




Sign in


 OTRF / **detection-hackathon-apt29**

Public


 Notifications


 Fork

41


 Star


132


 Code


 Issues


49

 Pull requests

 Actions

 Projects

 Security

 Insights

1.A) User Execution, Masquerading, Uncommonly Used Port #1

New issue

Open

Cyb3rWard0g opened this issue on May 2, 2020 · 16 comments



Cyb3rWard0g commented on May 2, 2020

Contributor



Description

The scenario begins with an initial breach, where a legitimate user clicks (T1204) an executable payload (screensaver executable) masquerading as a benign word document (T1036). Once executed, the payload creates a C2 connection over port 1234 (T1065) using the RC4 cryptographic cipher



Cyb3rWard0g changed the title ~~User Execution, Masquerading, Uncommonly Used Port~~ 1.A) User Execution, Masquerading, Uncommonly Used Port on May 2, 2020

Assignees

No one assigned

Labels

None yet

Projects

None yet

Milestone

No milestone

Development

No branches or pull requests

5 participants





emiliedns commented on May 2, 2020

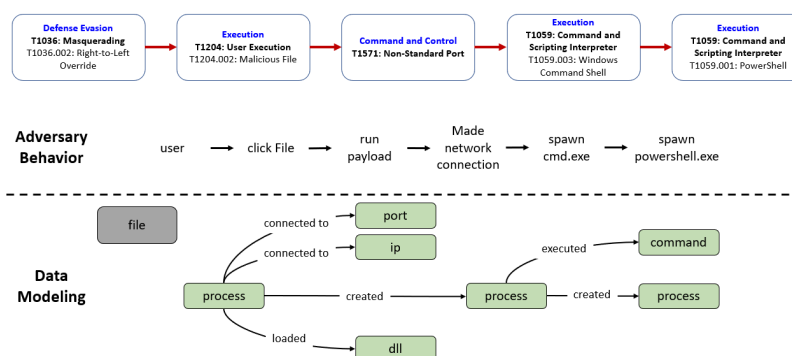


C:\ProgramData\victim\â€®cod.3aka3.scr uses Right To Left Override
rule idea: if it is possible, detect the use of unicode [U+202E] on non arabic machine for sysmon event 11 on executable files (exe, scr...)



Cyb3rPandaH commented on May 2, 2020

Collaborator



2



1



emiliedns commented on May 2, 2020 • edited



```
files = spark.sql(  
    '''  
    SELECT Image, TargetFilename  
    FROM apt29Table  
    WHERE Channel = "Microsoft-Windows-Sysmon/Operational"  
    AND EventID = 11 AND TargetFilename LIKE "%.scr%" '''  
    files.show(40)
```

is not picking anything so I am probably missing out something there :)

I tried to look for alternate data stream (download evidence) without much luck

```
files = spark.sql(  
    ""  
    SELECT TargetFilename  
    FROM apt29Table  
    WHERE Channel = "Microsoft-Windows-Sysmon/Operational"  
    AND EventID = 15 AND TargetFilename LIKE "%.exe%" ""  
    files.show(truncate = False, vertical = True)
```



Cyb3rPandaH commented on May 2, 2020 •
edited ▼

Collaborator

...

I think the record for event 11 that you are looking for is in capital letters LOL. I got one result when using SCR. In those cases you can use LOWER

```
In [21]: test = spark.sql(  
    ""  
    SELECT Image, TargetFilename  
    FROM apt29Table  
    WHERE Channel = "Microsoft-Windows-Sysmon/Operational"  
    AND EventID = 11  
    AND LOWER(TargetFilename) LIKE "%.scr%"  
    ""  
    test.show(truncate = False, vertical = True)  
    # AND TargetFilename LIKE "%.scr%"  
    -----  
    -RECORD 0-----  
    Image           | C:\windows\system32\svchost.exe  
    TargetFilename  | C:\Windows\Prefetch\A€*COD.3AKA3.SCR-7BD94A72.pf
```



1



1



Cyb3rWard0g commented
on May 2, 2020

Contributor

Author

...

```
files = spark.sql(  
    ""  
    SELECT Image, TargetFilename  
    FROM apt29Table  
    WHERE Channel = "Microsoft-Windows-  
    Sysmon/Operational"  
    AND EventID = 11 AND TargetFilename LIKE "%.scr%" ""  
    files.show(40)
```

is not picking anything so I am probably missing out something there :)

I tried to look for alternate data stream (download evidence) without much luck

```
files = spark.sql(
'''
SELECT TargetFilename
FROM apt29Table
WHERE Channel = "Microsoft-Windows-
Sysmon/Operational"
AND EventID = 15 AND TargetFilename LIKE "%.exe%" ''')
files.show(truncate = False, vertical = True)
```

Yeah it doesn't look like @emiliedns ..mm..

```
networkConnection8524 = spark.sql(
...

SELECT TargetFilename

FROM apt29Table

WHERE Channel = "Microsoft-Windows-Sysmon/Operational"

AND EventID = 15

AND NOT TargetFilename LIKE "%.etl"

''')

networkConnection8524.show(truncate = False, vertical =

-RECORD 0-----
TargetFilename | C:\WindowsAzure\Logs\WaAppAgent_000002
-RECORD 1-----
TargetFilename | C:\WindowsAzure\Logs\RuntimeEvents_000
-RECORD 2-----
TargetFilename | C:\Windows\System32\Tasks\Microsoft\Wi
-RECORD 3-----
TargetFilename | C:\Windows\System32\Tasks\Microsoft\Wi
-RECORD 4-----
TargetFilename | C:\Windows\System32\Tasks\Microsoft\Wi
-RECORD 5-----
TargetFilename | C:\Windows\System32\Tasks\Microsoft\Wi
-RECORD 6-----
TargetFilename | C:\Windows\System32\Tasks\Microsoft\Wi
-RECORD 7-----
TargetFilename | C:\Windows\System32\Tasks\Microsoft\Wi
-RECORD 8-----
TargetFilename | C:\Windows\System32\Tasks\Microsoft\Wi
```



```
-RECORD 9-----  
TargetFilename | C:\Windows\System32\Tasks\Microsoft\Wi
```



emiliedns commented on May 2, 2020

...

maybe because of the way the automation was done, the browser wasn't used, was it? that could explain



Cyb3rWard0g commented
on May 2, 2020

Contributor

Author

...

Correct [@emiliedns](#) :) Good one! I didnt remember that one
https://github.com/hunters-forge/OSSEM/blob/master/data_dictionaries/windows/sysmon/events/event-15.md



gonzalomarcos commented on May 3, 2020 • edited

...

Right To Left Override files executed

```
SELECT `@timestamp`, NewProcessName, SubjectLogonId, ProcessId, ParentProcessName, NewProcessId, SubS  
FROM apt29Table  
WHERE Channel = "Security"  
AND EventID = 4688  
AND SUBSTRING_INDEX(NewProcessName, '\\\\', -1) LIK
```

Results

@timestamp	2020-05-02T02:55:57.748Z
NewProcessName	C:\ProgramData\victim\â€cod.3aka3.
SubjectLogonId	0x3731f3
CommandLine	"C:\ProgramData\victim\â€cod.3aka3

ProcessId	0x1158
ParentProcessName	C:\Windows\explorer.exe
NewProcessId	0x214c
File	â€œcod.3aka3.scr



Cyb3rWard0g commented

Contributor

Author



on May 3, 2020

I like this approach [@gonzalomarcos](#) ! Thank you for sharing. i wonder how something like that can be written in Sigma.
[@thomaspatzke](#) is that something that can be done with Sigma?



neu5ron commented on May 4, 2020

Contributor



[@Cyb3rWard0g](#) I can't find the executable download anywhere if I should move this somewhere else let me know, but here is a sigma rule for that:

```
title: Executable from Webdav
status: experimental
date: 2020/05/01
description: Detects executable access via webdav6
author: 'Adam Swan'
references:
  - http://carnal0wnage.attackresearch.com/2012/06/webdav
  - https://github.com/OTRF/detection-hackathon-apt29
tags:
  - attack.command_and_control
  - attack.T1043
logsource:
  category: proxy
detection:
  selection_webdav:
    - c-useragent: '*WebDAV*'
    - c-uri: '*webdav*'
  selection_executable:
    - resp_mime_types: '*dosexec*'
    - c-uri: '*.exe'
```



```
condition: selection_webdav AND selection_executable
falsepositives:
  - unknown
level: medium
```



Cyb3rWard0g commented

Contributor

Author



on May 4, 2020

Hey [@neu5ron](#) , I believe that goes to this Issue right? [#19](#) Let me know. . [@patrickstjohn](#) created one but to detect if it was a python application. So that query works there too! Thats awesome! Thank you Adam! If you can move the query there it would be awesome to track it! 👍



Cyb3rWard0g commented

Contributor

Author



on May 11, 2020

I wonder how noisy the SeProfileSingleProcessPrivilege user privileges requested is for non SYSTEM

```
rtlo = spark.sql(
...
SELECT PrivilegeList, SubjectUserName, ObjectServer, Pro
FROM apt29Table
WHERE EventID = 4673 AND LOWER(Message) LIKE "%3aka3%"
...
)
rtlo.show(2,truncate = False, vertical = True)
```



Results:

```
-RECORD 0-----
PrivilegeList | SeProfileSingleProcessPrivilege
SubjectUserName | pbeesly
ObjectServer | Security
ProcessName | C:\ProgramData\victim\â€œcod.3aka3.sc
```





Cyb3rWard0g commented

on May 11, 2020

Contributor

Author



I liked this query [@cyb3rpanda](#) from the Initial Exploratory analysis notebook:

```
networkConnection8524 = spark.sql(  
    '''  
  
    SELECT o.`@timestamp`, o.ProcessId, a.ParentImage, o.Ima  
    FROM apt29Table o  
    INNER JOIN (  
        SELECT Description, CommandLine, CurrentDirectory, P  
        FROM apt29Table  
        WHERE Channel = "Microsoft-Windows-Sysmon/Operationa  
            AND EventID = 1  
            AND ParentImage LIKE "%explorer%"  
        ) a  
    ON o.ProcessGuid = a.ProcessGuid  
    WHERE Channel = "Microsoft-Windows-Sysmon/Operational"  
        AND EventID = 3  
    '''  
networkConnection8524.show(truncate = False, vertical =
```



I was going over the APT29 Evals results and some EDR solutions also look for that combination.

`Execution of file from Explorer.exe with a network connection". Some just mention that the file is malicious while others actually say that it used the RTLO technique. I assume they somehow look for the Unicode string. However, that basic logic above seems to be considered by several detection rules (additional context)



Cyb3rWard0g commented

on May 11, 2020

Contributor

Author



Detection Categories

Main - Technique

(originally file during evams was executed from C:\users\ and not C:\programdata) However, the execution of the file was captured from C:\programdata\ and it would have been captured anyways from C:\users)

Process creation / Execution from users directory

```
SELECT Message
FROM apt29Table
WHERE Channel = "Microsoft-Windows-Sysmon/Operational"
AND EventID = 1 AND LOWER(CurrentDirectory) LIKE "c:_use
```



Main - General

Information about new process running on endpoint leveraging registry modifications to `\Software\Microsoft\Windows NT\CurrentVersion\AppCompatFlags\Compatibility Assistant\Store\`

```
SELECT Message
FROM apt29Table
WHERE Channel = "Microsoft-Windows-Sysmon/Operational"
AND EventID = 13 AND LOWER(TargetObject) LIKE "%appcompa
```



Cyb3rWard0g commented
on May 11, 2020 • edited ▼

Contributor

Author



Main - Telemetry

Execution of payload was captured

```
SELECT Message
FROM apt29Table
WHERE Channel = "Microsoft-Windows-Sysmon/Operational"
AND EventID = 1
AND LOWER(ParentImage) LIKE "%explorer.exe"
AND LOWER(Image) LIKE "%3aka3%"
```



Results:

-RECORD 0



Message | Process Create:

RuleName: -

UtcTime: 2020-05-02 02:55:56.157

ProcessGuid: {47ab858c-e13c-5eac-a903-000000000400}

ProcessId: 8524

Image: C:\ProgramData\victim\â€@cod.3aka3.scr

FileVersion: -

Description: -

Product: -

Company: -

OriginalFileName: -

CommandLine: "C:\ProgramData\victim\â€@cod.3aka3.scr" /S

CurrentDirectory: C:\ProgramData\victim\

User: DMEVALS\pbeesly

LogonGuid: {47ab858c-dabe-5eac-f331-370000000000}

LogonId: 0x3731F3

TerminalSessionId: 2

IntegrityLevel: Medium

Hashes: SHA1=4B7FA56A4E85F88B98D11A6E018698AE3FBA5E62,MD

ParentProcessGuid: {47ab858c-dac4-5eac-f202-000000000400}

ParentProcessId: 4440

ParentImage: C:\Windows\explorer.exe

ParentCommandLine: C:\windows\Explorer.EXE



Cyb3rWard0g commented

on May 11, 2020

Contributor

Author



1.A.4 Standard Cryptographic Protocol

Procedure: Used RC4 stream cipher to encrypt C2 (192.168.0.5) traffic

Criteria: Evidence that the network data sent over the C2 channel is encrypted

rsc.3aka3.doc loading cryptographic libraries



SELECT Image, count(*) as count

FROM apt29Table

WHERE Channel = "Microsoft-Windows-Sysmon/Operational"

AND EventID = 7 AND LOWER(ImageLoaded) LIKE "%bcrypt.dll"



GROUP BY Image
ORDER BY count DESC

[Sign up for free](#) to join this conversation on GitHub. Already have an account? [Sign in to comment](#)

[Terms](#) [Privacy](#) [Security](#) [Status](#) [Docs](#) [Contact](#) [Manage cookies](#) [Do not share my personal information](#)



© 2024 GitHub, Inc.