**Lab 12: Malware Analysis**

Submit your lab to Canvas. If you have any questions, email Dr. Vance at anthony@vance.name.

## **Part 1: INTRODUCTION to WANNACRY**

### Overview

In this lab, you will create a sandbox environment to analyze malware called WannaCry. As a result, **be extremely cautious** on which machine you are performing this lab. Please talk to me if you have concerns.

**Warning**: This lab involves real malware. To complete this lab:

1. Run this lab within a VM.
2. Please pay attention to the network settings of the VM.
3. You are responsible for any loss of personal data. **Back up your data.**

You’ll need the Windows VM that you used in previously labs. If you still have it, you can move on to Part 2.

If you deleted the Windows VM, you can download it using the following link:

<https://www.dropbox.com/s/dvcib0leancn886/Windows%20InfoSec.ova?dl=0>

The password is “Password1” (no quotes).

Take a look at the reading assigned for the malware topic, listed on canvas, [Introduction](https://ebookcentral.proquest.com/lib/templeuniv-ebooks/reader.action?ppg=28&docID=1137570&tm=1535406480673) and [Chapter 0](https://ebookcentral.proquest.com/lib/templeuniv-ebooks/reader.action?ppg=34&docID=1137570&tm=1535406512351) of Practical Malware Analysis by Sikorski and Honig. The book is accessible for beginners and is also a handy reference for more advanced analysts.

## **PART 2: ANALYZE the MALWARE**

In this lab, you’ll analyze WannaCry. This program made international headlines last summer when it [wreaked havoc across the world](https://www.nytimes.com/2017/05/12/world/europe/uk-national-health-service-cyberattack.html) before being disabled by malware analyst Marcus Hutchins. It received the 2017 Pwnie Award for “Epic 0wnage”: <https://pwnies.com/winners>.

WannaCry is noteworthy for several reasons:

1. It caused a huge amount of global damage. [See an animated map of it spreading around the world](https://www.nytimes.com/interactive/2017/05/12/world/europe/wannacry-ransomware-map.html).
2. It incorporated a [zero-day exploit developed by the NSA](https://www.washingtonpost.com/business/technology/nsa-officials-worried-about-the-day-its-potent-hacking-tool-would-get-loose-then-it-did/2017/05/16/50670b16-3978-11e7-a058-ddbb23c75d82_story.html), which was stolen and posted by the Shadow Brokers hacking group, believed to [Russian operatives](https://www.nytimes.com/2016/08/17/us/shadow-brokers-leak-raises-alarming-question-was-the-nsa-hacked.html).
3. WannaCry put lives at risk when hospital systems in Europe [were disabled and patients had to have medical procedures canceled or moved to other hospitals](https://www.theguardian.com/society/2017/may/13/nhs-cyber-attack-patients-ransomware).
4. The rapid spread of WannaCry was stopped when then 22-year-old [malware analyst Marcus Hutchins](http://nymag.com/intelligencer/2018/03/marcus-hutchins-hacker.html) (aka MalwareTech) [inadvertently found the “killswitch”](https://www.wired.com/2017/05/accidental-kill-switch-slowed-fridays-massive-ransomware-attack/) for WannaCry.
5. On December 17, 2017, the White House formally attributed the WannaCry attack to North Korea. The U.K., Australia, Canada, New Zealand, and Japan, as well as Microsoft, joined the U.S. in this attribution: <https://www.wsj.com/articles/its-official-north-korea-is-behind-wannacry-1513642537>.

WannaCry is ransomware that encrypts the victim’s files and demands ransom in order to decrypt. But WannaCry does not decrypt victim’s files even after the ransom is paid.

### Confirm VM settings for your security

It is possible for WannaCry to “escape” the VM and spread to a wider network if two conditions are met:

1. The virtual machine has a network adapter that gives it access to other machines on a network, and
2. The other machines on that network are not patched against WannaCry’s spreading mechanism.

For #2, this version of WannaCry uses the [EternalBlue exploit](https://www.washingtonpost.com/business/technology/nsa-officials-worried-about-the-day-its-potent-hacking-tool-would-get-loose-then-it-did/2017/05/16/50670b16-3978-11e7-a058-ddbb23c75d82_story.html) to spread. Microsoft issued a patch for EternalBlue in March 2017. If you are running Windows and have installed updates since then, your machine is safe. If you are running a Mac you are safe because EternalBlue only works against Windows machines.

For #1, WannaCry will not spread to your host unless you changed the defaults on the Windows VM I distributed and set up a host-only network adapter. If you did, you should change it back to “NAT Network” with the “infosec-net” network name selected.

**Important:** Make sure that you take a snapshot of your Windows VM before proceeding. Having one will let you to restore your VM after WannaCry encrypts your VM. [See instructions for how to create and restore a snapshot here](https://www.virtualbox.org/manual/ch01.html#idm538).

**Warning**: If you don’t create a snapshot, you’ll need to download a new copy of the Windows VM after it is encrypted by WannaCry.

### PReliminary analysis

The WannaCry executable is zipped up in an encrypted folder on your desktop called “LIVE MALWARE WannaCry.exe”. However, **ignore this folder,** this zip contains only the “loader” or “encrypter” component. The “worm” component – the part that spreads WannaCry to other hosts – is a separate executable that I forgot to include on the VM. So, let’s grab it.

1. Make sure that Windows Defender is disabled.
2. Download the worm + loader executable [here](https://www.malware-traffic-analysis.net/2017/05/18/24d004a104d4d54034dbcffc2a4b19a11f39008a575aa614ea04703480b1022c.exe.zip).
3. Extract the malware by right-clicking the folder, choosing Extract all..., and entering the password infected. This will create a new folder called “24d004a104d4d54034dbcffc2a4b19a11f39008a575aa614ea04703480b1022c.exe.zip”, inside of which is the file:  24d004a104d4d54034dbcffc2a4b19a11f39008a575aa614ea04703480b1022c.exe. **Do not run this file yet!**

**This file is the WannaCry worm and loader (or dropper) only.**  It performs no encryption. However, it has stashed inside the WannaCry **encrypter**. The loader’s job is to drop the encrypter on a host, run it, and then scan the network and infect as many other machines as it can.

1. Open a cmd terminal via the Windows search bar. Therein, do the following:
	* Use cd to navigate into the extracted folder containing 24d004a104d4d54034dbcffc2a4b19a11f39008a575aa614ea04703480b1022c.exe
	* Hash the exe using the md5deep and sha256deep utilities.
2. Google the hashes to see what comes up.
3. Go to [VirusTotal.com](http://VirusTotal.com) and either search for the hash or upload the .exe file. If you do the latter, click the “View last analysis” button. Both approaches should return the same results.

VirusTotal.com can run files and URLs through a barrage of antivirus programs to detect malicious files. This can be much more effective than checking a file against only one or two detection programs. However, people who create viruses also use this site to see whether their malicious code flags, and tweak the code until it comes up clean. It is not uncommon for a piece of malware to only turn up a few hits in VirusTotal.

**Question:**What does Ad-Aware report this hash to be?

1. Browse to the website [Reverse.it](http://Reverse.it) (now owned by hybrid-analysis), and click on “Select file,” and upload the WannaCry worm + loader executable (the .exe) or search for its hash. Reverse.it will check the hash of the file against previously analyzed files. Select any of the Note the wide range of information that this site provides about the executable. Examine the IP address that this malware attempt to access—see the “Network Analysis” section (accessible via the menu on the right-hand side), and then the “Contacted Hosts” section. Keep this IP address in the back of your mind somewhere.

**Question:**What IP address does the worm attempt to access?

Next we will manually perform some of the analyses that reverse.it performs, in order to get a hands-on feel for the work involved.

### Static analysis of WannaCry

Static Analysis is usually the first step in malware analysis. It involves gathering as much information as possible from a potential malware file and determining its functionality, purpose, and identifying traits to the greatest degree possible. Static analysis does not involve running the malware file, and thus is less risky than dynamic analysis. However, caution must be used here (and in all stages of analysis), as certain analysis tools may execute the malware without warning.

#### Using the Strings Program

As its name implies, strings is a command-line tool that will parse through a file and pull out any character strings in either ASCII or Unicode. The result of running Strings can tell you many things about potential malware:

* Error messages can give telling information about the functionality of a program.
* IP addresses that the program will call to.
* DLL files the program will try to access.

In this lab, we will use several tools from a set of tools developed in 1996 which still rocks called [Sysinternals](https://docs.microsoft.com/en-us/sysinternals/).

* Download strings from [here](https://docs.microsoft.com/en-us/sysinternals/downloads/strings) and extract the downloaded zip file.
* Copy-paste strings.exe into the C:\bin directory.
* For convenience, add the C:\bin directory to your cmd path which will let you run the commands in that directory without having to reference the entire path to their location. Run the following from a PowerShell terminal (windows search bar > “powershell”):
* [Environment]::SetEnvironmentVariable("Path", $env:Path + ";C:\bin", [EnvironmentVariableTarget]::User)
* Now run the strings program to see a list of options and commands. From a new cmd prompt opened after adding c:\bin to the path, confirm that you can run strings:
* strings

Look at its options.

* From a cmd prompt in the directory where 24d004a104d4d54034dbcffc2a4b19a11f39008a575aa614ea04703480b1022c.exe is located, run the following:
* strings -n 12 24d004a104d4d54034dbcffc2a4b19a11f39008a575aa614ea04703480b1022c.exe > output.txt
* notepad output.txt

The “-n” option means “minimum string length.”

You may recognize this strings output from the Reverse.It report.

**Note**: Make it easier on yourself and Use tab-completion to fill out that filename. Do not copy-paste.

Also, a convenient shortcut to opening a windows shell in a certain directory is to use Windows Explorer (the graphical file explorer thing) to navigate to where you want to go, then right-click in the white area in that directory listing, and choose “Open PowerShell window here”. PowerShell can do everything a cmd prompt can do, and more.

Search through output.txt (ctrl+f) for a http:// URL. This URL is the “killswitch” domain which, when registered, [blocked the WannaCry worm from spreading](https://www.wired.com/2017/05/accidental-kill-switch-slowed-fridays-massive-ransomware-attack/). See [the comic on this page](https://blog.comae.io/wannacry-new-variants-detected-b8908fefea7e) to understand in four simple comic panels the facepalm reason why that functionality is even in the worm in the first place.

**Question:**What is the killswitch domain for this worm?

#### Resource Hacker

Resource Hacker is another useful tool that can often pull information from a malware sample without executing the malware. Windows application development tools will set values in this section, but developers can override and set whatever information they want, if they so desire. Regardless, examining the values can sometimes give hints about and insight into the malware.

1. Download “Resource Hacker” from [here](http://www.angusj.com/resourcehacker/#download), and install it. Open “Resource Hacker” by searching for it from the windows search bar.
2. Load the WannaCry worm into Resource Hacker.
3. Navigate through the folder tree structure to see information about the malware.
	1. Examine the “Version Info” node, and look at the “FileVersion”. This can be helpful to track to note versions of malware that may be floating around.

**Question:**What FileVersion does this malware claim to be?

**Note**: Look at the CompanyName, FileDescription, and LegalCopyright and have a laugh.

**Question:**What is the name of the company that this malware claims is its author?

* 1. Applications can embed applications within their resource section, which is accessible via the navbar on the left under the R tree. The loader stores the WannaCry encrypter here. When the loader runs, if the killswitch is not triggered, it will extract the embedded resource payload, install it, and run it. Let’s extract it manually and play with it.
		1. Expand the “R” node and right-click the star range. Choose Save Resources to a BIN file (a “BIN” or binary file is another name for an .exe executable in Windows). Save it where you like. This is the wannacry encrypter. Rename it to be wannacry.exe for your convenience, if you feel like it.

#### Strings Again, This Time on the WannaCry Encrypter

From a cmd prompt in the directory where your newly extracted wannacry.exe is located, run the following:

strings -n 12 wannacry.exe > output.txt

notepad output.txt

Read through the output.txt file and look for sensical strings. Some malware can use “packing” tricks to make it harder for malware analysts to perform static analysis (read more about it [here](https://securingtomorrow.mcafee.com/business/malware-packers-use-tricks-avoid-analysis-detection/)). “Packing” is a form of compressing that is sometimes also combined with encryption. If this malware were “packed”, strings would find no sensical text inside. If it is not packed, you will see English words which are names of Windows or custom functions that the malware may call internally.

**Question:**Is this malware likely "packed", or is it "unpacked"?

**Consider:** If you did not know what this malware did, what would seeing calls in the strings output to windows functions such as "CryptDecrypt", "CryptEncrypt", "CryptDestroyKey", and "CryptImportKey" tell you?

Answer: you probably have ransomware on your hands.

#### Analyzing DLLs with Dependency Walker

A few Dynamic Linked Library (DLL, Windows function library) calls appeared when we ran the Strings program above. We can use another program, Dependency Walker, to obtain more information about those DLLs and their use by the malware. Dynamic linking is an area that gives a great deal of insight into how a program functions, and it is of particular importance to malware analysis.

1. Download Dependency Walker from <http://www.dependencywalker.com/>. Extract the zip file contents, and run depends.exe. When the program loads, use the “open” button on the top-left and load wanncry.exe.
2. There’s a lot of information here, so let’s step through it.
	* The top-left pane shows all of the DLL files that are called by the WannaCry program.
	* The top-right pane shows the functions that are called by the selected DLL.
	* The right-middle pane shows all possible functions that could be called by the selected DLL, along with their ordinal values; a function can either be called by its name or by the ordinal value, so if you see an ordinal value called you can use this list to check its functionality. Calling functions by ordinal allows a program to call the function without ever using its name in the code. It can be a useful obfuscation technique.
	* The bottom two panes show additional information.

Spend some time looking through the different DLLs.

**Question:**Under ADVAPI32.DLL, find BCRYPT.DLL. What is function for ordinal value 37, i.e., the 37th function from the top?

**Note**: Bcrypt.dll is a family of functions, documented and listed [here](https://docs.microsoft.com/en-us/windows/desktop/api/bcrypt/). Bcrypt.dll is not the same as the Bcrypt family of algorithms that we studied earlier in the semester. bcrypt.dll lamely refers to "bestcrypt", a.k.a "Cryptography: Next Generation (cng)" (see [here](https://stackoverflow.com/questions/9711568/does-winapis-bcrypt-h-actually-support-bcrypt-hashing)). It is the built-in Windows way to perform encryption. And yes, that means that WannaCry is lamely using Windows functions, instead of making their own functions, to encrypt everything.

### Dynamic analysis of WannaCry

Now we come to the fun part. Dynamic analysis is used to gather information about a file that could not be gathered during static analysis. Dynamic Analysis is inherently riskier than static analysis because it involves investigating malware as it runs, or the state of the host machine after the malware has executed.

The drawbacks of dynamic analysis are that (1) malware may behave differently in a VM environment if the malware detects that it is in such a setting; (2) malware may behave differently depending on available network and Internet connections, and (3) running malware may potentially expose the host or other hosts on a network to risk, as mentioned previously. However, if you follow the instructions in the document, you’ll be fine.

In your Windows VM, download these additional Sysinternals tools:

* [Process Monitor](https://docs.microsoft.com/en-us/sysinternals/downloads/procmon)
* [Process Explorer](https://docs.microsoft.com/en-us/sysinternals/downloads/process-explorer)

####  Process Monitor

You can get basic information from Task Manager in Windows, but Process Monitor allows you to track every action of a given process. Be warned that Process Manager generates a lot of data, all of which it stores in RAM. So, it may quickly fill up the memory of a VM and crash it if left to run for a long time.

No one uses Process Monitor without applying filters. We will apply filters soon to all the data pouring in.

**Final reminder:** Make sure that you have a snapshot of your Windows VM before proceeding!

1. Launch Process Monitor. From within Process Monitor, launch the Process Tree (Tools > Process Tree).
2. With Process Monitor running, run the24d004a104d4d54034dbcffc2a4b19a11f39008a575aa614ea04703480b1022c.exe. OH NO YOU ARE UNDER ATTACK. Or are you?
3. Stop Process Monitor from collecting (File > Capture Events to toggle). Examine the Process Tree. Note the entry for 24d004a104d4d54034dbcffc2a4b19a11f39008a575aa614ea04703480b1022c.exe. Right-click it and Add Process to Include Filter. Back in the main process monitor view, we are now looking at only events associated with our malware worm.
4. In the icon menu on the top of Process Monitor, deselect all icons in the far-right section except for the third from the right – the one with two computers and who-knows-what-shape on top of them. This will filter to show only network calls. This should at least one TCP Reconnect operation. This is the loader calling out to the killswitch domain. If the call resolves successfully, then the malware immediately exits without doing anything else.

**Question:**What was the value of the "Result" field for the calls to 104.17.41.137?

#### Disable Network Adapter and run dropper again

The worm won’t find the killswitch if we don’t have an internet connection! Before we proceed, disable the VM’s network adapters. To do this, from the VM’s top-bar menu, select “Devices” > “Network” > click “Connect Network Adapter” to disconnect it. Verify that you do not have Internet access from within the VM before proceeding.

Open Process Monitor again. Do the following from the menu options:

* Filter > Reset Filter
* Edit > Clear display
* File > Capture Events (to enable capturing)
1. With Process Monitor running, right-click 24d004a104d4d54034dbcffc2a4b19a11f39008a575aa614ea04703480b1022c.exe and choose “Run as Administrator.” OH NO YOU ARE UNDER ATTACK. YES, THIS TIME YOU REALLY ARE.
2. The loader installed the file stored in its R resource section into the C:\Windows\ directory, renamed it tasksche.exe, and then ran it. This is the WannaCry encrypter.

**Question:**What is the sha256 hash of C:\Windows\tasksche.exe?

1. What does this encrypter do? It scans through the entire drive for files with interesting file extensions, and it encrypts them, storing an encrypted version of the decryption key in header info in the file itself. The encrypted variants of all of your precious files have a .WNCRY extension.

strings in tasksche.exe shows that it is looking for files with the following extensions:

.der .pfx .key .crt .csr .p12 .pem .odt .ott .sxw .stw .uot .3ds .max .3dm .ods .ots .sxc .stc .dif .slk .wb2 .odp .otp .sxd

.std .uop .odg .otg .sxm .mml .lay .lay6 .asc .sqlite3 .sqlitedb .sql .accdb .mdb .db .dbf .odb .frm .myd .myi .ibd .mdf .ldf

.sln .suo .cs .cpp .pas .asm .js .cmd .bat .ps1 .vbs .vb .pl .dip .dch .sch .brd .jsp .php .asp .rb .java .jar .class .sh .mp3

.wav .swf .fla .wmv .mpg .vob .mpeg .asf .avi .mov .mp4 .3gp .mkv .3g2 .flv .wma .mid .m3u .m4u .djvu .svg .ai .psd .nef .tiff

.tif .cgm .raw .gif .png .bmp .jpg .jpeg .vcd .iso .backup .zip .rar .7z .gz .tgz .tar .bak .tbk .bz2 .PAQ .ARC .aes .gpg .vmx

.vmdk .vdi .sldm .sldx .sti .sxi .602 .hwp .snt .onetoc2 .dwg .pdf .wk1 .wks .123 .rtf .csv .txt .vsdx .vsd .edb .eml .msg .ost

.pst .potm .potx .ppam .ppsx .ppsm .pps .pot .pptm .pptx .ppt .xltm .xltx .xlc .xlm .xlt .xlw .xlsb .xlsm .xlsx .xls .dotx .dotm

.dot .docm .docb .docx .doc

It then pops up an annoying notification about your ransomed state which will continue to pop up every sixty seconds or so. Best way to deal with that popup is to not “X” it out, but rather, drag it to a corner of your screen, almost out of sight.

Right-click one of the encrypted files on your desktop, and “Edit in Notepad++” to examine the contents. What used to be plaintext isn’t so plain anymore, is it? The first 8 bytes of any file encrypted by WannaCry are always the same. This is WannaCry’s “magic number” You can see it in notepad++ or in a hex examiner such as HxD (installed on the VM).

**Question:**What is the magic number for a WannaCry-encrypted file?

#### Decrypting the WannaCrypted files

Just kidding. You can’t.

Actually, you can with two longshots.

1. A decryption tool exists which can extract the prime numbers from memory which were used in the encryption. However, it has [big caveats](https://blog.malwarebytes.com/cybercrime/2017/05/wannadecrypt-your-files/): it relies on a Windows encryption implementation bug that only works for Windows OS versions less than 10, and “it relies on current running memory so once you reboot it will be gone and if you’ve done too much on the system since infection, it’s possible the key won’t be found (because it’s been overwritten by data from other applications using the same memory space).”
2. Also, there’s a chance that if you pay the ransom, you will get a response… [but](https://www.pcworld.com/article/3196880/security/paying-the-wannacry-ransom-will-probably-get-you-nothing-heres-why.html), “Those who do shouldn’t expect a quick response – or any response at all. Even after payment, the ransomware doesn’t automatically release your computer and decrypt your files, according to security researchers. Instead, victims have to wait and hope WannaCry’s developers will remotely free the hostage computer over the internet. It’s a process that’s entirely manual and contains a serious flaw: The hackers have no way to prove who paid off the ransom.”

**Best defense against ransomware?** Backups. Do you have one?

#### Bonus Reading!

In September 2018, the US Justice Department [indicted a North Korean agent who participated in the WannaCry and Sony hacks](https://arstechnica.com/information-technology/2018/09/us-indicts-north-korean-agents-for-wannacry-sony-attacks/).

**Question:**What is the name of the indicted agent?

### References

* <https://www.fireeye.com/blog/threat-research/2017/05/wannacry-malware-profile.html>
* <https://www.malware-traffic-analysis.net/2017/05/18/index2.html>

## **Learn more**

There are several excellent published malware analysis reports about WannaCry. If you want to learn more about malware analysis, compare your results to those in these reports:



Source: <https://www.endgame.com/blog/technical-blog/wcrywanacry-ransomware-technical-analysis>

* EndGame: <https://www.endgame.com/blog/technical-blog/wcrywanacry-ransomware-technical-analysis>
* FireEye: <https://www.fireeye.com/blog/threat-research/2017/05/wannacry-malware-profile.html>
* SecureWorks: <https://www.secureworks.com/research/wcry-ransomware-analysis>
* McAfee: <https://securingtomorrow.mcafee.com/mcafee-labs/analysis-wannacry-ransomware/>

## **Read more**

* NY Magazine profile of Marcus Hutchins, the malware analyst who stopped WannaCry: <http://nymag.com/selectall/2018/03/marcus-hutchins-hacker.html>
* Wikipedia: <https://en.wikipedia.org/wiki/WannaCry_ransomware_attack>
* NY Times: <https://www.nytimes.com/2017/05/12/world/europe/uk-national-health-service-cyberattack.html>

|  |  |
| --- | --- |
| Macintosh HD:Users:anthony:Desktop:Unknown-1.jpeg | *Countdown to Zero Day: Stuxnet and the Launch of the World's First Digital Weapon*, by Kim Zetter. Interesting and compelling read about the discovery of Stuxnet and how it changed the world. The primary source material for the film *Zero Days*.Available at [Temple Library](https://librarysearch.temple.edu/catalog/991020007999703811) Amazon: <http://amzn.com/077043617X> |
|  | *Practical Malware Analysis: A Hands-On Guide to Dissecting Malicious Software*, by Michael Sikorski, Andrew Honig<http://a.co/iccEPbT>  |
| /var/folders/25/h44_1jkn0pl51_ms9gt590xm0000gn/T/com.microsoft.Word/WebArchiveCopyPasteTempFiles/51TwRHpVJnL._SX396_BO1,204,203,200_.jpg | *Malware Analyst's Cookbook and DVD: Tools and Techniques for Fighting Malicious Code*, by Michael Ligh, Steven Adair, Blake Hartstein, Matthew Richard<http://a.co/aTkDRgK>  |