CyberDefense

MWS以外のデータセット紹介

MWS2023プレミーティング

2023/06/29

サイバーディフェンス研究所

中島将太

Malware関連のデータセット

- 有名なデータセット
  - Microsoft Malware Classification Challenge (BIG 2015)
  - EMBER dataset
  - SoReL-20M
- 興味深いデータセット
  - APT-Malware Dataset
  - Malpedia
- Google Dataset Search
- 研究利用のためのマルウェアの収集方法
  - VirusTotal
  - 無償利用可能サービス

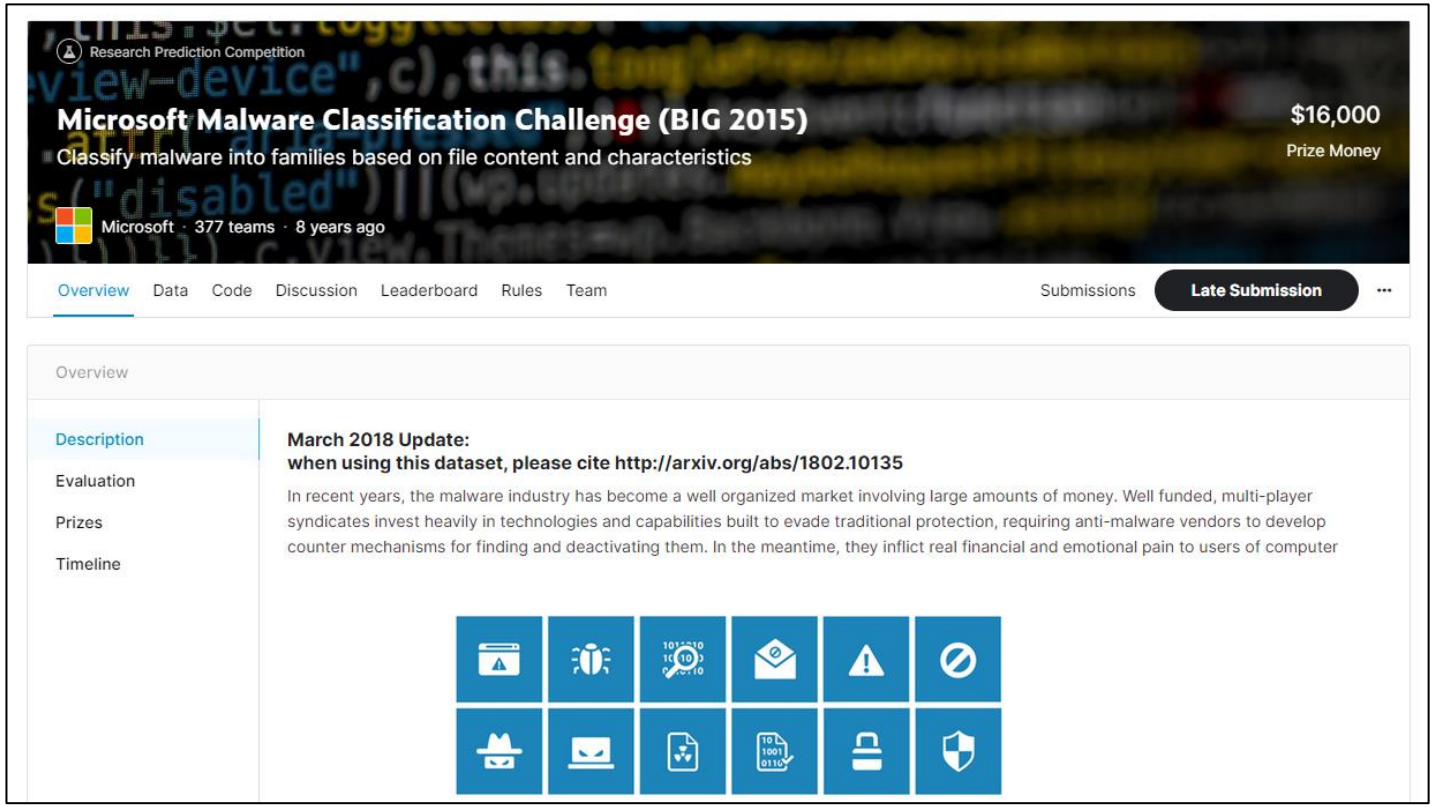
CyberDefense

有名なデータセット

Microsoft Malware Classification Challenge (BIG 2015)

- Microsoftが2015年にKaggleで開催したマルウェアの分類コンテストのデータセット
- 以下のマルウェアファミリが含まれる

1. Ramnit
2. Lollipop
3. Kelihos_ver3
4. Vundo
5. Simda
6. Tracur
7. Kelihos_ver1
8. Obfuscator.ACY
9. Gatak



Microsoft Malware Classification Challenge (BIG 2015)

- PEヘッダを削ったHEXダンプとIDAの逆アセンブルデータが提供されている

```
1   00401000 56 8D 44 24 08 50 8B F1 E8 1C 1B 00 00 C7 06 08
2   00401010 BB 42 00 8B C6 5E C2 04 00 CC CC CC CC CC CC CC
3   00401020 C7 01 08 BB 42 00 E9 26 1C 00 00 CC CC CC CC CC
4   00401030 56 8B F1 C7 06 08 BB 42 00 E8 13 1C 00 00 F6 44
5   00401040 24 08 01 74 09 56 E8 6C 1E 00 00 83 C4 04 8B C6
6   00401050 5E C2 04 00 CC CC CC CC CC CC CC CC CC CC CC CC
7   00401060 8B 44 24 08 8A 08 8B 54 24 04 88 0A C3 CC CC CC
8   00401070 8B 44 24 04 8D 50 01 8A 08 40 84 C9 75 F9 2B C2
9   00401080 C3 CC CC CC CC CC CC CC CC CC CC CC CC CC CC CC
10  00401090 8B 44 24 10 8B 4C 24 0C 8B 54 24 08 56 8B 74 24
11  004010A0 08 50 51 52 56 E8 18 1E 00 00 83 C4 10 8B C6 5E
12  004010B0 C3 CC CC CC CC CC CC CC CC CC CC CC CC CC CC CC
13  004010C0 8B 44 24 10 8B 4C 24 0C 8B 54 24 08 56 8B 74 24
14  004010D0 08 50 51 52 56 E8 65 1E 00 00 83 C4 10 8B C6 5E
15  004010E0 C3 CC CC CC CC CC CC CC CC CC CC CC CC CC CC CC
16  004010F0 33 C0 C2 10 00 CC CC CC CC CC CC CC CC CC CC CC
17  00401100 B8 08 00 00 00 C2 04 00 CC CC CC CC CC CC CC CC
18  00401110 B8 03 00 00 00 C3 CC CC CC CC CC CC CC CC CC CC
19  00401120 B8 08 00 00 00 C3 CC CC CC CC CC CC CC CC CC CC
20  00401130 8B 44 24 04 A3 AC 49 52 00 B8 FE FF FF FF C2 04
21  00401140 00 CC CC CC CC CC CC CC CC CC CC CC CC CC CC CC
22  00401150 A1 AC 49 52 00 85 C0 74 16 8B 4C 24 08 8B 54 24
23  00401160 04 51 52 FF D0 C7 05 AC 49 52 00 00 00 00 00 B8
24  00401170 FB FF FF FF C2 08 00 CC CC CC CC CC CC CC CC CC
25  00401180 6A 04 68 00 10 00 00 68 68 BE 1C 00 6A 00 FF 15
26  00401190 9C 63 52 00 50 FF 15 C8 63 52 00 8B 4C 24 04 6A
27  004011A0 00 6A 40 68 68 BE 1C 00 50 89 01 FF 15 C4 63 52
28  004011B0 00 B8 04 00 00 00 C2 04 00 CC CC CC CC CC CC CC
```

```
23975   .text:00471DFA 89 75 E8                                    mov     [ebp+var_18], esi
23976   .text:00471DFD 5E                                  pop     esi
23977   .text:00471DFE BB 01 24 43 06                             mov     ebx, 6432401h
23978   .text:00471E03 89 5D E8                                    mov     [ebp+var_18], ebx
23979   .text:00471E06 5B                                  pop     ebx
23980   .text:00471E07 8B 0D 0C 10 4B 00                          mov     ecx, dword_4B100C
23981   .text:00471E0D 83 FA 87                                    cmp     edx, 0FFFFFF87h
23982   .text:00471E10 74 0B                                       jz      short loc_471E1D
23983   .text:00471E12 81 FA 60 18 7E 7E                           cmp     edx, 7E7E1860h
23984   .text:00471E18 74 03                                       jz      short loc_471E1D
23985   .text:00471E1A 89 55 F8                                    mov     [ebp+var_8], edx
23986   .text:00471E1D
23987   .text:00471E1D                             loc_471E1D:                 ; CODE XREF: sub_471C4F+1C1CANj
23988   .text:00471E1D                                                         ; sub_471C4F+1C9CANj
23989   .text:00471E1D 8B E5                                       mov     esp, ebp
23990   .text:00471E1F 5D                                  pop     ebp
23991   .text:00471E20 C2 04 00                                    retn    4
23992   .text:00471E20                             sub_471C4F      endp
23993   .text:00471E20
23994   .text:00471E20                             ; ---------------------------------------------------------------------------
23995   .text:00471E23 88                                  db 88h
23996   .text:00471E24 20 04 4C 08 21 12 08 04 40 02 00 00 88 01 44 41            dd 84C0420h, 4081221h, 240h, 41440188h
23997   .text:00471E34                             ; [00000006 BYTES: COLLAPSED FUNCTION VirtualAlloc. PRESS KEYPAD CTRL-"+" TO EXPAND]
23998   .text:00471E3A                             ; [00000006 BYTES: COLLAPSED FUNCTION GetCurrentThreadId. PRESS KEYPAD CTRL-"+" TO EXPAND]
23999   .text:00471E40                             ; ---------------------------------------------------------------------------
24000   .text:00471E40 FF 25 1C F5 46 00                           jmp     ds:_lcreat
24001   .text:00471E46                             ; ---------------------------------------------------------------------------
24002   .text:00471E46 FF 25 E4 F4 46 00                           jmp     ds:FindNextVolumeA
24003   .text:00471E4C                             ; ---------------------------------------------------------------------------
24004   .text:00471E4C FF 25 E0 F4 46 00                           jmp     ds:WriteFile
24005   .text:00471E52                             ; ---------------------------------------------------------------------------
24006   .text:00471E52 FF 25 10 F5 46 00                           jmp     ds:RegisterConsoleIME
24007   .text:00471E58                             ; ---------------------------------------------------------------------------
24008   .text:00471E58 FF 25 24 F5 46 00                           jmp     ds:GetBinaryTypeW
24009   .text:00471E5E                             ; ---------------------------------------------------------------------------
24010   .text:00471E5E FF 25 18 F5 46 00                           jmp     ds:BackupWrite
24011   .text:00471E64                             ; ---------------------------------------------------------------------------
```

EMBER dataset

- Elasticが提供するデータセット
- LIEFで抽出した特徴量のみを提供している

README.md

Elastic Malware Benchmark for Empowering Researchers

The EMBER dataset is a collection of features from PE files that serve as a benchmark dataset for researchers. The EMBER2017 dataset contained features from 1.1 million PE files scanned in or before 2017 and the EMBER2018 dataset contains features from 1 million PE files scanned in or before 2018. This repository makes it easy to reproducibly train the benchmark models, extend the provided feature set, or classify new PE files with the benchmark models.

This paper describes many more details about the dataset: https://arxiv.org/abs/1804.04637

Features

The LIEF project is used to extract features from PE files included in the EMBER dataset. Raw features are extracted to JSON format and included in the publicly available dataset. Vectorized features can be produced from these raw features and saved in binary format from which they can be converted to CSV, dataframe, or any other format. This repository makes it easy to generate raw features and/or vectorized features from any PE file. Researchers can implement their own features, or even vectorize the existing features differently from the existing implementations.

The feature calculation is versioned. Feature version 1 is calculated with the LIEF library version 0.8.3. Feature version 2 includes the additional data directory feature, updated ordinal import processing, and is calculated with LIEF library version 0.9.0. We have verified under Windows and Linux that LIEF provides consistent feature representation for version 2 features using LIEF version 0.10.1 and that it does not on a Mac.

EMBER dataset


```
{"sha256": "9be720bdea0d62f182c01194ae6ad1520b5370c16ddc0088fa5c96e0e67b83ee", "md5": "b2b3fcbf73f718f287daa8268aa2d133", "appeared": "2018-01", "label": -1, "avclass":
"installmonster", "histogram": [61761, 34779, 42154, 33048, 32602, 27828, 41602, 34027, 34845, 23234, 35949, 25340, 27744, 24873, 40228, 33642, 32107, 21780, 29451, 22274, 23991,
22445, 29479, 25558, 26031, 17911, 32524, 21007, 26433, 22958, 32435, 23215, 24726, 20869, 26677, 21054, 21289, 16463, 26336, 23387, 20670, 16145, 28370, 21480, 21281, 17252,
29300, 22407, 23311, 16008, 25012, 17871, 21617, 18985, 25564, 19882, 25945, 15456, 31051, 21379, 23068, 20720, 29540, 20249, 28147, 23319, 30863, 26671, 18953, 22148, 31018,
26380, 20538, 16063, 24882, 20204, 16345, 17731, 26519, 23838, 24744, 18745, 25674, 21311, 21232, 21016, 27251, 22827, 21974, 15668, 27418, 18017, 20139, 18079, 28124, 20902,
24822, 20639, 24216, 20959, 17453, 16840, 24412, 18683, 26164, 17099, 26742, 21433, 17417, 16323, 23829, 23065, 29391, 17573, 25235, 18064, 25683, 22874, 24005, 20461, 28520,
16665, 27807, 19525, 18590, 16967, 25271, 21437, 32492, 25410, 29016, 32028, 23999, 20377, 27575, 25002, 16999, 19200, 25542, 32325, 18496, 20635, 27583, 20811, 18876, 15359,
21507, 16591, 13867, 14473, 22570, 17377, 14266, 12059, 23762, 15390, 16368, 13612, 25895, 17136, 23319, 21229, 25015, 22262, 15133, 14768, 21401, 19645, 19676, 16132, 25465,
19674, 17632, 16083, 25339, 22216, 22005, 14278, 21829, 16120, 18166, 16068, 22485, 17929, 21871, 12566, 24622, 16756, 20711, 17270, 26112, 19837, 33923, 25073, 30178, 24415,
17134, 17246, 26717, 21408, 16981, 14422, 21530, 15693, 13281, 14370, 21769, 18251, 26459, 19987, 23107, 17784, 18684, 16724, 24789, 21742, 17803, 12952, 23958, 16953, 14994,
12810, 26793, 18795, 33844, 23488, 26326, 19137, 16197, 12018, 20647, 16915, 25432, 16126, 24683, 23423, 15796, 14499, 21856, 21419, 31608, 18384, 21090, 15768, 20092, 15716,
22404, 19869, 24416, 11809, 22577, 16634, 17315, 13374, 30359, 57055], "byteentropy": [0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
0, 0, 0, 0, 0, 0, 0, 3137, 81, 114, 39, 89, 20, 25, 14, 154, 111, 23, 104, 103, 45, 22, 15, 6871, 332, 233, 125, 729, 337, 343, 181, 251, 200, 44, 188, 186, 118, 48, 54, 0, 0, 0, 0,
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1189, 30, 17, 42, 32, 21, 120, 54, 33, 96, 42, 19, 23, 160, 65, 105, 739, 7, 3, 4, 4, 4, 5, 6, 36, 114, 75, 45, 68, 359, 225, 354, 3703,
265, 225, 192, 614, 471, 391, 343, 364, 371, 445, 459, 487, 773, 727, 2458, 1095, 152, 116, 95, 70, 81, 201, 128, 153, 434, 492, 473, 371, 561, 613, 3157, 679, 89, 90, 95, 121,
96, 247, 168, 68, 76, 76, 71, 63, 50, 28, 31, 1689, 576, 465, 435, 434, 363, 362, 319, 394, 338, 380, 386, 371, 373, 371, 936, 1087287, 817564, 714055, 710203, 745035, 704751,
678383, 714818, 793435, 556407, 648317, 615421, 663043, 626222, 661506, 709825], "strings": {"numstrings": 31419, "avlength": 5.84086062573602, "printabledist": [2298, 1599,
1944, 1648, 1899, 1291, 2028, 1790, 1599, 1383, 1999, 1530, 1687, 1440, 2246, 1625, 1896, 1386, 2124, 1412, 1894, 1504, 1886, 1442, 2232, 1318, 2236, 1650, 2056, 1611, 2085,
1420, 2149, 2045, 2494, 2221, 1808, 1894, 2591, 1942, 1863, 1570, 1834, 1535, 1526, 1603, 2271, 2012, 2121, 1440, 2063, 2011, 2164, 1760, 2042, 1728, 1854, 1320, 1924, 1482,
1849, 1432, 1953, 1558, 2059, 2781, 2117, 2211, 1993, 2805, 2034, 1714, 2528, 2374, 2019, 1667, 2333, 1793, 2557, 2689, 2842, 1405, 2858, 2254, 3014, 2338, 1786, 1571, 2387,
1663, 2016, 1392, 1575, 1367, 1706, 1449], "printables": 183514, "entropy": 6.554454550415369, "paths": 0, "urls": 2, "registry": 0, "MZ": 74}, "general": {"size": 5748736,
"vsize": 13774848, "has_debug": 0, "exports": 0, "imports": 14, "has_relocations": 0, "has_resources": 1, "has_signature": 0, "has_tls": 1, "symbols": 0}, "header": {"coff":
{"timestamp": 1514305022, "machine": "I386", "characteristics": ["BYTES_REVERSED_LO", "RELOCS_STRIPPED", "EXECUTABLE_IMAGE", "CHARA_32BIT_MACHINE", "LINE_NUMS_STRIPPED",
"BYTES_REVERSED_HI", "LOCAL_SYMS_STRIPPED"]}, "optional": {"subsystem": "WINDOWS_GUI", "dll_characteristics": [], "magic": "PE32", "major_image_version": 0,
"minor_image_version": 0, "major_linker_version": 2, "minor_linker_version": 25, "major_operating_system_version": 5, "minor_operating_system_version": 0,
"major_subsystem_version": 5, "minor_subsystem_version": 0, "sizeof_code": 5234688, "sizeof_headers": 4096, "sizeof_heap_commit": 4096}}, "section": {"entry": "UPX0", "sections":
[{"name": "UPX0", "size": 0, "entropy": -0.0, "vsize": 8019968, "props": ["CNT_UNINITIALIZED_DATA", "MEM_EXECUTE", "MEM_READ", "MEM_WRITE"]}, {"name": "UPX1", "size": 5233152,
"entropy": 7.938365063177856, "vsize": 5234688, "props": ["CNT_INITIALIZED_DATA", "MEM_EXECUTE", "MEM_READ", "MEM_WRITE"]}, {"name": ".rsrc", "size": 512512, "entropy": 7.
966332178411693, "vsize": 516096, "props": ["CNT_INITIALIZED_DATA", "MEM_READ", "MEM_WRITE"]}]}, "imports": {"KERNEL32.DLL": ["LoadLibraryA"], "advapi32.dll": ["LogonUserA"],
"comctl32.dll": ["ImageList_Add"], "gdi32.dll": ["Pie"], "msvcrt.dll": ["memset"], "netapi32.dll": ["NetWkstaGetInfo"], "ole32.dll": ["OleDraw"], "oleacc.dll":
["LresultFromObject"], "oleaut32.dll": ["VariantInit"], "shell32.dll": ["Shell_NotifyIconW"], "user32.dll": ["GetDC"], "version.dll": ["VerQueryValueW"], "winmm.dll":
["timeGetTime"], "winspool.drv": ["OpenPrinterW"]}, "exports": [], "datadirectories": [{"name": "EXPORT_TABLE", "size": 154, "virtual_address": 11444224}, {"name":
"IMPORT_TABLE", "size": 864, "virtual_address": 13769992}, {"name": "RESOURCE_TABLE", "size": 511240, "virtual_address": 13258752}, {"name": "EXCEPTION_TABLE", "size": 0,
"virtual_address": 0}, {"name": "CERTIFICATE_TABLE", "size": 0, "virtual_address": 0}, {"name": "BASE_RELOCATION_TABLE", "size": 0, "virtual_address": 0}, {"name": "DEBUG",
"size": 0, "virtual_address": 0}, {"name": "ARCHITECTURE", "size": 0, "virtual_address": 0}, {"name": "GLOBAL_PTR", "size": 0, "virtual_address": 0}, {"name": "TLS_TABLE",
"size": 24, "virtual_address": 13256652}, {"name": "LOAD_CONFIG_TABLE", "size": 0, "virtual_address": 0}, {"name": "BOUND_IMPORT", "size": 0, "virtual_address": 0}, {"name":
"IAT", "size": 0, "virtual_address": 0}, {"name": "DELAY_IMPORT_DESCRIPTOR", "size": 3056, "virtual_address": 11440128}, {"name": "CLR_RUNTIME_HEADER", "size": 0,
"virtual_address": 0}]}
```

SoReL-20M

- SophosとReversingLabsが公開したデータセット
- 200万のマルウェアサンプルが含まれる
  - データ量は8TB
- マルウェアの生バイナリ、学習済みのモデル、メタ情報を処理済みデータなどがまとめて公開されている

CyberDefense

SoReL-20M

```
s3://sorel-20m/09-DEC-2020/
|   Terms and Conditions of Use.pdf -- the terms you agree to by using this data and code
|
+---baselines
|   +---checkpoints
|   |   +---FFNN - per-epoch checkpoints for 5 seeds of the feed-forward neural network
|   |   +---lightGBM - final trained lightGBM model for 5 seeds
|   |
|   +---results
|       |  ffnn_results.json - index file of results, required for plotting
|       |  lgbm_results.json - index file of results, required for plotting
|       |
|       +---FFNN
|       |   +---seed0-seed4 - individual seed results, ~1GB each
|       |
|       +---lightgbm
|           +---seed0-seed4 - individual seed results, ~1GB each
|
+---binaries
|      approximately 8TB of zlib compressed malware binaries
|
+---lightGBM-features
|      test-features.npz - array of test data for lightGBM; 37GB
|      train-features.npz - array of training data for lightGBM; 113GB
|      validation-features.npz - array of validation data for lightGBM; 22GB
|
+---processed-data
    |   meta.db - contains index, labels, tags, and counts for the data; 3.5GB
    |
    +---ember_features - LMDB directory with baseline features, ~72GB
    +---pe_metadata - LMDB directory with full metadata dumps, ~480GB
```

CyberDefense

興味深いデータセット

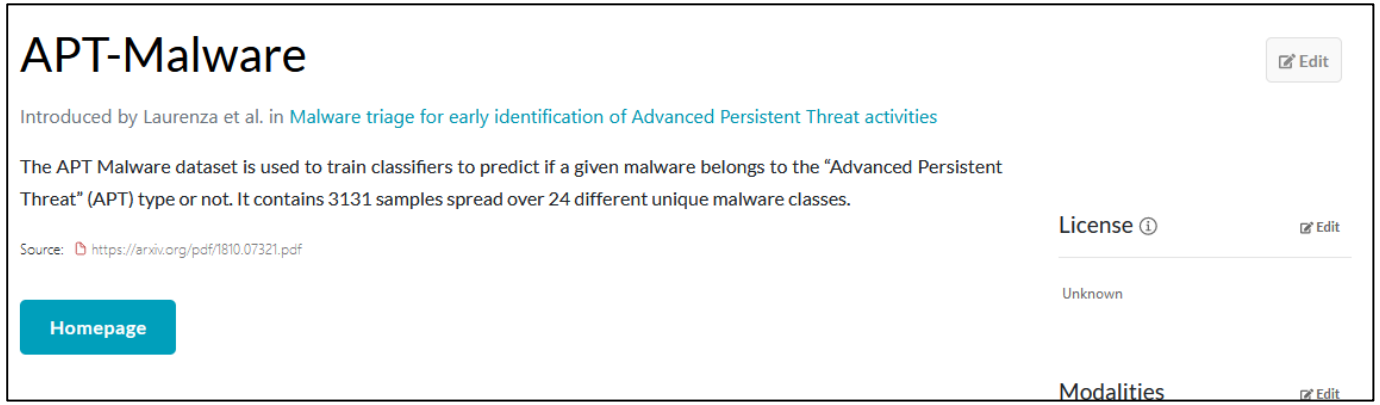
APT Malware dataset

- マルウェアがAPTのマルウェアであるかを予測する分類器を作成する研究
- 19アクターの24ファミリ、3131サンプルを学習
- 論文で利用したデータセットを公開している

Table 1: APTs elements per classes

Class	Count
Patchwork	559
APT29	205
Winnti Group	176
Carbanak	105
Volatile Cedar	35
NSA	13
Transparent Tribe	267
Shiqiang	31
Roaming Tiger	14
Mirage	54

Class	Count
Lazarus Group	58
Sandwork	44
Hurricane Panda	315
APT30	101
Violin Panda	23
Desert Falcon	45
Molerats	25
APT28	68
Lotus Blossom	48



CyberDefense

APT Malware dataset

- PEFrameを使ってマルウェアの静的な情報から4000以上の特徴を抽出して7クラスを作成

Optional Header (30 features). Every file has an optional header that provides information to the loader. This header is optional in the sense that some files (specifically, object files) do not have it. For image files, this header is required. An object file can have an optional header, but generally this header has no function in an object file except to increase its size. Features are extracted from the optional header of the PE and contain information about the logical layout of the PE file, such as the address of the entry point, the alignment of sections, and the sizes of part of the file in memory.

MS-DOS Header (17 features). The MS-DOS executable-file header is composed of four distinct parts: a collection of header information (such as the signature word, the file size, etc.), a reserved section, a pointer to a Windows header (if one exists), and a stub program. MS-DOS uses the stub program to display a message if Windows has not been loaded when the user attempts to run a program. In this contest, we are interested in features related to the execution of the file, including the number of bytes in the last page of the file, the number of pages, or the starting address of the Relocation Table.

File Header (18 features). The Windows executable-file header contains information that the loader requires for segmented executable files. This includes the linker version number, data specified by the linker, data specified by the resource compiler, tables of segment data, and tables of resource data. Moreover, the features related to this class highlight information about timestamp and the CPU platform that the PE is intended for.

Obfuscated String Statistics (3 features). Binaries contain program messages stored as strings that can be useful to understand their behavior. Classical tools like *String* extract byte sequences that can be readable strings to find these messages. Malware authors encode strings in their program to avoid extraction, in fact, even simple schemes can defeat this kind of tool and complicate static and dynamic analysis. In addition to PEFrame, we use functionalities of the FireEye Labs Obfuscated String Solver (*FLOSS*[3]). It is an open-source tool that automatically detects, extracts, and decodes obfuscated strings, such as malicious domains, IP addresses, suspicious file paths, and so on, from Windows Portable Executable files availing of advanced static analysis techniques. We leverage this tool to compute some statistics, such as how many entry-points or relocations are present in the file.

Mutex (7 features). Mutex are objects commonly used to avoid simultaneous access to a resource, like a variable. If different software checks for the same mutex, then they can be linked. Our features are Boolean values that map the use of particular mutex identified in the training data.

Packer (64 features). Packers are software that compress binaries, keeping them executable. Similarly to the mutex related features, our features highlight if some particular packers are recognized.

Imported API (3917 features). Each software imports functions from common libraries or external files. The combination of imported functions can show similar behavior. We store this information as a vector with a Boolean value for each imported function, using the list of functions present in the training set as a taxonomy.

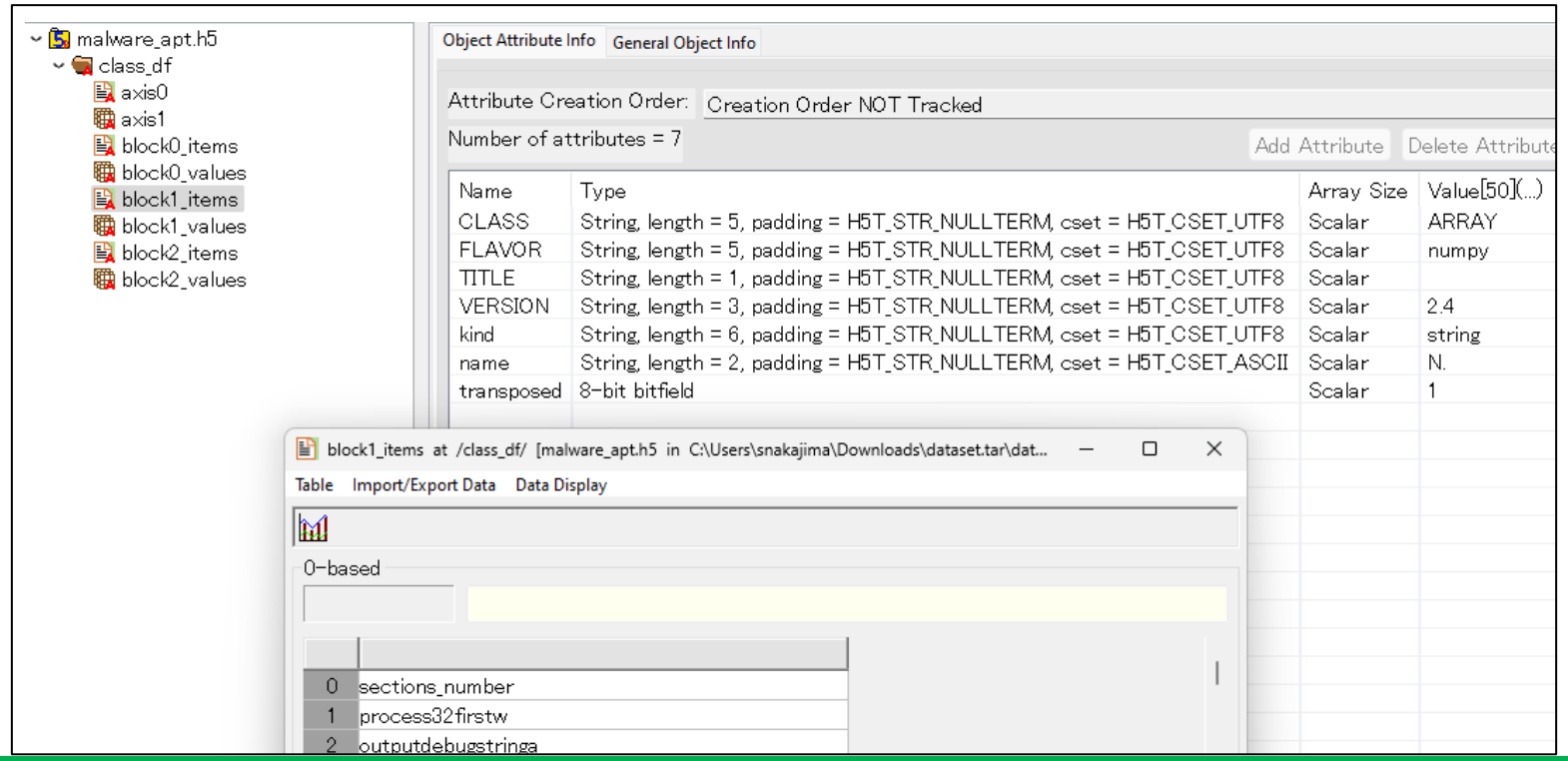
Buckets (98 features). Similar size in functions and directories can be a proof of similarity in the file structure and thus in the behavior. We observed that, usually, function size values range from 0 to 1,822 bytes, while directory size values range from 0 to 2,638 kbytes. To track these properties, both of them are subdivided into 49 buckets. Each bucket represents a range and counts the elements whose size is in the range. To choose the different ranges we observe the distribution of sizes and lengths in the training set, trying to form buckets that can better characterize the various classes.

CyberDefense

APT Malware dataset

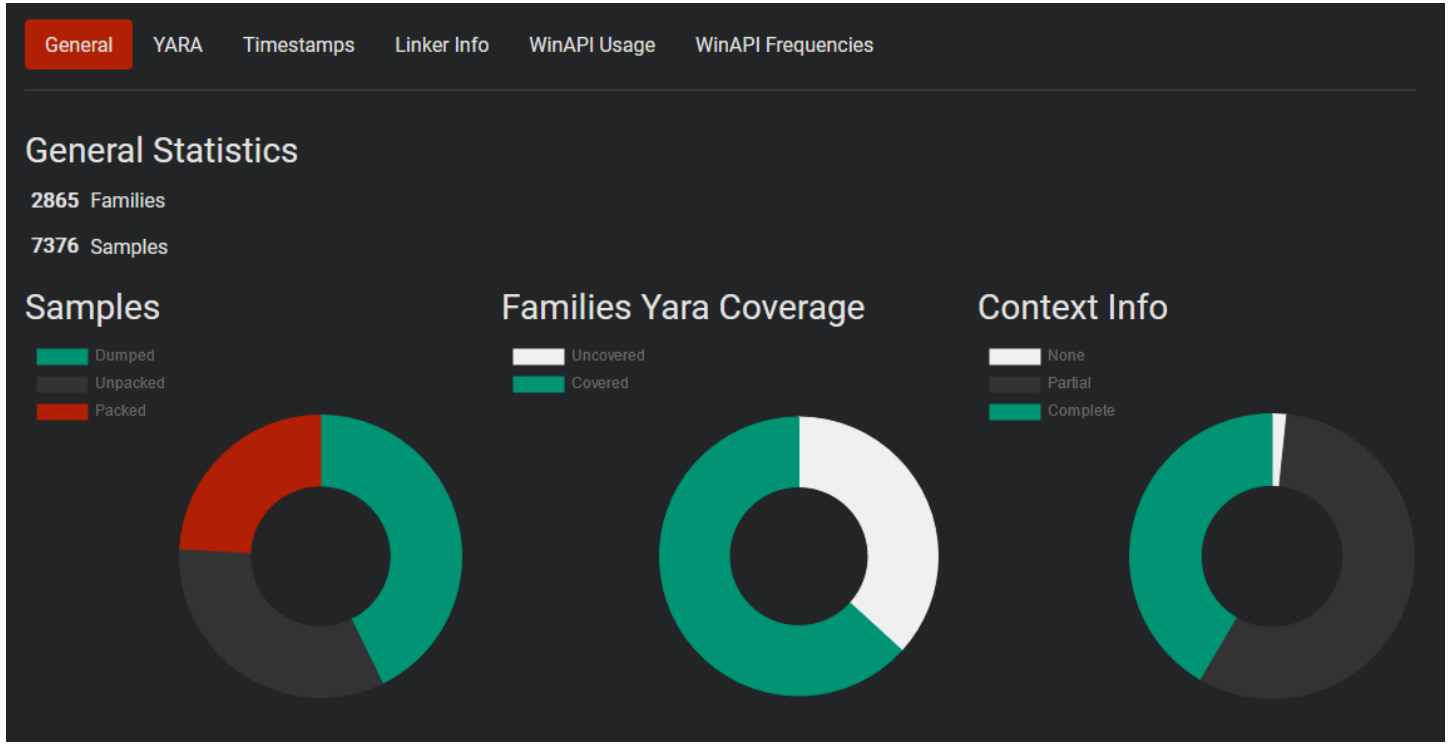
File Details

- *dataset.tar.gz* contains two hdf files containing features of APT-malware and normal malware
- *test_article.py* contains the code to test each implementation, included computation of some metrics
- *Checking_Result.py* contains the code to help computing quality and time metrics
- *ThresholdRandomForest.py* is a separate file containing all the methods to implement the functionalities of the first work
- *select** contain data obtained from our tests about best classes, best parameters and best columns to reproduce the published results



Malpedia

- ドイツのFraunhofer FKIE(通信・情報処理・人間工学研究所)が運営するマルウェア情報集約サービス
  - ユーザが有益なブログや検体、YARAルールを投稿
  - ログインするとマルウェアのダウンロードも可能



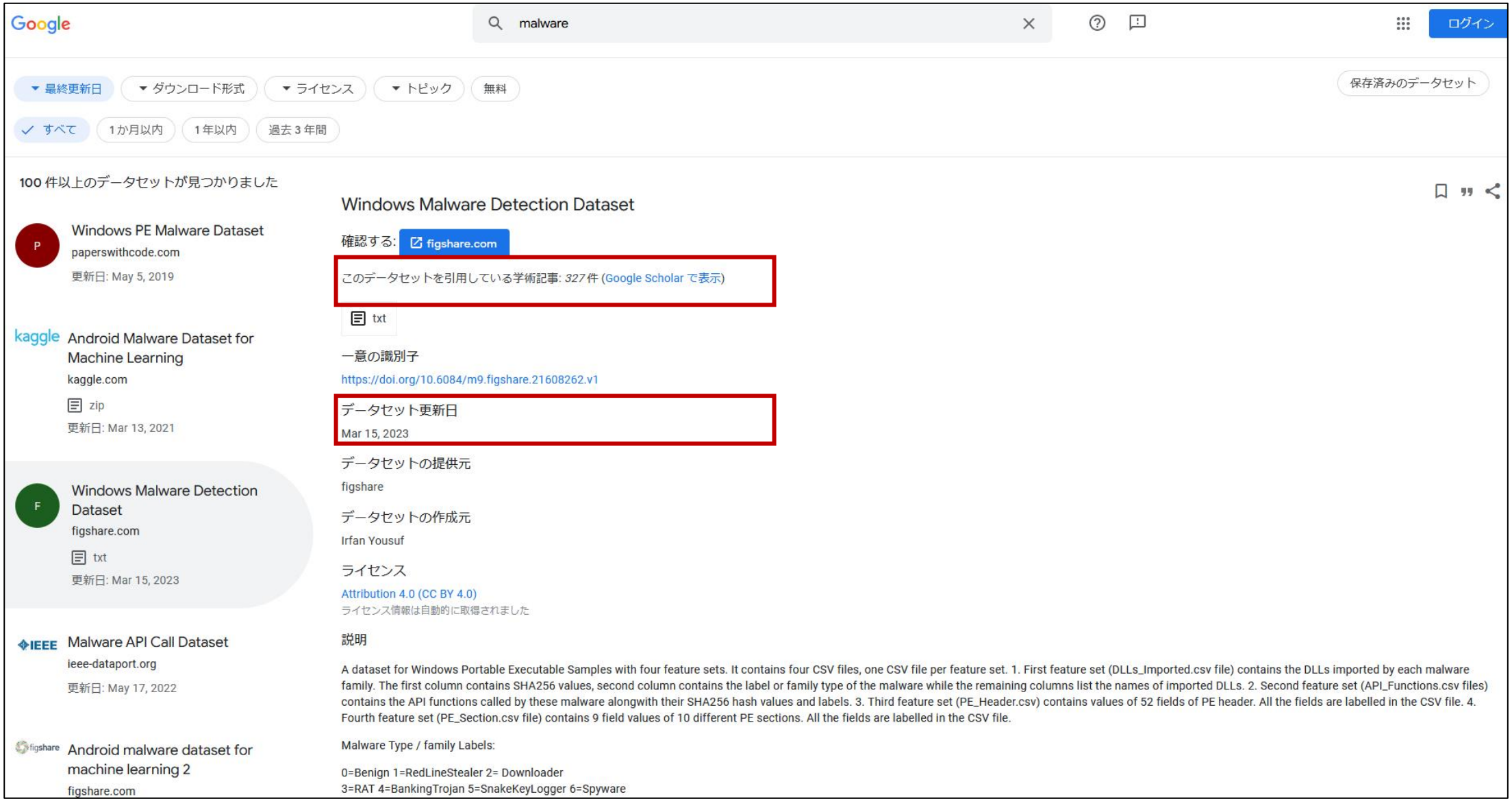
CyberDefense

Google Dataset Search

Google Dataset Search

CyberDefense

Google Dataset Search



CyberDefense

研究利用のためのマルウェアの収集方法

VirusTotal

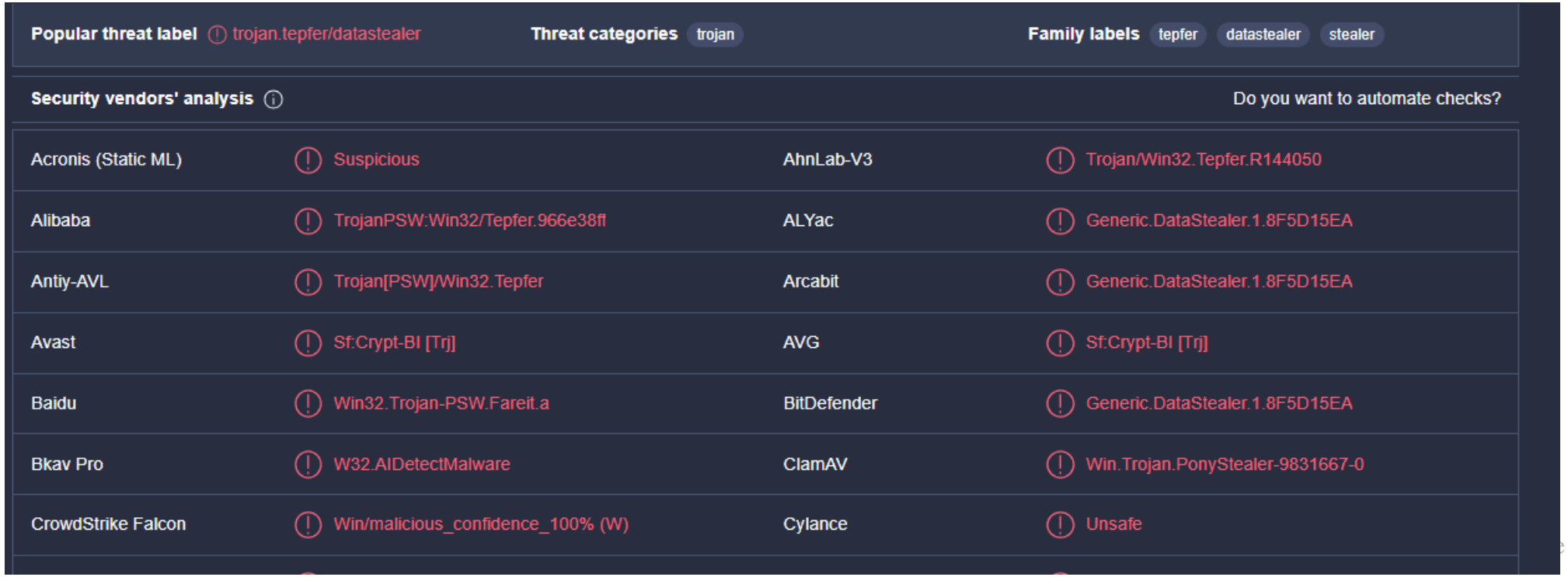
- 業界のデファクトスタンダードなサービス
- 有償契約ユーザのみダウンロード可能
  - APIの提供

弊社も販売パートナーです

CyberDefense

VirusTotal

- 検知名を正解ラベルを利用可能
  - 正解ラベルとして利用するべきかの議論はまた別の話・・・
- 最近はベンダーの検知名を丸めたラベルを公式が提供



Popular threat label trojan.tepfer/datastealer **Threat categories** trojan **Family labels** tepfer datastealer stealer

Security vendors' analysis Do you want to automate checks?

Acronis (Static ML)	Suspicious	AhnLab-V3	Trojan/Win32.Tepfer.R144050
Alibaba	TrojanPSW:Win32/Tepfer.966e38ff	ALYac	Generic.DataStealer.1.8F5D15EA
Antiy-AVL	Trojan[PSW]/Win32.Tepfer	Arcabit	Generic.DataStealer.1.8F5D15EA
Avast	Sf:Crypt-BI [Trj]	AVG	Sf:Crypt-BI [Trj]
Baidu	Win32.Trojan-PSW.Fareit.a	BitDefender	Generic.DataStealer.1.8F5D15EA
Bkav Pro	W32.AIDetectMalware	ClamAV	Win.Trojan.PonyStealer-9831667-0
CrowdStrike Falcon	Win/malicious_confidence_100% (W)	Cylance	Unsafe

erDefense

無償利用可能な公開サービス

マルウェア共有サービス/IoC共有サービス

マルウェアのサンプルやIoCを共有するためのサービスがいくつかあります。もっとも登録が多く有名なサービスはVirusTotalですが、有償アカウントのみダウンロードが可能であるため、ここでは無償で利用可能なサービスを紹介します。

- malpedia
- MalwareBazaar
- URLhaus
- ThreatFox
- MalShare
- VirusShare
- VirusBay
- Vx Vaul
- theZoo
- vx-underground

オンラインサンドボックス

無料で使用できるオンラインサンドボックスのサービスは本来マルウェアの挙動を解析するサービスですが、他者の投稿したマルウェアをダウンロードすることができます。

- Hybrid Analysis
- Any Run
- Triage
- Joe Sandbox
- cape
- Cuckoo Sandbox

CyberDefense

参考URL

- https://www.kaggle.com/competitions/malware-classification/overview
- https://github.com/elastic/ember
- https://paperswithcode.com/dataset/apt-malware
- https://github.com/GiuseppeLaurenza/I_F_Identifier
- https://malpedia.caad.fkie.fraunhofer.de/
- https://github.com/pinksawtooth/how_to_become_a_malware_analyst

CyberDefense

ONLY HUMANS CAN COUNTER HUMAN-DRIVEN THREATS

CyberDefense