files on execution.

Terminate and Stay Resident Program
https://en.wikipedia.org/wiki/Terminate_and_stay_resident_program

- Memory-resident viruses hide in memory (using DOS's TSRs) and execute the host file when it is executed
- Boot sector viruses rewrite the boot sector to point to the virus. When computer is booted, the virus runs in memory (often infecting other removable media) and then runs the normal boot loader

The boot sector is just 512 B long so the virus code is usually located in other sectors.

- Boot sector viruses were very common in early days of computing

OXFORD BROOKES UNIVERSITY

Computer Virus classification – targets

- Another common way of classifying viruses is by target -
- Executable virus – targets executables on the machine. Almost always (with only a few exceptions) platform specific. Most likely to be able to escape detection

Just like Word Macros. They let us automate tasks but provide a new attack vector…

- Macro virus – targets data files for software that supports an internal executable language. Multi-platform. Easy to detect and avoid

Visual Basic is an example that runs on Windows

- Script virus – can be considered to be a general form of the macro virus. Runs in scripting environment provided by the host and normally exploits vulnerabilities in 3rd party software. May be multi-platform. Varying degrees of difficulty to detect and avoid (depending on scripting language)

Network worms

Huge infection rate thanks to leveraging the network infraestructre !

OXFORD BROOKES UNIVERSITY

- AKA computer worms, or just worms
- Utilises network to spread enabling fast and large infection rates
- May use social engineering or system vulnerabilities to spread
- Normally classified by propagation technique
- Mass mailers – spread via email. Often uses social engineering and users address book. Normally requires user to activate it
- File sharing & IRC worms – spread via file sharing systems (e.g. torrents) and rely on user activation
- IM Worms – works in a similar fashion to mass mailers

They leverage Internet Messaging systems! Just like email worms, they usually contain a link to download the malicious file which will be automatically executed too…

- LAN Worms – uses shared folders on a network. May exploit vulnerabilities in OS or common enterprise software

They look for folders with sharing enabled! Careful with public folders…

- Internet worms – compromises vulnerable internet services on victim machine

They look for the netoworked services the target host is running

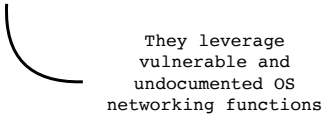
OXFORD BROOKES UNIVERSITY

Trojan Horse

- Overloaded term, two main definitions -
- Destructive malware that hides itself
- Non-replicating malware
- Key concept is that the malware actively tries to avoid detection

OXFORD BROOKES UNIVERSITY

Backdoors & RATs



- A backdoor is malware that enables an attacker to re-enter a compromised system
- Bypasses conventional authentication systems and normally tries to hide it's operation
- A RAT (Remote Access Trojan) is a backdoor with a UI
- RATs can be thought of as the first point and click malware systems

OXFORD BROOKES UNIVERSITY

Information Stealers

We'll need to find some malware on an infected XP machine with one of them being a keylogger for coursework 2! We just need to find malware for coursework 2.

- Malware that steals any type of information
- Most common types are -
- Keyloggers – capture and log keystrokes. Good for username and password capture

We have software and hardware keyloggers. The HW keylogger is usually a device attached to the keyboard itself. The SW version listens for keystrokes sent to an I guess OS daemon registering keystrokes.

The information will be retrieved later

Usually triggered by clicks, keyboard inputs or regular intervals. They try to circunvent virtual onscreen keyboards!

- Desktop recorders – similar to keyloggers except that screen is also captured at predefined intervals. Good for GUI based systems, but large overhead for images
- Memory scrapers – capture information in memory. Modern OSes make this much harder to be effective

Everything is decrypted in memory so that in can be processed. That's why memory scrapers try to look here!

- Network loggers – capture network traffic. Modern switched networks render this of limited use

OXFORD BROOKES UNIVERSITY

Ransomware

- Information or system is held to ransom
- Bitcoins are now often used as financial exchange mechanism
- Most common types are -
- Data encryption – data is encrypted and ransom is paid to obtain key to decrypt. A sensible backup strategy will often help against this

People are usually not encouraged to pay the ransom as they'll be likely left with unencrypted data anyway…

- Data destruction – user is threatened with data destruction (normally reformat system). In majority of cases it is just a threat

Usually after encrypting the data so that the user cannot copy it and try to decode it afterwards and stuff…

- User lockout – user is locked out until they pay. Can work by just changing the password, or by replacing authentication mechanism. Easily repaired by experienced user

We need a different system to pay xD

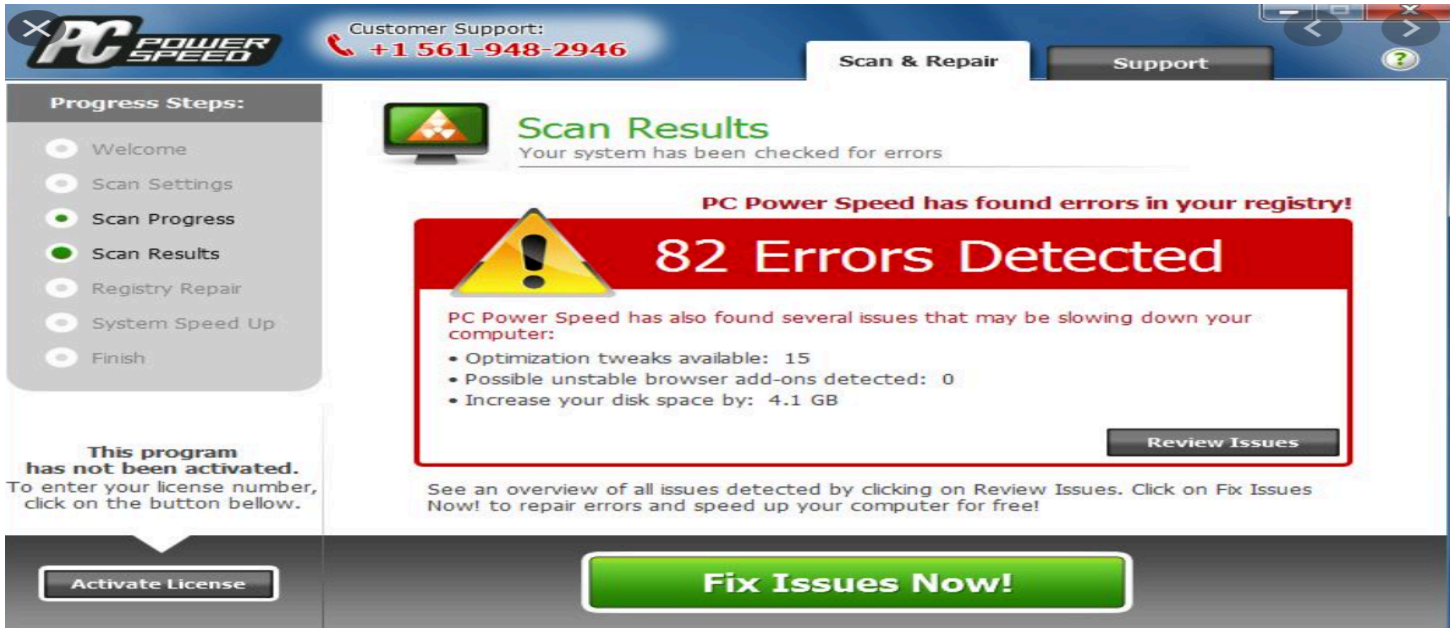
Scareware

It usually asks for:
- Downloading additional programs (i.e. more malware)
- The user's credit card…
- A payment

Errors like the one in the slice show quite often when surfing the web!

OXFORD BROOKES UNIVERSITY

- Similar to ransomware
- User is warned of dangerous system activity (e.g. virus infection, malware, illegal user behaviour, etc.) and required to pay for software or fine
- In most cases there is no actual penalty for ignoring the message (though may be hard to get rid of)
- If additional software is downloaded, it will often install additional malware



OXFORD BROOKES UNIVERSITY

Fakeware

- Fakeware passed itself along as a legitimate program update.
- Disguise as an update of popular software
- If users believe a fakeware is a legitimate update, it has a higher chance of being installed on the system.

Figure: Fakeware disguising itself as a Flash Player update.

Greyware

OXFORD BROOKES UNIVERSITY

- Software that has the potential to be classed as malware depending on the circumstances
- Main categories are –

Lots of apps actually come packaged with adware to have some kind of revenue! (i.e. ads on free apps)

- Adware – most common type of greyware. Directed advertising using observed behaviour of the user. May come as part of another piece of software. Nuisance and privacy issues
- Spyware – commercial information stealers. Collect information about the user often without their knowledge. Often used by parents and occasionally by companies

Huge privacy issues!

- Hacktools- short for hacking tools. These are programs that give users access to a target system. Hacktools are similar to network administrator tools. Most of them function the same way. The only difference is the intent for which they are used. A network administrator tool can be used to manage a network to make sure that everything is running smoothly, but in the wrong hands, that same tool can be used to compromise a network.

Pretty much like what happens with gins…

OXFORD BROOKES UNIVERSITY

Some recent malware

Malware	Year	Injection Technique	Propagation Techniques
StormWorm	2007–2008	Email attachments File execution	File dropper Overwrite/deletion P2P C2 structure and Fast Flux communication chaining
AutoIT	2008	File execution	Copies generated onto removable drives by overwriting the autorun.inf
Downadup	2009	File execution	File transfer, file sharing, copying itself across network shares or shares with weak passwords
Bacteraloh	2009	File execution (P2P network-based)	Disguised as a crack utility that a user downloads and executes locally
Koobface	2009	Client-side exploit	Spread through social-networking sites with a loaded URL linked to the malware on sites such as Facebook, MySpace, Friendster, and LiveJournal
Stuxnet	2010	File execution (vulnerabilities)	Tailored to specifically attack a nuclear infrastructure
SpyZeuS	2010	Email attachments File execution	A combination of banking Trojans Zeus and SpyEye
Duqu	2011	File execution (vulnerabilities)	Multifile malware with each file having different functionalities, including information-stealing capabilities
Flame	2012	File execution	An attack toolkit discovered in 2012 but believed to be in operation since 2010 Sniffs network traffic, logs keystrokes, records audio conversations, and takes screenshots
CryptoLocker	2013	File execution	Ransomware that poses as if coming from the FBI, denies access to system files, and then asks for a ransom
BlackEnergy	2014	Email attachments File execution	Collects data from the compromised system's hard drive

Sean M. Bodmer; Michael A. Davis; Christopher C. Elisan; Aaron LeMasters (2016) *Hacking Exposed Malware & Rootkits: Security Secrets and Solutions, Second Edition, 2nd Edition*

Protective Mechanism

Main idea: The attacker will do everything he/she can to protect the malware forever or at least for as long as possible.

OXFORD BROOKES UNIVERSITY

Just relying on assembly means the analyst will take forever to work out what the malware does…

That's why classifying malware is so important: it'll let us what to know what to look for faster!

- The attacker know the risk of malware being captured.
- The attacker will look for different evasion technologies to protect their Malware
- The goal is to avoid detection and delay the action from Malware Analyst
- This is important for the analyst to be able to recognise and mitigate these protective mechanism

OXFORD BROOKES UNIVERSITY

Propagation strategies

- Many modern AV and anti-malware systems rely on signatures to detect malware
- A signature is a sequence of bytes that uniquely identifies the malware
- Modern malware actively tries to change it's signature to avoid detection
- To do this it relies on two main techniques -
- Metamorphosis – the malware changes it's code
- Obfuscation – the malware tries to hide it's code
- The main idea is to buy more time

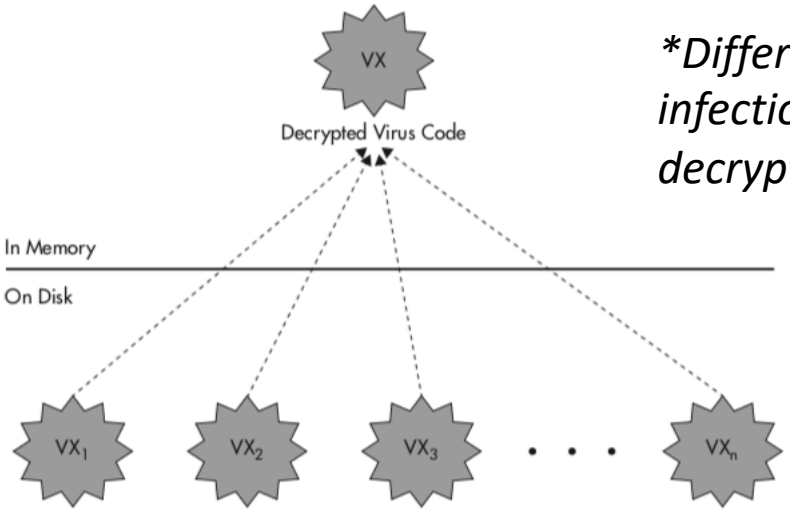
Polymorphic Malware

This still needs to be decrypted in memory so we can just detonate the malware and inspect what it loads onto memory. It dodges static scanning but NOT dynamic analysis!

OXFORD BROOKES UNIVERSITY

- Encrypted Malware has three components:
  - The encryption/decryption engine
  - The encryption/decryption key
  - The malware code

- Attacker introduce mutation engine.
- Mutation engine is part of the malware code which use to alters the code of another application without changing their functions
-



**Different generations of polymorphic infections look the same in memory when decrypted.*

Metamorphic malware

OXFORD BROOKES UNIVERSITY

- Metamorphic malware changes the bytes in it's body to avoid detection
- Simple metamorphosis can be easily enumerated by the anti-malware system
- More complex metamorphic malware uses encryption to avoid detection
- With metamorphism, each malware infection is totally different, both on disk and in memory.

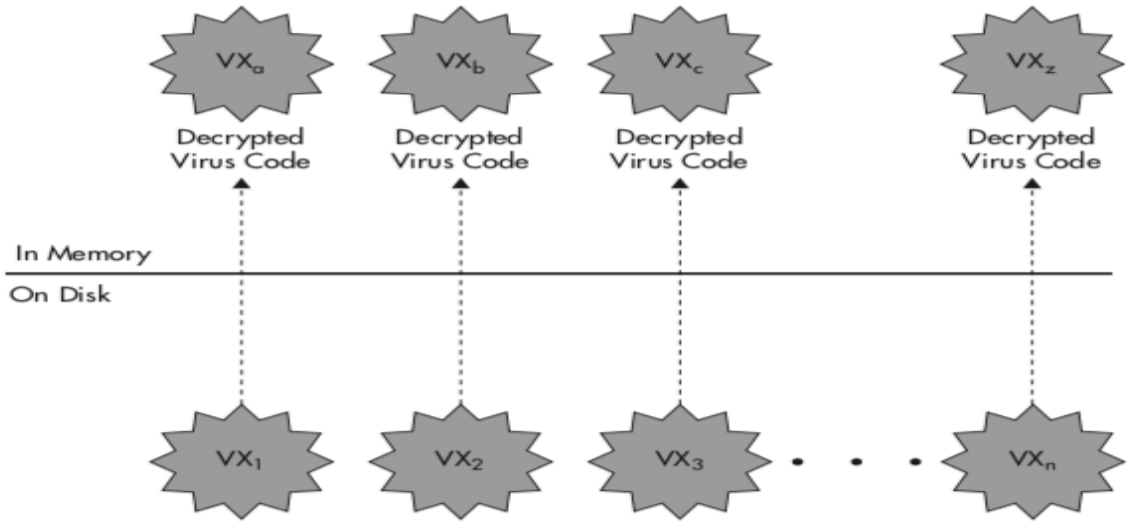


Figure 2: Metamorphic malware infections differ on disk and in memory.

OXFORD BROOKES UNIVERSITY

Anti-Reversing

- The best way to understand Malware is through reversing.
- Time and effort are the enemies
- Example (Ultimate Packer for Executables), enabled attackers to easily distribute malware with far lesser resistance

Once the packing routine was reversed UPX lost its appeal…

-However, the malware packed by UPX can be unpacking using OllyDbg in a minutes .

- Rule of thumb- The successful anti reversing technique is when it takes a reverser a longer time to understand the malware than the need of the malware to survive.

-The idea is to make reversing process as difficult as possible

It can sometimes seem impossible!

If we cannot decompile the code and have access to the sources there is nothing we can reverse…

Example of most common anti-reversing techniques:

-Anti-Decompilers

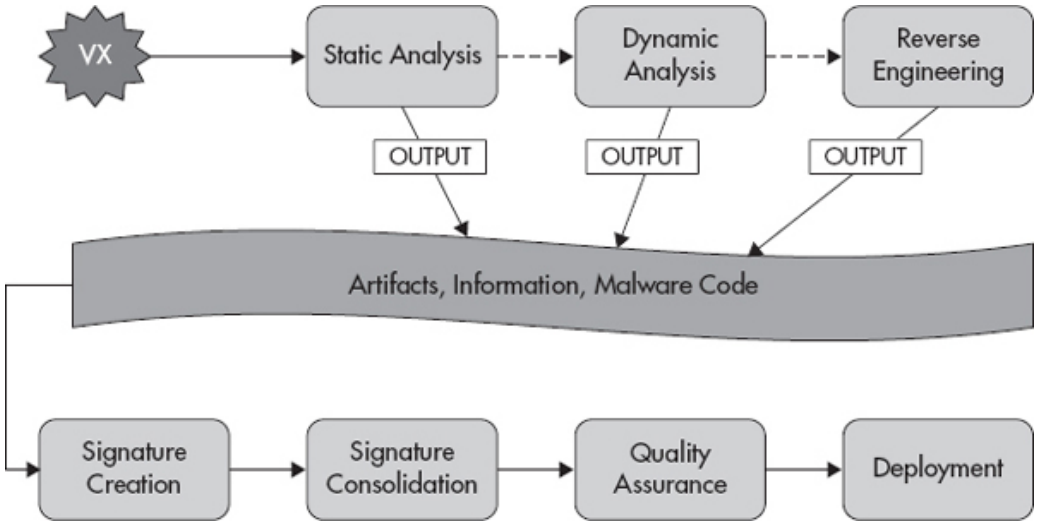
-Anti-Disassemblers

They only break the most common dissasemblers but other more niche ones may work. We are just trying to buy time!

Dealing with Malware

- In order to successfully deal with Malware, it must be analysed and a method of detection and removal (if possible) established
- Malware must be captured (normally from an infected system) and then transferred to a safe system (a **sandbox**) for processing
- Care must be taken to ensure that the malware cannot escape from the sandbox

Sandbox -> Safe environment on which to try things out! Sandboxes are usually placed around the application only unlike VMs which are placed "around" the entire OS. In other words, sandboxes are in some way lightweight versions of VMs. Thing is we usually employ VMs to try and analyse the samples on serveral platforms! Some others prefer sandboxes as they take up less resources… Everybody suit themselves!



Christopher Elisan (2012) *Malware, Rootkits & Botnets: A Beginner's Guide*

OXFORD BROOKES UNIVERSITY

Malware Analysis

- Malware analysis is the process of investigating and gathering information about a piece of malware. It is the main mechanism for providing information for an anti-malware system
- Malware analysis normally stops short of reverse engineering
- Malware analysis involves two key stages -
- Static Analysis
- Dynamic Analysis

OXFORD BROOKES UNIVERSITY

Malware analysis – Static analysis

The program is NOT run!

Describes the process of analyzing the code or structure of a program to determine its function. The program itself is not run at this time.

Using AV tools to confirm maliciousness

-Run through multiple antivirus programs

-Problem when the malware writers can easily modify their code, hence goes undetected by AV software

-Website such as VirusTotal will allow us to upload a suspicious file for scanning by multiple Antivirus engines.

-VirusTotal also generates a report that provides the total number of engines that marked the file as malicious, the malware name, and, if avail- able, additional information about the malware

OXFORD BROOKES UNIVERSITY

Hashing: Fingerprint for Malware

- Hashing is a common method used to uniquely identify malware
- Provide unique hash that identifies the malware (fingerprint)
- For example, using the freely available md5deep program to calculate the hash of the Solitaire program that comes with Windows would generate the following output:

C:\>**md5deep c:\WINDOWS\system32\sol.exe 373e7a863a1a345c60edb9e20ec3231**
c:\WINDOWS\system32\sol.exe

Once have the hash value, we can search the hash online to see either the hash has previously been identified

OXFORD BROOKES UNIVERSITY

Finding Strings

- A *string* in a program is a sequence of characters such as “the.”
- A program contains strings if it prints a message, connects to a URL, or copies a file to a
specific location.
- Fortunately, most invalid strings are obvious, because they do not represent legitimate text.

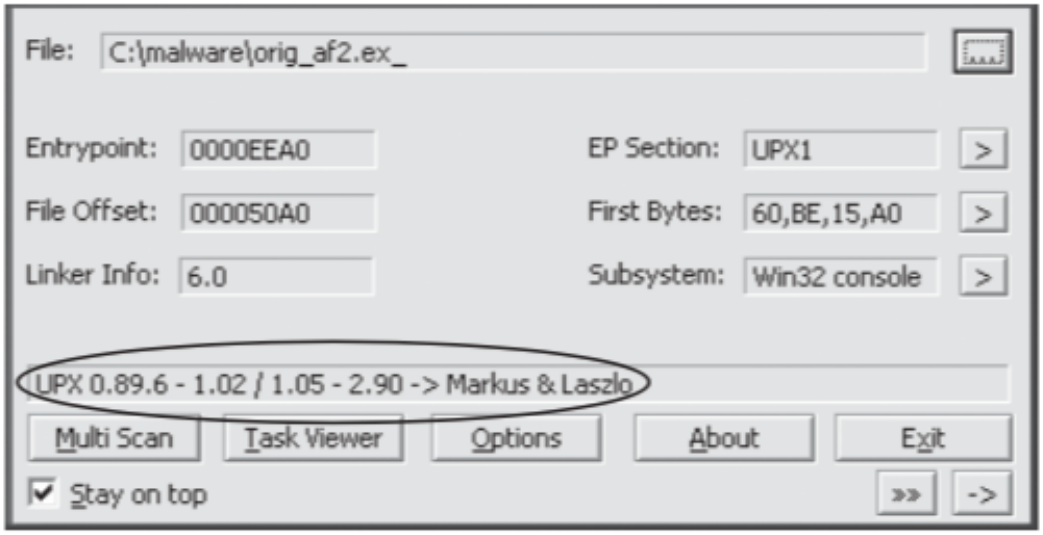
For example, the following excerpt shows the result of running Strings against the file *bp6.ex_*:

```
C:>strings bp6.ex_
VP3
VW3
t$@
D$4
99.124.22.1 ❹
e-@
GetLayout ❶
GDI32.DLL ❸
SetLayout ❷
M}C
Mail system DLL is invalid.!Send Mail failed to send message. ❺
```

OXFORD BROOKES UNIVERSITY

Detecting Packers with PEiD

PEiD can be use to detect packed files. The development and support for PEiD has been discontinued since April 2011. But it still the best tool available for packer and compiler detection.



Example of file orig_af2.ex_ analysed using PEiD. PEiD has identified the file as being packed with UPX version 0.89.6-1.02 or 1.05-2.90.

To unpack malware packed by UPX, simply download UPX (*http:// upx.sourceforge.net/*) and run it like so, using the packed program as input:

$ upx -d *PackedProgram.exe*

OXFORD BROOKES UNIVERSITY

Malware analysis – dynamic analysis

- Dynamic analysis is any examination performed after executing malware. It is the second step in the malware analysis process.
- Dynamic analysis is typically performed after basic static analysis has reached a dead end, whether due to obfuscation, packing, or the analyst having exhausted the available static analysis techniques.
- Unlike static analysis, dynamic analysis observe the malware's true functionality, for example, the existence of an action string in a binary does not mean the action will actually execute
- It also the efficient way to identify malware functionality, for example the keylogger, dynamic analysis can allow you to locate the keylogger's log file on the system, discover the kinds of records it keeps, decipher where it sends its information, and so on.

OXFORD BROOKES UNIVERSITY

Malware analysis – sandbox

- The sandbox is a safe environment for the execution of malware
- The sandbox is heavily instrumented, but in all other respects is a standard system
- Most modern sandboxes are VMs as this allows greater automation, but malware with anti-forensic capabilities can detect VMs and not execute under those circumstances
- Multiple VMs can be used to analyse the malware's behaviour on a network
- NB a sandbox can be automated, but the automation will not tell you if a programme is malware or not – it will just tell you it's behaviour

OXFORD BROOKES UNIVERSITY

Monitoring with Process Monitor

- Process Monitor, or procmon, is an advanced monitoring tool for Windows that provides a way to monitor certain registry, file system, network, process, and thread activity.

- Procmon monitors all system calls it can gather as soon as it is run. Because many system calls exist on a Windows machine (sometimes more than 50,000 events a minute) it's usually impossible to look through them all.

- To stop procmon from capturing events, choose **File>Capture Events**. Before using procmon for analysis, first clear all currently captured events to remove irrelevant data by choosing **Edit>Clear Display**. Next, run the subject malware with capture turned on. After a few minutes, you can discontinue event capture.

OXFORD BROOKES UNIVERSITY

ProcMon Display

- Procmon displays configurable columns containing information about individual events, including the event's sequence number, timestamp, name of the process causing the event, event operation, path used by the event, and result of the event.

- Figure 2 shows a collection of procmon events that occurred on a machine running a piece of malware named *mm32.exe*. The word *SUCCESS* in the Result column tells you that this operation was successful.

Seq.	Time ...	Process Name	Operation	Path	Result	Detail
200	1:55:31.	mm32.exe	CloseFile	Z:\Malware\mw2mmgr32.dll	SUCCESS	
201	1:55:31.	mm32.exe	ReadFile	Z:\Malware\mw2mmgr32.dll	SUCCESS	Offset: 11,776, Length: 1,024, I/O Flag
202	1:55:31.	mm32.exe	ReadFile	Z:\Malware\mw2mmgr32.dll	SUCCESS	Offset: 12,800, Length: 32,768, I/O Fla
203	1:55:31.	mm32.exe	ReadFile	Z:\Malware\mw2mmgr32.dll	SUCCESS	Offset: 1,024, Length: 9,216, I/O Flags
204	1:55:31.	mm32.exe	RegOpenKey	HKLM\Software\Microsoft\Windows NT\CurrentVersion\Image File Exec	NAME NOT ...	Desired Access: Read
205	1:55:31.	mm32.exe	ReadFile	Z:\Malware\mw2mmgr32.dll	SUCCESS	Offset: 45,568, Length: 25,088, I/O Fla
206	1:55:31.	mm32.exe	QueryOpen	Z:\Malware\imagehlp.dll	NAME NOT ...	
207	1:55:31.	mm32.exe	QueryOpen	C:\WINDOWS\system32\imagehlp.dll	SUCCESS	CreationTime: 2/28/2006 8:00:00 AM,
208	1:55:31.	mm32.exe	CreateFile	C:\WINDOWS\system32\imagehlp.dll	SUCCESS	Desired Access: Execute/Traverse, S
209	1:55:31.	mm32.exe	CloseFile	C:\WINDOWS\system32\imagehlp.dll	SUCCESS	
210	1:55:31.	mm32.exe	RegOpenKey	HKLM\Software\Microsoft\Windows NT\CurrentVersion\Image File Exec	NAME NOT ...	Desired Access: Read
211	1:55:31.	mm32.exe	ReadFile	Z:\Malware\mw2mmgr32.dll	SUCCESS	Offset: 10,240, Length: 1,536, I/O Flag
212	1:55:31.	mm32.exe	CreateFile	C:\Documents and Settings\All Users\Application Data\mw2mmgr.txt	SUCCESS	Desired Access: Generic Write, Read
213	1:55:31.	mm32.exe	ReadFile	C:\$Directory	SUCCESS	Offset: 12,288, Length: 4,096, I/O Flag
214	1:55:31.	mm32.exe	CreateFile	Z:\Malware\mm32.exe	SUCCESS	Desired Access: Generic Read, Dispo
215	1:55:31.	mm32.exe	ReadFile	Z:\Malware\mm32.exe	SUCCESS	Offset: 0, Length: 64

Figure 2: Procmon mm32.exe example

Processes with Process Explorer

OXFORD BROOKES UNIVERSITY

- The Process Explorer, free from Microsoft, is an extremely powerful task manager that should be running when you are performing dynamic analysis.
- It can provide valuable insight into the processes currently running on a system.
- Process Explorer can be use to list active processes, DLLs loaded by a process, various process properties, and overall system information. On top of that, it also can be use to kill a process, log out users and launch/validate processes

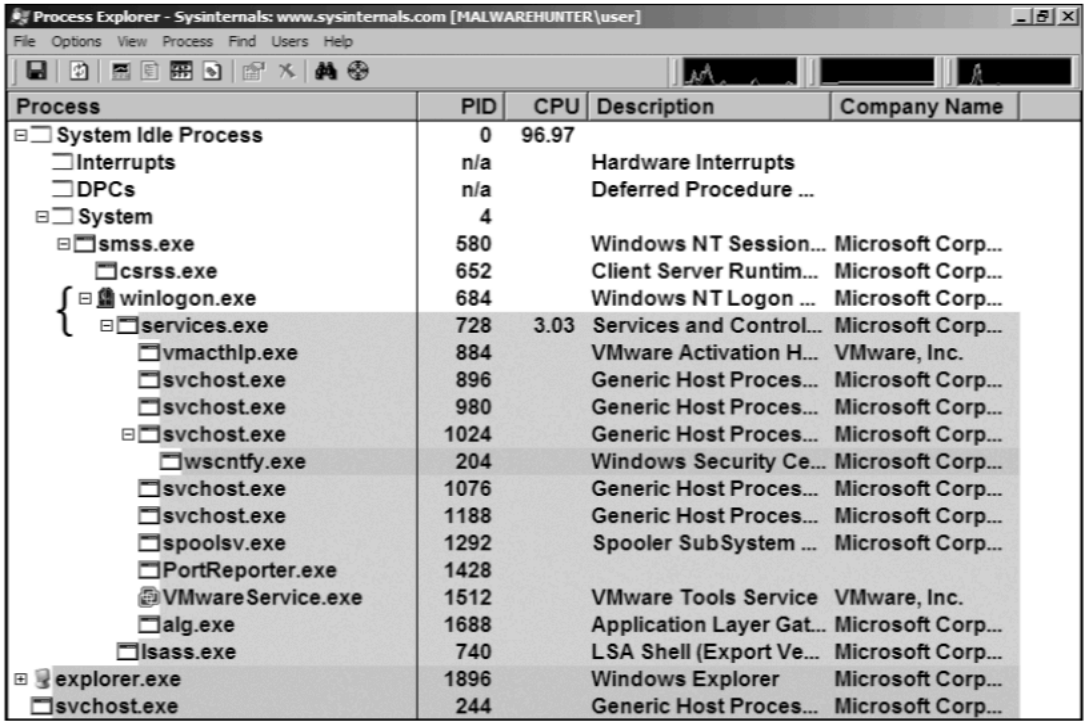


Figure 3: Process Explorer examining svchost.exe malware

Packet Sniffing with Wireshark

OXFORD BROOKES UNIVERSITY

- Wireshark is an *open source sniffer*, a packet capture tool that intercepts and logs network traffic. Wireshark provides visualization, packet-stream analysis, and in-depth analysis of individual packets.
- Can be used for both good or evil
- It can be used to analyze internal networks and network usage, debug application issues, and study protocols in action. But it can also be used to sniff passwords, reverse-engineer network protocols, steal sensitive information, and listen in on the online chatter at your local coffee shop.

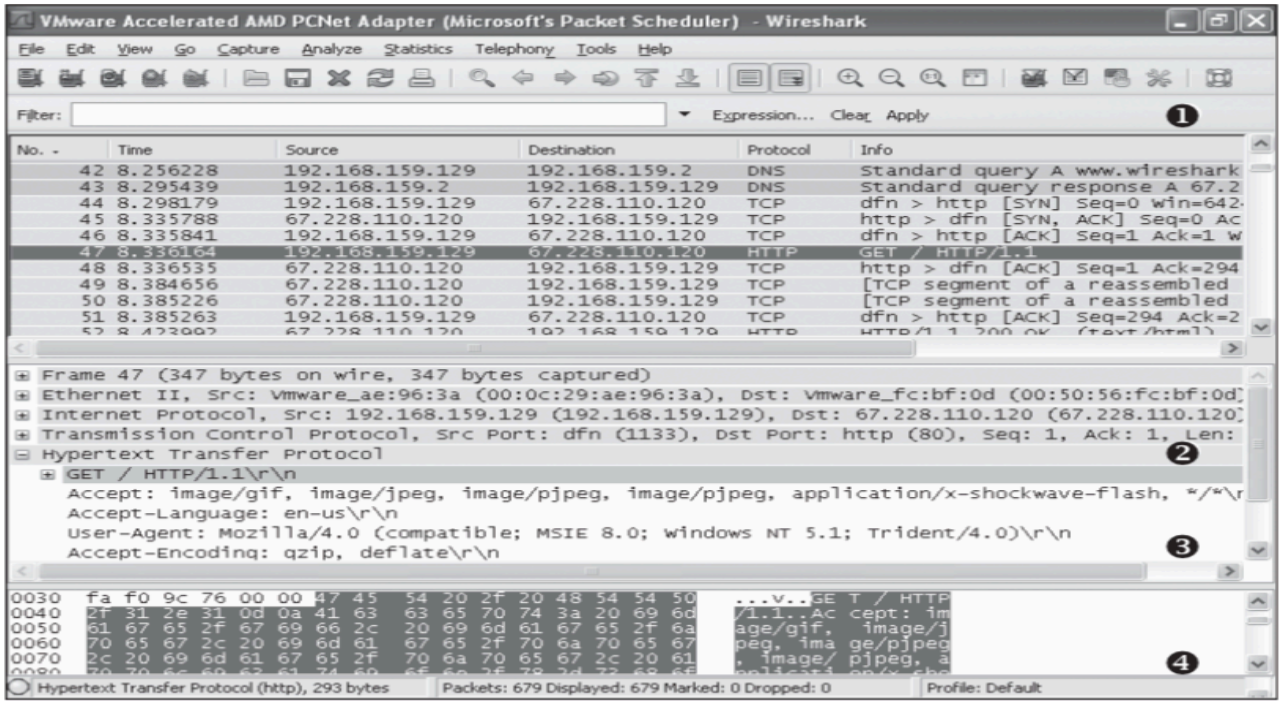


Figure 4: *Wireshark DNS and HTTP example*

Unusual netowkr traffic -> Possibility there's a malaware sample doing nasty stuff!

Packet Sniffing with Wireshark cont..

OXFORD BROOKES UNIVERSITY

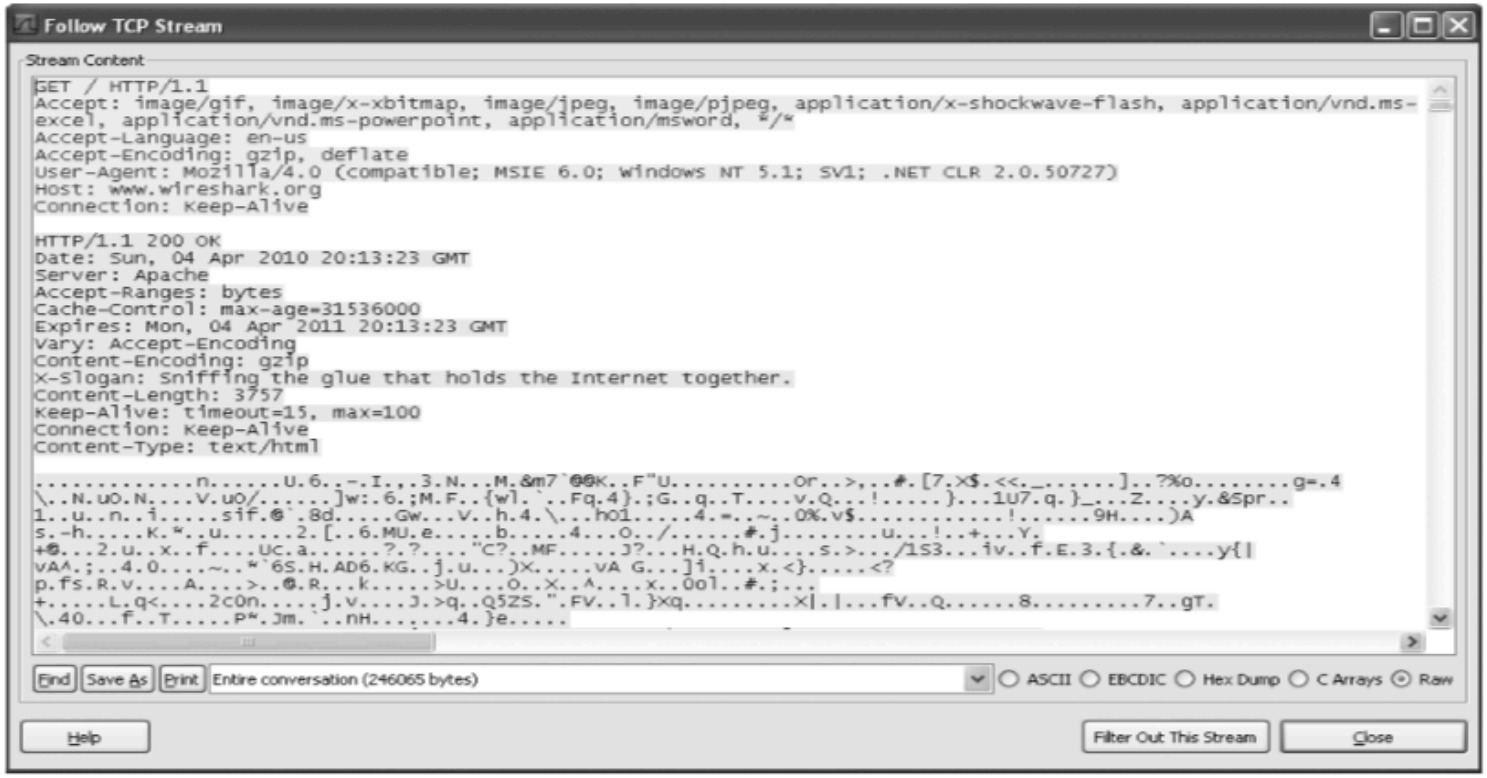


Figure 5:*Wireshark's Follow TCP Stream window*

.To use Wireshark to view the contents of a TCP session, right-click any TCP packet and select **Follow TCP Stream**. As you can see in Figure 5, both ends of the conversation are displayed in session order, with different colors showing each side of the connection. To capture packets, choose **Capture>Interfaces** and select the interface you want to use to collect packets. Options include using promiscuous mode or setting a capture filter.