

Detecting Windows Based Exploit Chains by Means of Event Correlation and Process Monitoring

By **Muhammad Mudassar Yamin**

COINS Winter School

5-10 May 2019 | Finse

What is an Exploit Chain?

- An exploit chain is a group of exploits that executes synchronously, in order to achieve the system exploitation.
- Unlike high-risk vulnerabilities that allow system exploitation using only one execution step, an exploit chain takes advantage of multiple medium and low risk vulnerabilities

Single Vulnerability Being Exploited To Achieve Exploitation

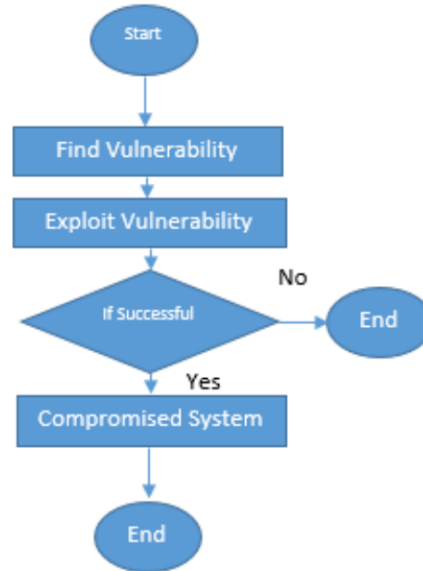


Fig (1) Example of traditional exploit with a single vulnerability

Multiple Vulnerabilities Exploited To Achieve Exploitations



Fig (2) Example of exploit chain with multiple vulnerability

Example of Exploit Chain

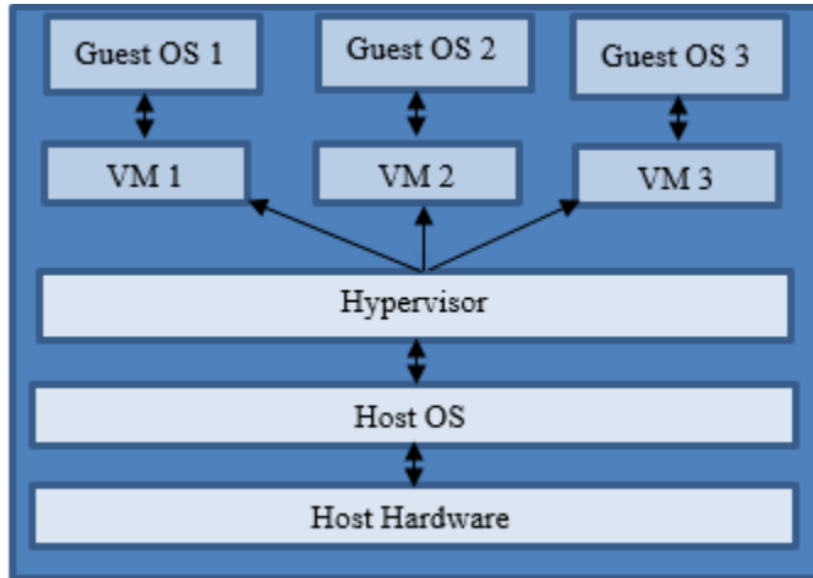


Fig (3) Isolated guest and host in virtualized environment

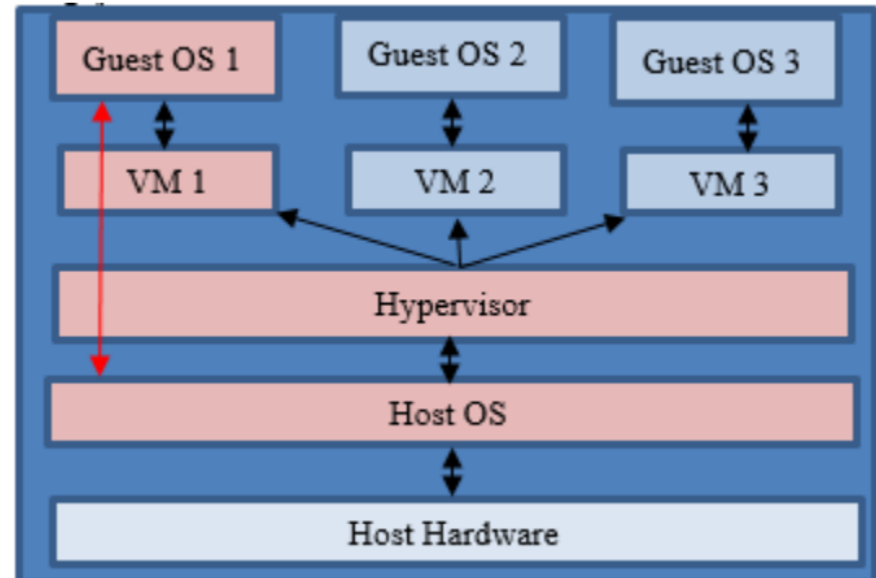


Fig (4) Broken isolation between guest and host

Guest To Host Exploit

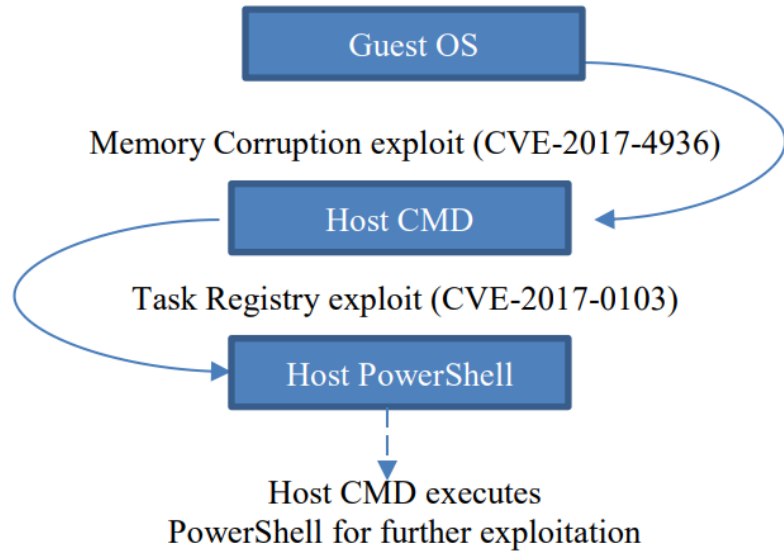


Fig (5) Guest to host escape exploit chain

The screenshot shows the Process Hacker interface for the user IPHONE\Mudassar. The 'Processes' tab is active, displaying a list of running processes. The 'powershell.exe' process is highlighted in yellow.

Name	PID	CPU	I/O total ...	Private b...
cmd.exe	9428			3.74 MB
conhost.exe	14192			6.3 MB
vmware.exe	10324			29.07 MB
cmd.exe	2248			2.63 MB
conhost.exe	10036			6.04 MB
powershell.exe	9416	0.01		48.03 MB

Fig (6) Guest to host exploit execution

Event Logging

- Event logging mechanism allows the identification of the type of computer events happening in Windows based systems when an exploit is executed.
- Structure of New Windows Process Creation Event
 - **SubjectUserSid** : Security id of account from where the process is executed
 - **SubjectUserName** : Account name from where the process is executed
 - **SubjectDomainName** : Domain Name
 - **SubjectLogonId** : Logon id of account from where the process is executed
 - **NewProcessId** : Unique hexadecimal new process identifier
 - **NewProcessName** : New process name executed by parent process
 - **ProcessId** : Unique hexadecimal process identifier
 - **CommandLine** : Command which is executed
 - **TargetUserSid** : Security id of account on which process executed
 - **TargetUserName** : User name
 - **TargetDomainName** : Computer name
 - **TargetLogonId** : Login id of account on which process executed
 - **ParentProcessName** : Name of process which executes new process
 - **MandatoryLabel** : Secure object control integrity label assigned to new process

Process Monitoring

- Process being executed after exploitation
 - CMD
 - PowerShell
 - Regsvr
 - Rundll32
- ..etc

Algorithm [noun]:

Word used by programmers when they do not want to explain what they did.

Exploit Chain Detector Algorithm

Exploit Chain Detector (ECD) Algorithm

Input: a list of ordered Windows event logs A; a list of process names to be monitored B

/ an event logs has the following attributes: NewProcessId, ProcessId, ProcessName, TargetDomainName*/*

/ B contains a list of process names that are executed after a vulnerability is exploited retrieved from report¹ [15] */*

Output: a list of string stacks D, a Boolean represents if exploit chains are detected c

/ D will contain all exploit chains detected by the algorithm, and c is true if one chain is found*/*

Initialization: create an empty event log a ; initialize c with the value false ; create integer m with initial value 0

```
1 for (i=0; i<Size(A); i++) do
2     if (Ai.ProcessId ∈ B) then
3         a=Ai
4         for (j=i; i<Size(A); j++) do
5             if (a.ProcessId == Aj.NewProcessId && a.TargetDomainName == Aj.TargetDomainName) then
6                 Dm.Push(a.ProcessName)
7                 a=Aj
8                 if (A(j+m).NewProcessId==Null) then
9                     c=true
10                    m=m+1
11                end if
12            end if
13        end for
14    end if
15 end for
```

Working

```
- EventData
  SubjectUserSid S-1-5-21-703565726-1159332285-768548448-1001
  SubjectUserName Mudassar
  SubjectDomainName IPHONE
  SubjectLogonId 0x8abbe
  NewProcessId 0xf18
  NewProcessName C:\Windows\SysWOW64\WindowsPowerShell\v1.0\powershell.exe
  TokenElevationType %%1938
  ProcessId 0xa10
  CommandLine powershell
  TargetUserSid S-1-0-0
  TargetUserName -
+ System
- EventData
  SubjectUserSid S-1-5-21-703565726-1159332285-768548448-1001
  SubjectUserName Mudassar
  SubjectDomainName IPHONE
  SubjectLogonId 0x8abbe
  NewProcessId 0xa10
  NewProcessName C:\Windows\SysWOW64\cmd.exe
  TokenElevationType %%1938
  ProcessId 0x270
  CommandLine "C:\Windows\System32\cmd.exe"
  TargetUserSid S-1-0-0
  TargetUserName -
+ System
- EventData
  SubjectUserSid S-1-5-21-703565726-1159332285-768548448-1001
  SubjectUserName Mudassar
  SubjectDomainName IPHONE
  SubjectLogonId 0x8abbe
  NewProcessId 0x270
  NewProcessName C:\Program Files (x86)\VMware\VMware Workstation\vmware.exe
  TokenElevationType %%1938
  ProcessId 0x189c
  CommandLine "C:\Program Files (x86)\VMware\VMware Workstation\vmware.exe"
  TargetUserSid S-1-0-0
  TargetUserName -
```

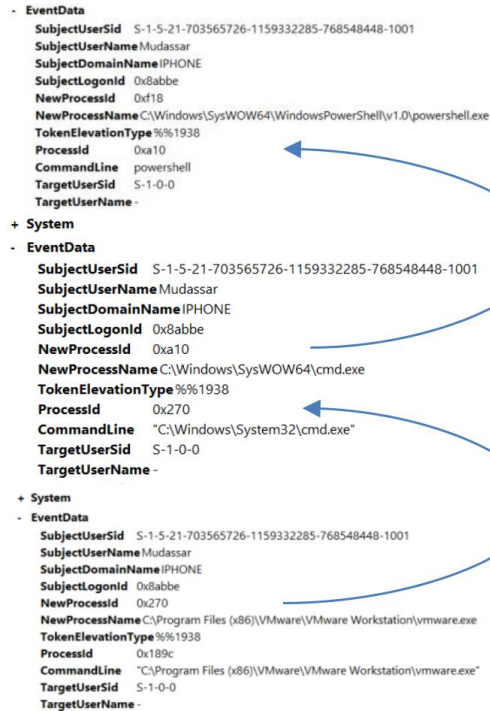
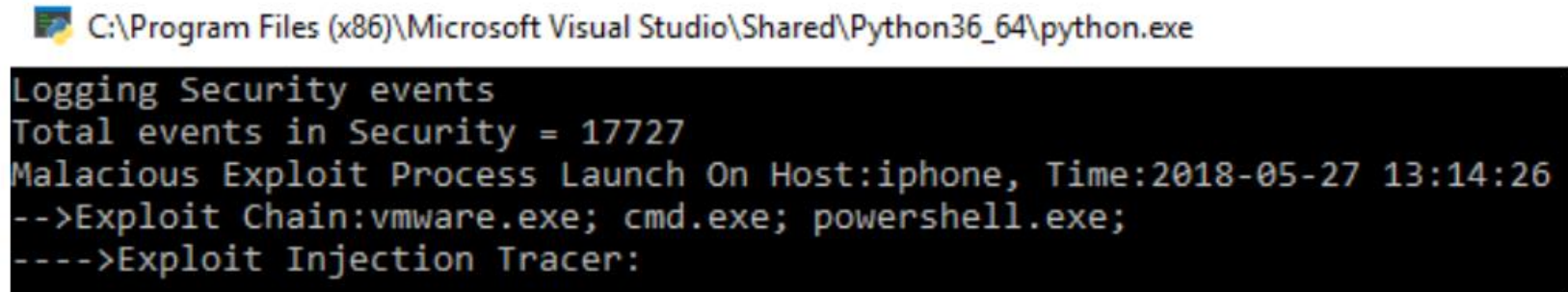


Fig (7) Windows event logs generated from a guest to host exploit

Detected Exploit Chain



```
C:\Program Files (x86)\Microsoft Visual Studio\Shared\Python36_64\python.exe  
Logging Security events  
Total events in Security = 17727  
Malicious Exploit Process Launch On Host:iphone, Time:2018-05-27 13:14:26  
-->Exploit Chain:vmware.exe; cmd.exe; powershell.exe;  
---->Exploit Injection Tracer:
```

Fig (8) Guest-to-host exploit detection

Experiments

Nick Carr @ItsReallyNick · 18 Jun 2018
"No engines detected this file" 🤔
Let's change that: gist.github.com/itsreallynick/...
^apply this #yara rule to unzipped OOXML contents as well!

Pictured: "alpha.docx"
@VirusTotal (0/60): virustotal.com/#/file/52f5356...
Technique by @enigma0x3: posts.specterops.io/the-tale-of-se...

No engines detected this file

SHA-256 52f53561f68b971cc32734
File name alpha.docx
File size 17.1 KB
Last analysis 2018-06-15 23:20:02 UTC

0 / 60

Detection	Details	Relations	Community
Ad-Aware		Clean	✓
AegisLab		Clean	✓
AhnLab-V3		Clean	✓
Alibaba		Clean	✓

Nick Carr @ItsReallyNick Following

A PowerPoint version of the @enigma0x3 method.

"Exploit Sample.pptx"
Pops calc.
MD5:
098aa57c9f3d8b365c074832e08d1cf0
Possibly by @MuddasarYamin
Uploaded 2 minutes ago:
[virustotal.com/#/file/ffa050f ...](https://virustotal.com/#/file/ffa050f...)

10 engines detected this file

SHA-256 098aa57c9f3d8b365c074832e08d1cf0
File name Exploit Sample.pptx
File size 2.0 MB
Last analysis 2018-06-16 13:01:01 UTC

Engine	Detection
Avast	Trj/MS-Office-Exploit-1
BitDefender	Trj/MS-Office-Exploit-1
Bkav	Trj/MS-Office-Exploit-1
Cybereason	Trj/MS-Office-Exploit-1
Emsisoft	Trj/MS-Office-Exploit-1
Fortinet	Trj/MS-Office-Exploit-1
Genie	Trj/MS-Office-Exploit-1
Gridin	Trj/MS-Office-Exploit-1
Hybrid	Trj/MS-Office-Exploit-1
Ikarus	Trj/MS-Office-Exploit-1
Jiangmin	Trj/MS-Office-Exploit-1
K7AntiVirus	Trj/MS-Office-Exploit-1
MaxSecure	Trj/MS-Office-Exploit-1
McAfee	Trj/MS-Office-Exploit-1
Microsoft	Clean

File Info: Title: Exploit Sample.pptx, Package: PowerPoint Presentation, LinkUpToDate: No, LastModifiedBy: Muhammad Muddasar Yamin, Application: Microsoft Office PowerPoint, Zip(FileName): [Content_Type].xml, CreateDate: 2018-06-20 11:19:23Z, ZipRequirePassword: No, PresentationFormat: Widescreen, ModifyDate: 2018-06-20 11:19:23Z, ZipCrc: 0x1318186d, Slides: 1, Words: 0, ScaleCrop: No, RevisionNumber: 1, MIMEType: application/vnd.openxmlformats-officedocument.presentationml.presentation, ZipBiffFlag: 0x0006, File Type: PPTX, Paragraphs: 0, AppVersion: 15.0, TotalCompressionRatio: 5387

4:30 AM - 20 Jun 2018

2 Retweets 8 Likes

Experimental Result Comparison

Solution	Detection Yes/No
Proposed Algorithm	Yes
Ad-Aware	No
AegisLab	No
AhnLab-V3	No
ALYac	No
Antiy-AVL	No
Arcabit	No
Avast	No
Avast Mobile Security	No
AVG	No

Table (1) Result of Comparative Detection Analysis of Developed algorithm and Different Software Security Software

List of Publications

- Yamin, Muhammad Mudassar, Basel Katt, and Vasileios Gkioulos. "Detecting Windows Based Exploit Chains by Means of Event Correlation and Process Monitoring." In Future of Information and Communication Conference, pp. 1079-1094. Springer, Cham, 2019.
- Yamin, M. M., & Katt, B. A Survey of Automated Information Exchange Mechanisms Among CERTs.
- Yamin, M. M., & Katt, B. (2019, January). Mobile device management (MDM) technologies, issues and challenges. In Proceedings of the 3rd International Conference on Cryptography, Security and Privacy (pp. 143-147). ACM.
- Yamin, Muhammad Mudassar, Basel Katt, Kashif Sattar, and Maaz Bin Ahmad. "Implementation of Insider Threat Detection System Using Honeypot Based Sensors and Threat Analytics." In Future of Information and Communication Conference, pp. 801-829. Springer, Cham, 2019.
- Yamin, Muhammad Mudassar, and Basel Katt "Ethical Problems and Legal Issues in Development and Usage Autonomous Adversaries in Cyber Domain"
- Yamin, Muhammd Mudassar, and Basel Katt. "Detecting Malicious Windows Commands Using Natural Language Processing Techniques." In International Conference on Security for Information Technology and Communications, pp. 157-169. Springer, Cham, 2018.
- Yamin, Muhammad Mudassar, and Basel Katt. "Inefficiencies in Cyber-Security Exercises Life-Cycle: A Position Paper." AAAI Adversarial Aware Learning Symposium 2018
- Yamin, Muhammad Mudassar, Basel Katt, Espen Torseth, Vasileios Gkioulos, and Stewart James Kowalski. "Make it and Break it: An IoT Smart Home Testbed Case Study." In Proceedings of the 2nd International Symposium on Computer Science and Intelligent Control, p. 26. ACM, 2018.
- Awan, K. M., ur Rehman, Z., Yamin, M. M., & Shah, P. A. (2017, December). Implementation of information security techniques on modern android based Kiosk ATM/remittance machines. In 2017 International Conference on Information and Communication Technologies (ICICT) (pp. 75-80). IEEE.

“Questions are guaranteed in life;
answers aren’t”