Project title: Resolution of **NotPetya** Ransomware Attack on a Multinational Shipping and logistics Company – A case study of software supply chain business process

Project Objective: Carryout a detailed analysis of the NotPetya Ransomware attack. Goal is to examine the Tactics, Techniques & Procedures (TTPs) used by the threat actors, identify potential threat groups and recommend effective mitigation strategies to prevent similar incident in the future.

Executive Summary

- **Incident ID:** CSA-SOC-APR-2025–02
- **Incident Severity:** Critical (Threat actor successfully accessed the corporate network with elevated privilege and destroyed them)
- **Incident Status:** Systems and business restorations/recoveries completed

NotPetya Ransomware was an advancement of Petya with more sophistication for greater devastating impact on the victims. While Petya was a traditional ransomware for financial gains, NotPetya was a wiper and a destroyer. It is perceived to be state sponsored attack by Russian military against Ukraine with destructive intent.

NotPetya group **(ID G0034 - Sandworm Team)** capitalized and exploited the vulnerabilities in Server Message Block version-1 (SMBv1) that was still in use by their victims which was outdated and obsolete. Used Mimikatz to steal credential from victims' windows systems and used PSExec tool of windows to carry out remote code execution (RCE) and executed fileless ransomware on the victim's systems. With the combination of Mimikatz and PSExec tools, they were able to bypass the victim's security monitoring systems and achieved obfuscation effortlessly and exploited EternalBlue SMBv1 vulnerability script – **smb-ms17-010**. They successfully corrupted the MBRs/MFTs on the victims systems making it impossible for recovery. Rebooting of systems for recovery failed as they were just left with the bare metals. Victims only recovered by falling back on existing offline backups and in some cases through clean installs of the systems.

Server Message Blocks (SMB) is a client-server service for files and printer sharing over the network. It runs on tcp ports 139 for Linux and 445 for windows systems respectively.

It will be interesting to know that SMBv1 has been outdated and now replaced with SMBv2 & SMBv3 now.

The Vulnerabilities of SMBv1 Include;

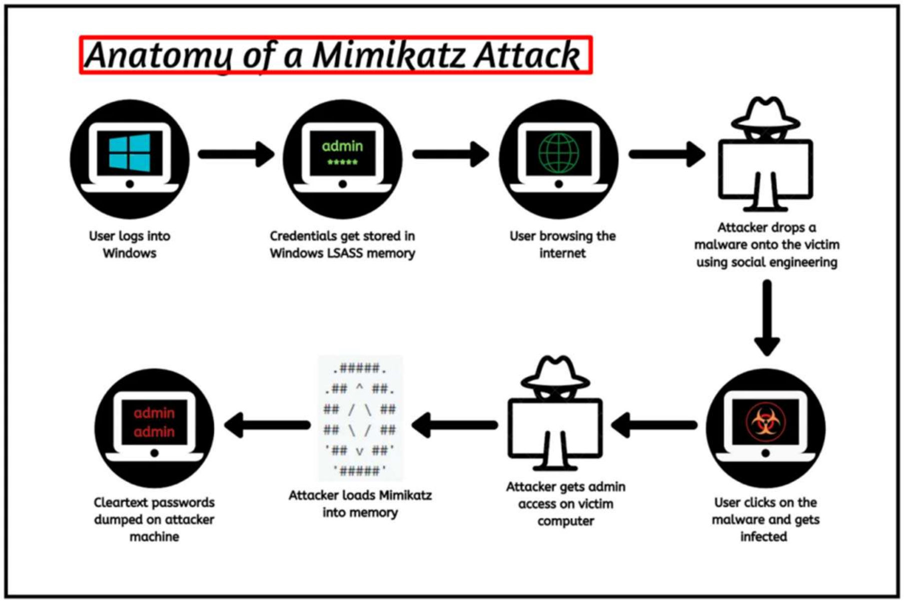
- It does not support MFA for authentication
- The password hashing does not provide for salting making it easy to crack using hashing tools – NTLM
- It uses NTLM and Kerberos for windows credentials management. Kerberos has now replaced NTLM to improve security of credentials. NotPetya sample was analyzed using Software Reverse Engineering (SRE) tools and techniques in Ghidra, PEStudio and Virustotal for more insight. It was interesting to discover that majority of the function codes in the Ransomware were written to two critical Windows Operating systems libraries – kernel32.dll & shell32.dll. Below are their roles in OS operations;

- **KERNEL32.DLL:** The Kernel32.dll file is an essential component of the Windows operating system. It is a dynamic link library file that contains various functions and resources required for the proper functioning of the Windows kernel. The kernel is the core part of the operating system that manages system resources, such as memory, processes, and input/output operations. It provides a set of functions that allow applications to interact with the operating system. These functions include memory management, process creation and termination, file input/output, and error handling. In simpler terms, Kernel32.dll acts as a bridge between the software applications and the operating system, enabling them to communicate and perform tasks.

- **SHELL32.DLL:** "Shell32.dll" is an essential component of the Microsoft Windows Operating System. It is a Dynamic Link Library (DLL) providing many functions of the Windows Shell, the graphical user interface (GUI) for Windows that includes the desktop, Start Menu, Autoplay, and Taskbar, and in some versions also Flip3D and the charms. "Shell32.dll" is especially needed to open web pages and files. It is a Component Object Model (COM) "server", usable by managed (.NET) or native code, residing in "C:\Windows\System32. It should not and essentially cannot be removed: if anything other than Windows Update or TrustedInstaller deletes or replaces it, Windows Resource Protection (WRP) will silently restore it from a system cache.

- **EternalBlue vulnerability:** It is an exploit that allows cyber threat actors to remotely execute arbitrary code and gain access to a network by sending specially crafted packets. It exploits a software vulnerability in Microsoft's Windows operating systems (OS) Server Message Block (SMB) version 1 (SMBv1) protocol, a network file and printer sharing protocol that allows access to files on a remote server. This exploit potentially allows cyber threat actors to compromise the entire network and all devices connected to it. Due to EternalBlue's ability to compromise networks, if one device is infected by malware via EternalBlue, every device connected to the network is at risk. This makes recovery difficult, as all devices on a network may have to be taken offline for remediation. This vulnerability was patched and is listed on Microsoft's security bulletin as MS17-010

This is how the group was able to encrypt / corrupt the MBRs/MFTs of the windows OS and prevented systems restart.

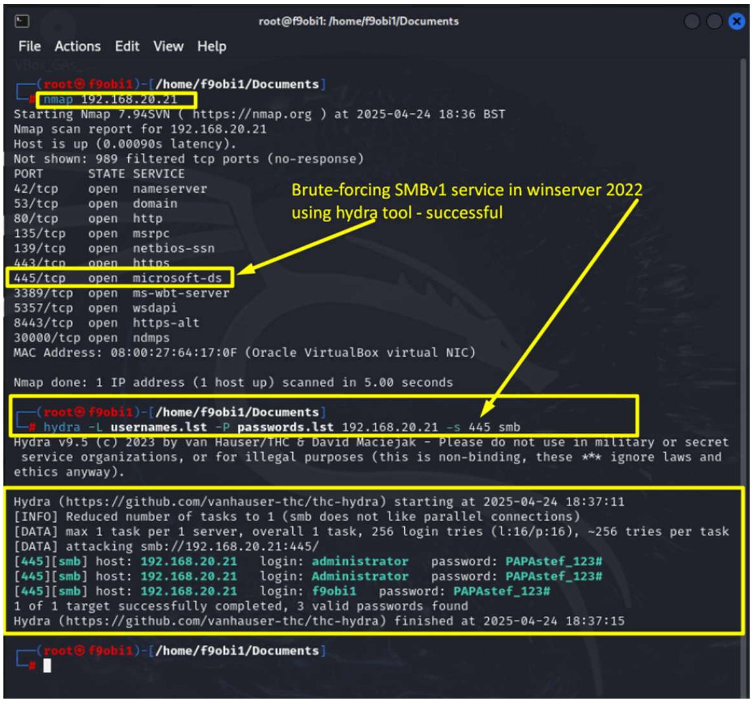

Anatomy of a Mimikatz Attack
admin
*****
User logs into Windows
Credentials get stored in Windows LSASS memory
User browsing the internet
Attacker drops a malware onto the victim using social engineering
User clicks on the malware and gets infected
Attacker gets admin access on victim computer
Attacker loads Mimikatz into memory
admin
admin
Cleartext passwords dumped on attacker machine

Project Execution Steps

Network / SOC setup for simulations

- Installed Virtualbox software on a host computer that has type-2 Hypervisor built-in
- Installed the following virtual machines (VMs) on the Virtualbox
  - PFsense software firewall VM
  - Wazuh server manager
  - Kali Linux, Ubuntu Linux desktop, Windows 10 pro and window server 2022 machines
- **Tools used:** Nmap, Ghidra, PEStudio, SSH & SMB services, Enum4linux & MITRE ATT&CK Framework. Also simulated SMB brute-forcing attempts using Hydra
- Simulated Nmap & SMB enumerations on windows and Linux systems
- Also simulated SMB brute-forcing as part of enumeration

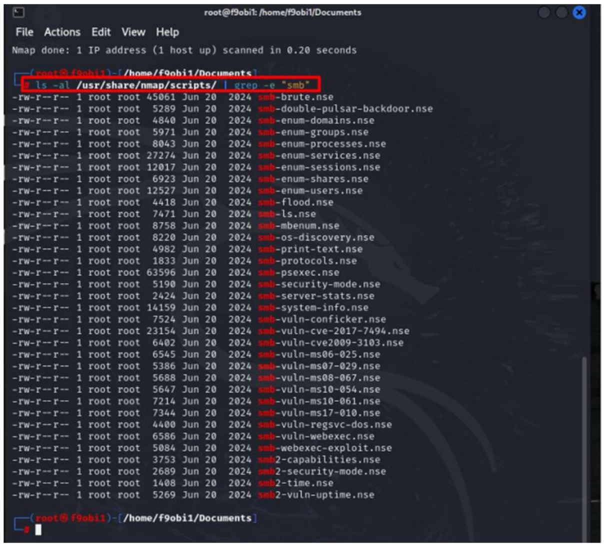
SMB Brute-force using Hydra


root@f9obi1: /home/f9obi1/Documents
File Actions Edit View Help

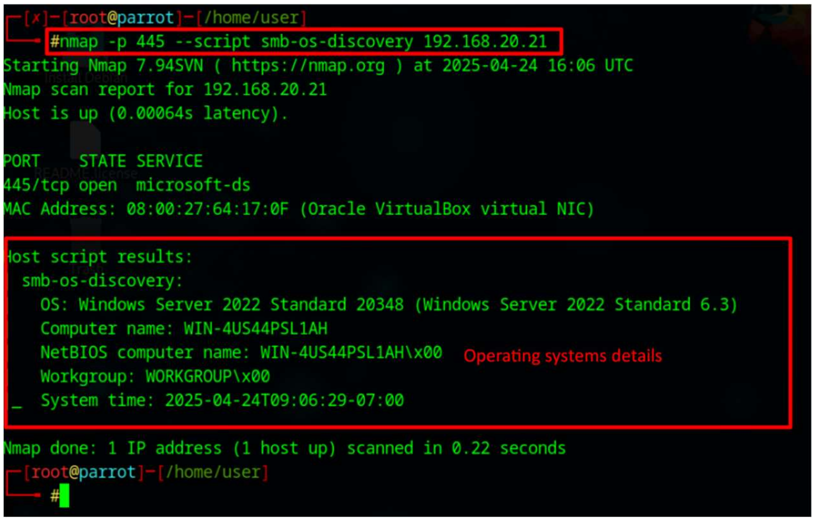
Brute-forcing SMBv1 service in winserver 2022 using hydra tool - successful

```
┌──(root㉿f9obi1)-[/home/f9obi1/Documents]
└─# nmap 192.168.20.21
Starting Nmap 7.94SVN ( https://nmap.org ) at 2025-04-24 18:36 BST
Nmap scan report for 192.168.20.21
Host is up (0.00090s latency).
Not shown: 989 filtered tcp ports (no-response)
PORT      STATE SERVICE
42/tcp    open  nameserver
53/tcp    open  domain
80/tcp    open  http
135/tcp   open  msrpc
139/tcp   open  netbios-ssn
443/tcp   open  https
445/tcp   open  microsoft-ds
3389/tcp  open  ms-wbt-server
5357/tcp  open  wsdapi
8443/tcp  open  https-alt
30000/tcp open  ndmps
MAC Address: 08:00:27:64:17:0F (Oracle VirtualBox virtual NIC)

Nmap done: 1 IP address (1 host up) scanned in 5.00 seconds

┌──(root㉿f9obi1)-[/home/f9obi1/Documents]
└─# hydra -L usernames.lst -P passwords.lst 192.168.20.21 -s 445 smb
Hydra v9.5 (c) 2023 by van Hauser/THC & David Maciejak - Please do not use in military or secret
 service organizations, or for illegal purposes (this is non-binding, these *** ignore laws and
ethics anyway).

Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2025-04-24 18:37:11
[INFO] Reduced number of tasks to 1 (smb does not like parallel connections)
[DATA] max 1 task per 1 server, overall 1 task, 256 login tries (l:16/p:16), ~256 tries per task
[DATA] attacking smb://192.168.20.21:445/
[445][smb] host: 192.168.20.21   login: administrator   password: PAPAstef_123#
[445][smb] host: 192.168.20.21   login: Administrator   password: PAPAstef_123#
[445][smb] host: 192.168.20.21   login: f9obi1   password: PAPAstef_123#
1 of 1 target successfully completed, 3 valid passwords found
Hydra (https://github.com/vanhauser-thc/thc-hydra) finished at 2025-04-24 18:37:15

┌──(root㉿f9obi1)-[/home/f9obi1/Documents]
└─# 
```

SMB Scripts Enumeration using Nmap

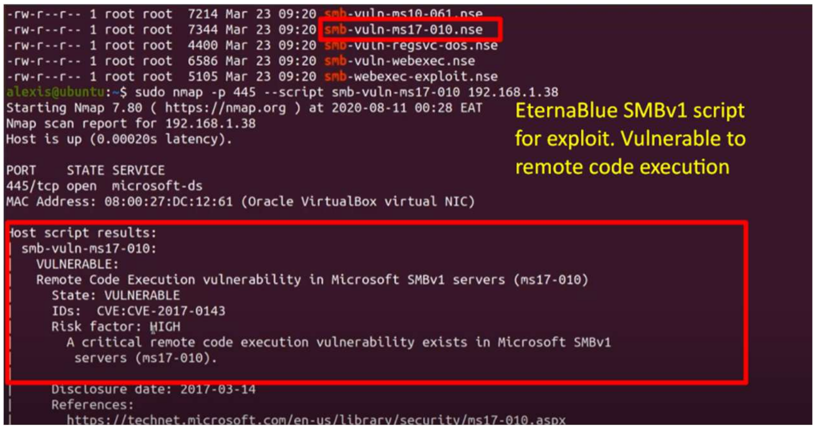

root@f9obi1: /home/f9obi1/Documents
File Actions Edit View Help

```
Nmap done: 1 IP address (1 host up) scanned in 0.20 seconds

┌──(root㉿f9obi1)-[/home/f9obi1/Documents]
└─# ls -al /usr/share/nmap/scripts/ | grep -e "smb"
-rw-r--r-- 1 root root 45061 Jun 20  2024 smb-brute.nse
-rw-r--r-- 1 root root  5289 Jun 20  2024 smb-double-pulsar-backdoor.nse
-rw-r--r-- 1 root root  4840 Jun 20  2024 smb-enum-domains.nse
-rw-r--r-- 1 root root  5971 Jun 20  2024 smb-enum-groups.nse
-rw-r--r-- 1 root root  8043 Jun 20  2024 smb-enum-processes.nse
-rw-r--r-- 1 root root 27274 Jun 20  2024 smb-enum-services.nse
-rw-r--r-- 1 root root 12017 Jun 20  2024 smb-enum-sessions.nse
-rw-r--r-- 1 root root  6923 Jun 20  2024 smb-enum-shares.nse
-rw-r--r-- 1 root root 12527 Jun 20  2024 smb-enum-users.nse
-rw-r--r-- 1 root root  4418 Jun 20  2024 smb-flood.nse
-rw-r--r-- 1 root root  7471 Jun 20  2024 smb-ls.nse
-rw-r--r-- 1 root root  8758 Jun 20  2024 smb-mbenum.nse
-rw-r--r-- 1 root root  8220 Jun 20  2024 smb-os-discovery.nse
-rw-r--r-- 1 root root  4982 Jun 20  2024 smb-print-text.nse
-rw-r--r-- 1 root root  1833 Jun 20  2024 smb-protocols.nse
-rw-r--r-- 1 root root 63596 Jun 20  2024 smb-psexec.nse
-rw-r--r-- 1 root root  5190 Jun 20  2024 smb-security-mode.nse
-rw-r--r-- 1 root root  2424 Jun 20  2024 smb-server-stats.nse
-rw-r--r-- 1 root root 14159 Jun 20  2024 smb-system-info.nse
-rw-r--r-- 1 root root  7524 Jun 20  2024 smb-vuln-conficker.nse
-rw-r--r-- 1 root root 23154 Jun 20  2024 smb-vuln-cve-2017-7494.nse
-rw-r--r-- 1 root root  6402 Jun 20  2024 smb-vuln-cve2009-3103.nse
-rw-r--r-- 1 root root  6545 Jun 20  2024 smb-vuln-ms06-025.nse
-rw-r--r-- 1 root root  5386 Jun 20  2024 smb-vuln-ms07-029.nse
-rw-r--r-- 1 root root  5688 Jun 20  2024 smb-vuln-ms08-067.nse
-rw-r--r-- 1 root root  5647 Jun 20  2024 smb-vuln-ms10-054.nse
-rw-r--r-- 1 root root  7214 Jun 20  2024 smb-vuln-ms10-061.nse
-rw-r--r-- 1 root root  7344 Jun 20  2024 smb-vuln-ms17-010.nse
-rw-r--r-- 1 root root  4400 Jun 20  2024 smb-vuln-regsvc-dos.nse
-rw-r--r-- 1 root root  6586 Jun 20  2024 smb-vuln-webexec.nse
-rw-r--r-- 1 root root  5084 Jun 20  2024 smb-webexec-exploit.nse
-rw-r--r-- 1 root root  3753 Jun 20  2024 smb2-capabilities.nse
-rw-r--r-- 1 root root  2689 Jun 20  2024 smb2-security-mode.nse
-rw-r--r-- 1 root root  1408 Jun 20  2024 smb2-time.nse
-rw-r--r-- 1 root root  5269 Jun 20  2024 smb2-vuln-uptime.nse

┌──(root㉿f9obi1)-[/home/f9obi1/Documents]
└─#
```



```
┌─[✗]─[root@parrot]─[/home/user]
└──╼ #nmap -p 445 --script smb-os-discovery 192.168.20.21
Starting Nmap 7.94SVN ( https://nmap.org ) at 2025-04-24 16:06 UTC
Nmap scan report for 192.168.20.21
Host is up (0.00064s latency).

PORT    STATE SERVICE
445/tcp open  microsoft-ds
MAC Address: 08:00:27:64:17:0F (Oracle VirtualBox virtual NIC)

Host script results:
| smb-os-discovery:
|   OS: Windows Server 2022 Standard 20348 (Windows Server 2022 Standard 6.3)
|   Computer name: WIN-4US44PSL1AH
|   NetBIOS computer name: WIN-4US44PSL1AH\x00
|   Workgroup: WORKGROUP\x00
|_  System time: 2025-04-24T09:06:29-07:00

Nmap done: 1 IP address (1 host up) scanned in 0.22 seconds
┌─[root@parrot]─[/home/user]
└──╼ #
```

Operating systems details



```
-rw-r--r-- 1 root root  7214 Mar 23 09:20 smb-vuln-ms10-061.nse
-rw-r--r-- 1 root root  7344 Mar 23 09:20 smb-vuln-ms17-010.nse
-rw-r--r-- 1 root root  4400 Mar 23 09:20 smb-vuln-regsvc-dos.nse
-rw-r--r-- 1 root root  6586 Mar 23 09:20 smb-vuln-webexec.nse
-rw-r--r-- 1 root root  5105 Mar 23 09:20 smb-webexec-exploit.nse
alexis@ubuntu:~$ sudo nmap -p 445 --script smb-vuln-ms17-010 192.168.1.38
Starting Nmap 7.80 ( https://nmap.org ) at 2020-08-11 00:28 EAT
Nmap scan report for 192.168.1.38
Host is up (0.00020s latency).

PORT    STATE SERVICE
445/tcp open  microsoft-ds
MAC Address: 08:00:27:DC:12:61 (Oracle VirtualBox virtual NIC)

Host script results:
| smb-vuln-ms17-010:
|   VULNERABLE:
|   Remote Code Execution vulnerability in Microsoft SMBv1 servers (ms17-010)
|     State: VULNERABLE
|     IDs:  CVE:CVE-2017-0143
|     Risk factor: HIGH
|       A critical remote code execution vulnerability exists in Microsoft SMBv1
|        servers (ms17-010).
|
|     Disclosure date: 2017-03-14
|     References:
|       https://technet.microsoft.com/en-us/library/security/ms17-010.aspx
```

EternaBlue SMBv1 script for exploit. Vulnerable to remote code execution

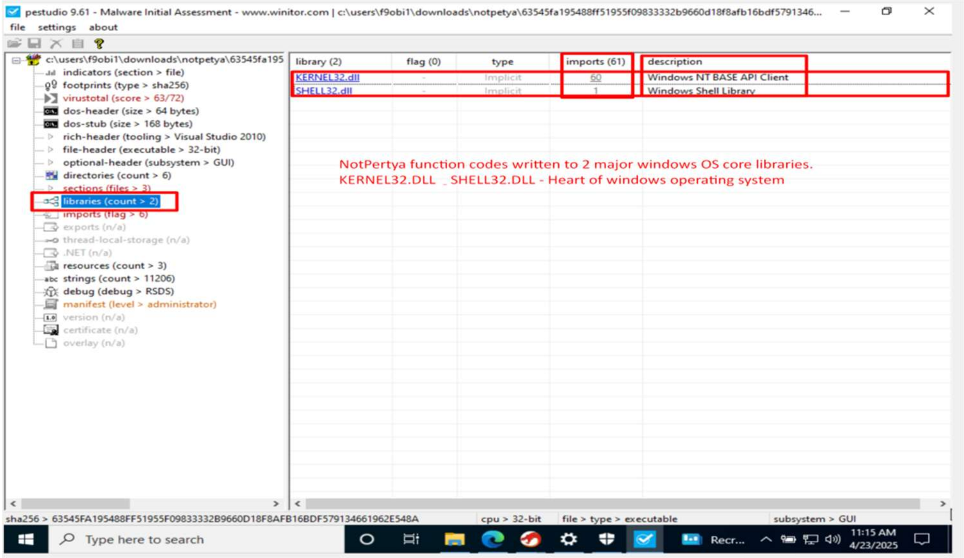
TTPs analysis using MITRE ATT&CK Framework

The NotPetya threat Actors used combination of TTPs to execution their attacks. They primarily used combination of EternaBlue Exploit, PSExec and Mimikatz to strengthen their sophistication. Some of them are shown below

T1210 - Exploit of remote services: They exploited the vulnerabilities of SMBv1 (out dated service) to spread rapidly across networks. This technique made it easy for them to maintain lateral movement across networks and infect systems

- **T1083 – File and directory discovery:** The ransomware searched for specific files with specific extensions to identify and encrypt target data
- **T1004 - Credential Access:** Used Mimikatz and PSExec tools to steal credentials and impersonate user tokens. This enabled further lateral and persistence. This is particular with WINLogon Helper DLL credentials to execute codes with system privileges
- **T1021 – Lateral movement:** Combination of SMBv1 exploitation and credential theft help the attackers to maintain lateral movement across systems and networks. This exploitation of remote services like SSH/RDP/SMB network services
- **T1486 – Data Encryption / System for impact:** The Ransomware used 2048-bit RSA encryption to encrypt user files and system components. It encrypted Master Boot Records MBRs & Master Files Tables MFTs of target systems making them irrecoverable
- **T1562/T1566 – Defense evasion/Phishing techniques:** It was able to evade company's security monitoring tools and systems and obfuscated its activities. Also employed Spear phishing technique for privileged accounts information gathering for elevated access
- **T1573 – Persistence:** It was also able to create scheduled tasks to re-encrypts systems even after attempts to recover. This process ensured long term impact and prolonged business downtimes for affected companies. This related to advanced persistence threats **(APT)** technique

PEStudio – Static Analysis Simulation



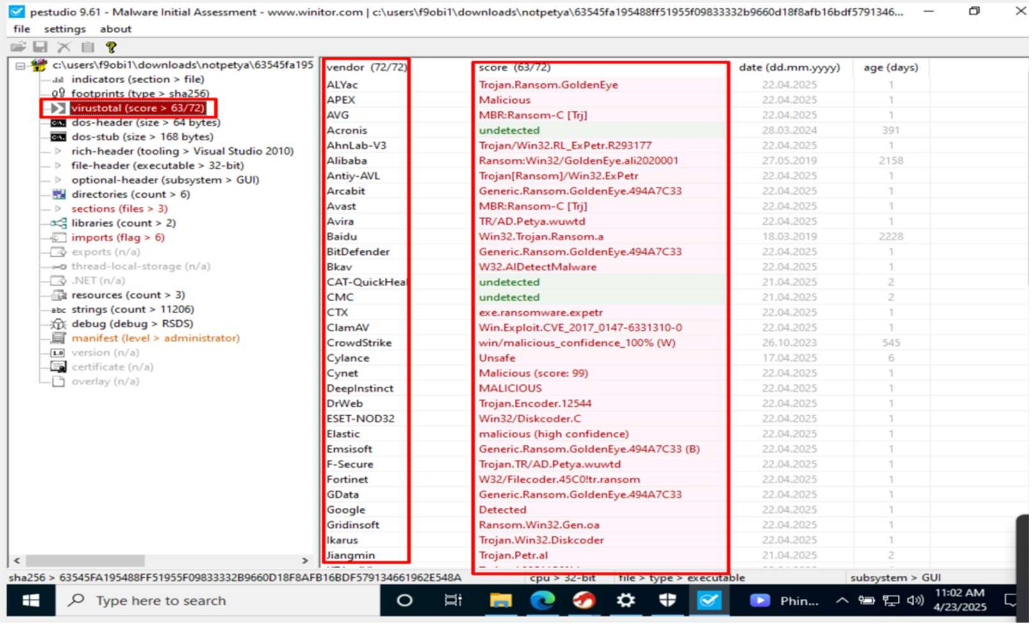
pestudio 9.61 - Malware Initial Assessment - www.winitor.com | c:\users\f9obi1\downloads\notpetya\63545fa195488ff51955f09833332b9660d18f8afb16bdf5791346...

c:\users\f9obi1\downloads\notpetya\63545fa195
- indicators (section > file)
- footprints (type > sha256)
- virustotal (score > 63/72)
- dos-header (size > 64 bytes)
- dos-stub (size > 168 bytes)
- rich-header (tooling > Visual Studio 2010)
- file-header (executable > 32-bit)
- optional-header (subsystem > GUI)
- directories (count > 6)
- sections (files > 3)
- libraries (count > 2)
- imports (flag > 6)
- exports (n/a)
- thread-local-storage (n/a)
- .NET (n/a)
- resources (count > 3)
- strings (count > 11206)
- debug (debug > RSDS)
- manifest (level > administrator)
- version (n/a)
- certificate (n/a)
- overlay (n/a)

library (2)	flag (0)	type	imports (61)	description
KERNEL32.dll	-	Implicit	60	Windows NT BASE API Client
SHELL32.dll	-	Implicit	1	Windows Shell Library

NotPertya function codes written to 2 major windows OS core libraries.
KERNEL32.DLL _ SHELL32.DLL - Heart of windows operating system

sha256 > 63545FA195488FF51955F09833332B9660D18F8AFB16BDF579134661962E548A | cpu > 32-bit | file > type > executable | subsystem > GUI



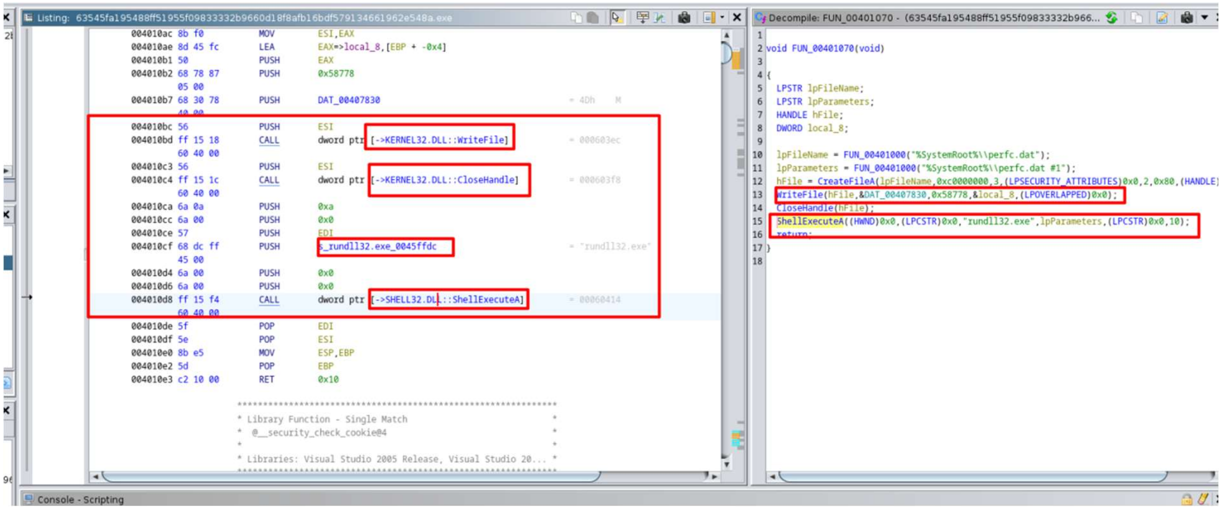
pestudio 9.61 - Malware Initial Assessment - www.winitor.com | c:\users\f9obi1\downloads\notpetya\63545fa195488ff51955f09833332b9660d18f8afb16bdf5791346...

```
c:\users\f9obi1\downloads\notpetya\63545fa195
    indicators (section > file)
    footprints (type > sha256)
    virustotal (score > 63/72)
    dos-header (size > 64 bytes)
    dos-stub (size > 168 bytes)
    rich-header (tooling > Visual Studio 2010)
    file-header (executable > 32-bit)
    optional-header (subsystem > GUI)
    directories (count > 6)
    sections (files > 3)
    libraries (count > 2)
    imports (flag > 6)
    exports (n/a)
    thread-local-storage (n/a)
    .NET (n/a)
    resources (count > 3)
    strings (count > 11206)
    debug (debug > RSDS)
    manifest (level > administrator)
    version (n/a)
    certificate (n/a)
    overlay (n/a)
```

vendor (72/72)	score (63/72)	date (dd.mm.yyyy)	age (days)
ALYac	Trojan.Ransom.GoldenEye	22.04.2025	1
APEX	Malicious	22.04.2025	1
AVG	MBR:Ransom-C [Trj]	22.04.2025	1
Acronis	undetected	28.03.2024	391
AhnLab-V3	Trojan/Win32.RL_ExPetr.R293177	22.04.2025	1
Alibaba	Ransom:Win32/GoldenEye.ali2020001	27.05.2019	2158
Antiy-AVL	Trojan[Ransom]/Win32.ExPetr	22.04.2025	1
Arcabit	Generic.Ransom.GoldenEye.494A7C33	22.04.2025	1
Avast	MBR:Ransom-C [Trj]	22.04.2025	1
Avira	TR/AD.Petya.wuwtd	22.04.2025	1
Baidu	Win32.Trojan.Ransom.a	18.03.2019	2228
BitDefender	Generic.Ransom.GoldenEye.494A7C33	22.04.2025	1
Bkav	W32.AIDetectMalware	22.04.2025	1
CAT-QuickHeal	undetected	21.04.2025	2
CMC	undetected	21.04.2025	2
CTX	exe.ransomware.expetr	22.04.2025	1
ClamAV	Win.Exploit.CVE_2017_0147-6331310-0	22.04.2025	1
CrowdStrike	win/malicious_confidence_100% (W)	26.10.2023	545
Cylance	Unsafe	17.04.2025	6
Cynet	Malicious (score: 99)	22.04.2025	1
DeepInstinct	MALICIOUS	22.04.2025	1
DrWeb	Trojan.Encoder.12544	22.04.2025	1
ESET-NOD32	Win32/Diskcoder.C	22.04.2025	1
Elastic	malicious (high confidence)	22.04.2025	1
Emsisoft	Generic.Ransom.GoldenEye.494A7C33 (B)	22.04.2025	1
F-Secure	Trojan.TR/AD.Petya.wuwtd	22.04.2025	1
Fortinet	W32/Filecoder.45C0!tr.ransom	22.04.2025	1
GData	Generic.Ransom.GoldenEye.494A7C33	22.04.2025	1
Google	Detected	22.04.2025	1
Gridinsoft	Ransom.Win32.Gen.oa	22.04.2025	1
Ikarus	Trojan.Win32.Diskcoder	22.04.2025	1
Jiangmin	Trojan.Petr.al	21.04.2025	2

sha256 > 63545FA195488FF51955F09833332B9660D18F8AFB16BDF579134661962E548A | cpu > 32-bit | file > type > executable | subsystem > GUI

Ghidra – Static Analysis Simulation

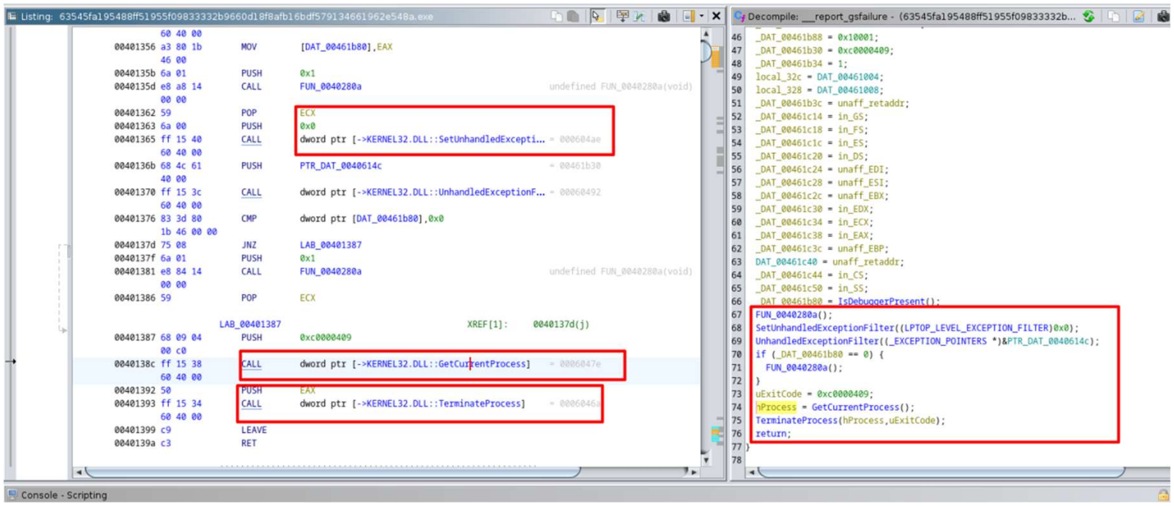


Listing: 63545fa195488ff51955f09833332b9660d18f8afb16bdf579134661962e548a.exe

```
004010ac 8b f0        MOV     ESI,EAX
004010ae 8d 45 fc     LEA     EAX=>local_8,[EBP + -0x4]
004010b1 50           PUSH    EAX
004010b2 68 78 87     PUSH    0x58778
         05 00
004010b7 68 30 78     PUSH    DAT_00407830                          = 4Dh    M
         40 00
004010bc 56           PUSH    ESI
004010bd ff 15 18     CALL    dword ptr [->KERNEL32.DLL::WriteFile]      = 000603ec
         60 40 00
004010c3 56           PUSH    ESI
004010c4 ff 15 1c     CALL    dword ptr [->KERNEL32.DLL::CloseHandle]    = 000603f8
         60 40 00
004010ca 6a 0a        PUSH    0xa
004010cc 6a 00        PUSH    0x0
004010ce 57           PUSH    EDI
004010cf 68 dc ff     PUSH    s_rundll32.exe_0045ffdc               = "rundll32.exe"
         45 00
004010d4 6a 00        PUSH    0x0
004010d6 6a 00        PUSH    0x0
004010d8 ff 15 f4     CALL    dword ptr [->SHELL32.DLL::ShellExecuteA]   = 00060414
         60 40 00
004010de 5f           POP     EDI
004010df 5e           POP     ESI
004010e0 8b e5        MOV     ESP,EBP
004010e2 5d           POP     EBP
004010e3 c2 10 00     RET     0x10

              **************************************************************
              * Library Function - Single Match                            *
              *  @__security_check_cookie@4                                *
              *                                                            *
              * Libraries: Visual Studio 2005 Release, Visual Studio 20... *
              **************************************************************
```

Decompile: FUN_00401070 - (63545fa195488ff51955f09833332b966...

```

void FUN_00401070(void)

{
  LPSTR lpFileName;
  LPSTR lpParameters;
  HANDLE hFile;
  DWORD local_8;

  lpFileName = FUN_00401000("%SystemRoot%\\perfc.dat");
  lpParameters = FUN_00401000("%SystemRoot%\\perfc.dat #1");
  hFile = CreateFileA(lpFileName,0xc0000000,3,(LPSECURITY_ATTRIBUTES)0x0,2,0x80,(HANDLE)
  WriteFile(hFile,&DAT_00407830,0x58778,&local_8,(LPOVERLAPPED)0x0);
  CloseHandle(hFile);
  ShellExecuteA((HWND)0x0,(LPCSTR)0x0,"rundll32.exe",lpParameters,(LPCSTR)0x0,10);
  return;
}

```

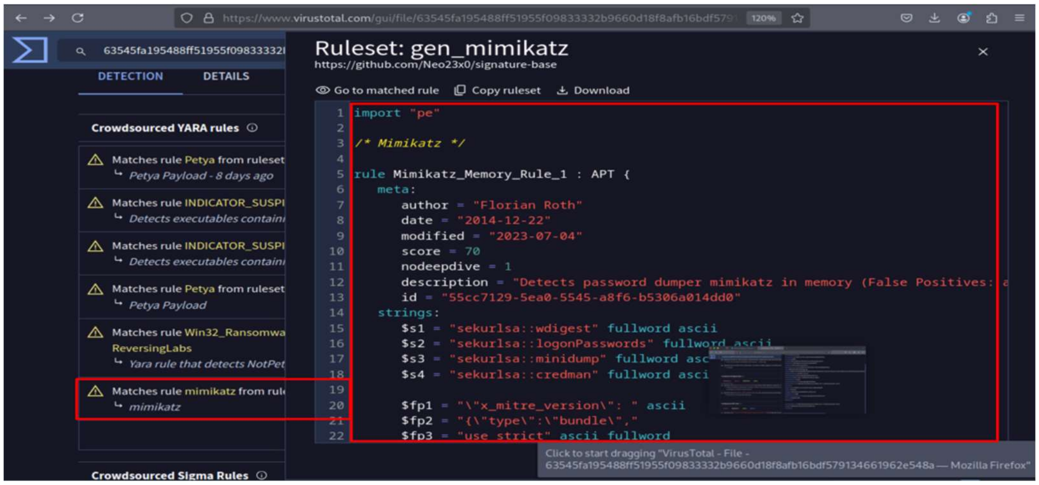


Listing: 63545fa195488ff51955f09833332b9660d18f8afb16bdf579134661962e548a.exe
Decompile: __report_gsfailure - (63545fa195488ff51955f09833332b...
Console - Scripting

```
                 60 40 00
        00401356 a3 80 1b        MOV        [DAT_00461b80],EAX
                 46 00
        0040135b 6a 01           PUSH       0x1
        0040135d e8 a8 14        CALL       FUN_0040280a                       undefined FUN_0040280a(void)
                 00 00
        00401362 59              POP        ECX
        00401363 6a 00           PUSH       0x0
        00401365 ff 15 40        CALL       dword ptr [->KERNEL32.DLL::SetUnhandledExcepti... = 000604ae
                 60 40 00
        0040136b 68 4c 61        PUSH       PTR_DAT_0040614c                   = 00461b30
                 40 00
        00401370 ff 15 3c        CALL       dword ptr [->KERNEL32.DLL::UnhandledExceptionF... = 00060492
                 60 40 00
        00401376 83 3d 80        CMP        dword ptr [DAT_00461b80],0x0
                 1b 46 00 00
        0040137d 75 08           JNZ        LAB_00401387
        0040137f 6a 01           PUSH       0x1
        00401381 e8 84 14        CALL       FUN_0040280a                       undefined FUN_0040280a(void)
                 00 00
        00401386 59              POP        ECX

                            LAB_00401387                    XREF[1]:     0040137d(j)
        00401387 68 09 04        PUSH       0xc0000409
                 00 c0
        0040138c ff 15 38        CALL       dword ptr [->KERNEL32.DLL::GetCurrentProcess]  = 0006047e
                 60 40 00
        00401392 50              PUSH       EAX
        00401393 ff 15 34        CALL       dword ptr [->KERNEL32.DLL::TerminateProcess]   = 0006046a
                 60 40 00
        00401399 c9              LEAVE
        0040139a c3              RET
```

```
  _DAT_00461b88 = 0x10001;
  _DAT_00461b30 = 0xc0000409;
  _DAT_00461b34 = 1;
  local_32c = DAT_00461004;
  local_328 = DAT_00461008;
  _DAT_00461b3c = unaff_retaddr;
  _DAT_00461c14 = in_GS;
  _DAT_00461c18 = in_FS;
  _DAT_00461c1c = in_ES;
  _DAT_00461c20 = in_DS;
  _DAT_00461c24 = unaff_EDI;
  _DAT_00461c28 = unaff_ESI;
  _DAT_00461c2c = unaff_EBX;
  _DAT_00461c30 = in_EDX;
  _DAT_00461c34 = in_ECX;
  _DAT_00461c38 = in_EAX;
  _DAT_00461c3c = unaff_EBP;
  DAT_00461c40 = unaff_retaddr;
  _DAT_00461c44 = in_CS;
  _DAT_00461c50 = in_SS;
  _DAT_00461b80 = IsDebuggerPresent();
  FUN_0040280a();
  SetUnhandledExceptionFilter((LPTOP_LEVEL_EXCEPTION_FILTER)0x0);
  UnhandledExceptionFilter((_EXCEPTION_POINTERS *)&PTR_DAT_0040614c);
  if (_DAT_00461b80 == 0) {
    FUN_0040280a();
  }
  uExitCode = 0xc0000409;
  hProcess = GetCurrentProcess();
  TerminateProcess(hProcess,uExitCode);
  return;
}
```

VirusTotal – Static Analysis Simulation



https://www.virustotal.com/gui/file/63545fa195488ff51955f09833332b9660d18f8afb16bdf579
120%
63545fa195488ff51955f09833332
DETECTION
DETAILS
Go to matched rule
Copy ruleset
Download
Crowdsourced Sigma Rules

Ruleset: gen_mimikatz
https://github.com/Neo23x0/signature-base

Crowdsourced YARA rules

- Matches rule Petya from ruleset
  ↳ Petya Payload - 8 days ago
- Matches rule INDICATOR_SUSPI
  ↳ Detects executables contain
- Matches rule INDICATOR_SUSPI
  ↳ Detects executables contain
- Matches rule Petya from ruleset
  ↳ Petya Payload
- Matches rule Win32_Ransomwa ReversingLabs
  ↳ Yara rule that detects NotPet
- Matches rule mimikatz from rul
  ↳ mimikatz

```
import "pe"

/* Mimikatz */

rule Mimikatz_Memory_Rule_1 : APT {
   meta:
      author = "Florian Roth"
      date = "2014-12-22"
      modified = "2023-07-04"
      score = 70
      nodeepdive = 1
      description = "Detects password dumper mimikatz in memory (False Positives: a
      id = "55cc7129-5ea0-5545-a8f6-b5306a014dd0"
   strings:
      $s1 = "sekurlsa::wdigest" fullword ascii
      $s2 = "sekurlsa::logonPasswords" fullword ascii
      $s3 = "sekurlsa::minidump" fullword asc
      $s4 = "sekurlsa::credman" fullword asci

      $fp1 = "\"x_mitre_version\": " ascii
      $fp2 = "{\"type\":\"bundle\","
      $fp3 = "use strict" ascii fullword
```

Click to start dragging "VirusTotal - File - 63545fa195488ff51955f09833332b9660d18f8afb16bdf579134661962e548a — Mozilla Firefox"

Threat Actor(s) Profiling

- **Threat actors identities:** The threat actors group is called **Sandworm Team** with group **ID G0034.** Sandworm Team is a destructive threat group that has been attributed to Russia's General Staff Main Intelligence Directorate (GRU) Main Center for Special Technologies (GTsST) military unit 74455. This group has been active since at least 2009. This seems to be Russian state Actors group and most likes state sponsored against Ukraine. There are some other associated groups like ELECTRUM, Telebots, IRON VIKING, Black Energy group, etc https://attack.miter.org/groups/G0034/

- **Motivations:** The motive behind NotPetya is widely believed to be disruption and sabotage, particularly targeting Ukraine, rather than financial gain.
- **Capabilities: Wiper Malware Masquerading as Ransomware:** Unlike typical ransomware that allows victims to recover their files upon payment of a ransom, NotPetya was designed to permanently damage or destroy data. It displayed a ransom note, but the mechanism for decrypting the files was ineffective, indicating that the attackers never intended to restore the affected systems.
- Track records : They attacked Ukraine electric power systems in 2015, 2016 and 2022

Attribution Indicators

- **NotPetya** has been attributed by several countries, including the United States and the United Kingdom, to state-sponsored actors associated with the Russian government. This attribution is based on the malware's characteristics, targets, and the geopolitical context, suggesting its use as a cyber weapon in the broader context of political tensions between Russia and Ukraine. NotPetya is widely attributed to state-sponsored actors, specifically the Russian military, as part of cyber operations against Ukraine. This attribution is based on its targets, timing, and destructive nature, aligning with geopolitical motives rather than criminal profit unlike Petya ransomware
- **NotPetya** is a prime example of how cyber warfare and cyber sabotage have become integral components of modern geopolitical conflicts, demonstrating the potential for malware to cause widespread and severe damage to national economies and global infrastructure.
- **Wiper Malware Masquerading as Ransomware:** Unlike typical ransomware that allows victims to recover their files upon payment of a ransom, NotPetya was designed to permanently damage or destroy data. It displayed a ransom note, but the mechanism for decrypting the files was ineffective, indicating that the attackers never intended to restore the affected systems.

- **Code similarities/infrastructure patterns/Targeting techniques:** The NotPetya codes is unlike that of Petya was a typical Ransomware that provided accompanying scripts for decryption after ransom is paid. NotPetya was more destructive, not intended for victims to recover as there was no accompanying decryption scrip. It was an advancement of Petya codes and automation.

- **Petya:** First identified in 2016, Petya was a genuine piece of ransomware designed to extort money from victims by encrypting files on their computer and demanding a ransom for the decryption key. – www.portnox.com/cybersecyrity 101
- **NotPetya:** Emerging in 2017, NotPetya masqueraded as ransomware similar to Petya but was primarily designed for disruption and destruction. It is considered a wiper malware disguised as ransomware. – www.portnox.com/cybersecyrity

Threat Landscape analysis:

The current Ransomware threats in supply chain landscape include targeted attacks, data corruption / exfiltration and the rise of Ransomware-as-a-Service (RaaS). There is also threat of malicious codes injection into open-source libraries and exploitation of developer's resources

- ### Recent ransomware trends: **(www.wavenet.co.uk)**

- Targeted attacks include business and infrastructures with weak cybersecurity postures. Good example is the recent Deloitte attack in London. Victims of NotPetya attack were those with outdated SMB service (SMBv1) which posed high vulnerability and thus exploited. Software development is also evolving rapidly with increased incentives for the threat actors, hence the supply chain landscape is also targeted.
- The case of data exfiltration and exploitation of open source libraries for software developers/suppliers also cannot be over emphasized
- **RaaS:** Ransomware remains one of the most significant threats to organizations, showing no signs of slowing down. In fact, 66% of organizations were affected by a ransomware attack last year 2024. Worse yet, these attacks are evolving, with a new trend with multiple extortions. In double extortion, attackers first demand payment to unlock encrypted files or systems, followed by a second ransom to prevent the release of sensitive data online or on the dark web.
- Ransomware-as-a-Service (RaaS) platforms are expected to grow, enabling less-skilled criminals to carry out sophisticated attacks. The use of AI-driven encryption and more advanced payload delivery methods will make ransomware even more effective and harder to defend against.
- **Supply Chain Attacks:** Supply chain attacks will remain a significant security threat in 2025 due to the increasing complexity and interconnected nature of global supply chains. More and more organizations are relying on third-party vendors for critical services and software, and as a result, the attack surface expands, providing cyber criminals with more entry points to exploit.

  These attacks are particularly dangerous because they target trusted relationships, allowing threat actors to bypass traditional security defenses and gain access to sensitive systems across multiple organizations. With the rise of digital transformation, reliance on cloud services, and the continued use of open-source software, supply chain vulnerabilities will become even more attractive for attackers, making vendor risk management and monitoring an essential undertaking for organizations moving forward.

- **AI Driven Attacks:** As AI and machine learning (ML) become more widespread, cyber criminals are leveraging them to automate cyber-attacks, craft advanced phishing emails, and exploit vulnerabilities. This growth shows no sign of slowing.
- The rest Threats include Insider threats, Application Interfaces (APIs) and Board-level cyber representation. The board of directors at various organizations should not leave the issue of cyber security to the IT staff/managers to handle without fair hearing from them and support them

- **Threat actors emerging TTPs:** The recently emerging threat actors TTPs in software supply chain landscape include targeting Remote Monitoring and Management (RMM) tools, fileless malwares using PowerShell/PSExec tools, custom programmable logical controllers (PLCs) malwares and advanced cloud security automation systems. Also leveraging golden SAML attacks, integrating automated intelligence while focusing on supply chain vulnerabilities.
  They are also augmenting traditional ransomware with operational disruption attacks

Mitigation strategies

Having researched, studied and analyzed the NotPetya incident and associated TTPs especially with respect to software supply chain vulnerabilities, there are several opportunities for cybersecurity posture improvement in general and software supply chain in particular

- **Security posture improvement opportunities:**

- The need for regular comprehensive vulnerability assessment and management cannot be over-emphasized. The victims of NotPetya incident obviously did not identify or paid attention to existing vulnerabilities in their respective cyber landscape. Identifying and eliminating vulnerabilities is an integral part of risk management that fences out potential threat Actors and maintains security-in-depth.
- Monitor, harden and secure continuous integration, continuous delivery & continuous deployment (CI/CD) pipelines applying zero trust to prevent malicious code injection into softwares and their subsequent updates/patches. Maintain tight access controls on the CI/CD pipelines in SDLC
- Software services security patching and upgrade requirements. The NotPetya attackers leveraged on the existing SMBv1 that some companies were still using even when newer versions were already available. The numerous vulnerabilities in the SMBv1 were fully exploited. It is expedient to always maintain regular systems security update/upgrades to eliminate known vulnerabilities. Replace all SMBv1 with currently updated SMBv2/SMBv3 to eliminate known deficiencies/vulnerabilities
- Good knowledge of threat actors capabilities to bypass security monitoring systems like SIEM tools and access control systems is crucial implementing strategies that will make life difficult for potential threat actors thereby keeping them away from the business
- Establishing as system that can help identify insider-threat actors and implement controls to check their activities while still in the organization is also crucial.
- Businesses should explore and implement measures to prevent possibilities of threat actors obfuscation by bypassing security monitoring system (SIEMS) using tools like Mimikatz & PSExec tools of the window systems
- **Network Segmentation:** The importance of network segmentation cannot be over emphasized. Segmenting critical endpoints and services into different LAN networks in the organization makes the entire security posture more resilient. When a network segment is attacked, business can continue with other available segments thereby effectively managing downtime impacts and reduce business losses

- ## Software supply chain security improvement:

  Using Software supply chain as a case-study in this project and the identified attendant vulnerabilities, the following improvement opportunities should be considered;

- Integrating cybersecurity services in the entire software development lifecycle (SDLC) is crucial to maintaining its integrity throughout the entire supply chain process (From cradle to grave). Threat actors are out looking for loopholes/weak point in every stage of the SDLC /supply chain, try to exploit them when they find one. Also implement DevSecOps and automate security testing
- Considering Threat actors exploitation of open-source libraries, software developers and supply chain managers should properly risk assess the open source platforms, applications and tools before using them. Properly manage the risk of using them from cradle to grave and maintain the integrity/reliability for the software products through their lifecycle. Threat actors have strong appetite for open-source resources
- Implement Software Bill of Materials (SBOM). Create SBOMs for softwares with detailed inventories of dependencies and their vulnerabilities. Analyze the identified vulnerabilities and manage associated risks.

  Also require SBOMs from third party vendors to get better visibility into their software and dependencies and associated risk mitigations
- Implement regular security audits, vulnerability management and penetration testing, develop actionable items and manage them to closure to software integrity and users trust even while in production line.
- **Data security and governance:** Secure software data transfer at all time bot on transit and at rest. Implement Data Loss Prevention (DLP) to avoid sensitive data escaping from the organization's control throughout the supply chain process or SDLC
- **Training:** Security training of everyone directly or indirectly involved in the software supply chain process is crucial to safeguarding it in its lifetime. It also help to build and maintain the trust and confidence of the customers/users and remain in business

  Credit: Software Supply Chain Security | Wiz