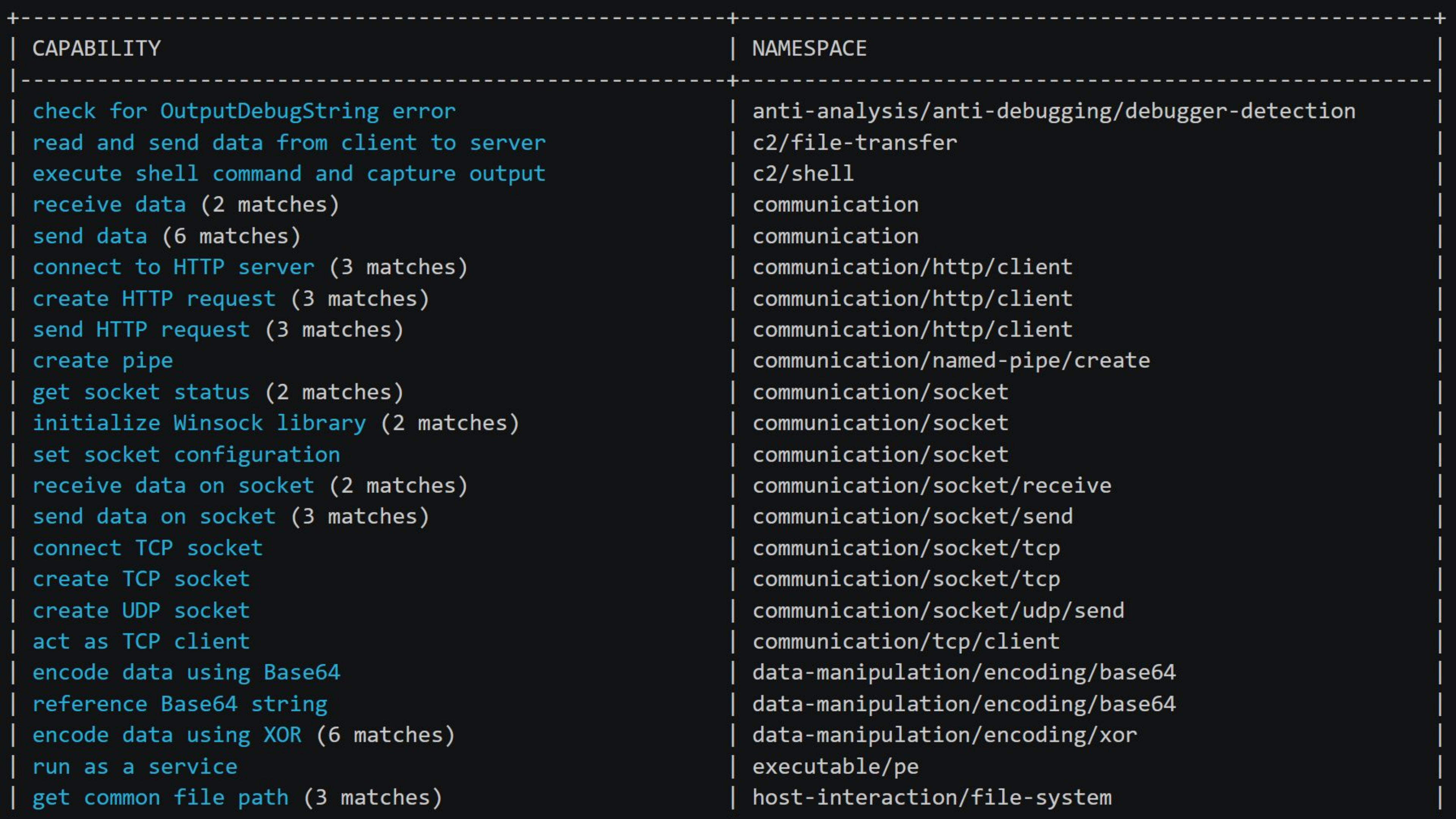
These slides may not be redistributed without written permission.

(BruCON also may not re-post or share these slides without written permission.)

MANDIANT
NOW PART OF Google Cloud

Detecting Malware Capabilities with capa

September 28, 2023



```
+-------------------------------------------------------+------------------------------------------------------+
| CAPABILITY                                            | NAMESPACE                                            |
|-------------------------------------------------------+------------------------------------------------------|
| check for OutputDebugString error                     | anti-analysis/anti-debugging/debugger-detection      |
| read and send data from client to server              | c2/file-transfer                                     |
| execute shell command and capture output              | c2/shell                                             |
| receive data (2 matches)                              | communication                                        |
| send data (6 matches)                                 | communication                                        |
| connect to HTTP server (3 matches)                    | communication/http/client                            |
| create HTTP request (3 matches)                       | communication/http/client                            |
| send HTTP request (3 matches)                         | communication/http/client                            |
| create pipe                                           | communication/named-pipe/create                      |
| get socket status (2 matches)                         | communication/socket                                 |
| initialize Winsock library (2 matches)                | communication/socket                                 |
| set socket configuration                              | communication/socket                                 |
| receive data on socket (2 matches)                    | communication/socket/receive                         |
| send data on socket (3 matches)                       | communication/socket/send                            |
| connect TCP socket                                    | communication/socket/tcp                             |
| create TCP socket                                     | communication/socket/tcp                             |
| create UDP socket                                     | communication/socket/udp/send                        |
| act as TCP client                                     | communication/tcp/client                             |
| encode data using Base64                              | data-manipulation/encoding/base64                    |
| reference Base64 string                               | data-manipulation/encoding/base64                    |
| encode data using XOR (6 matches)                     | data-manipulation/encoding/xor                       |
| run as a service                                      | executable/pe                                        |
| get common file path (3 matches)                      | host-interaction/file-system                         |
```

outline

introduction	01
motivation	02
capa tool	03
reading rules	04
writing rules	05
conclusion	06

agenda

35:00 min

Us talking

60:00 min

You working on labs

20:00 min

Lab reviews and discussions

5:00 min

Break and buffer ;)

01

introduction

about us

Willi Ballenthin
FLARE

Mike Hunhoff
FLARE

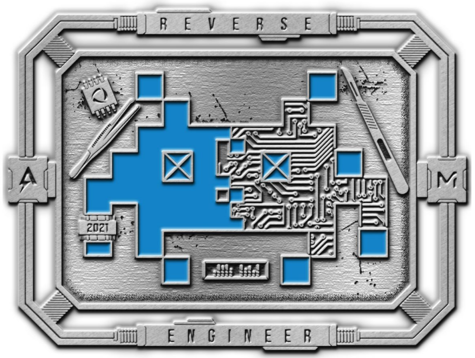
Moritz Raabe
FLARE

MANDIANT
NOW PART OF Google Cloud

the FLARE team

- Worldwide center of malware analysis excellence
- Open source development
- Education and knowledge sharing

https://flare-on.com



02

motivation

reality

Analysis shortcomings and gaps in the community

Forensic, intelligence, and malware analysts are faced with the challenge of understanding and triaging unknown programs on a daily basis

Experienced reverse engineers have trained eyes and brains that quickly recognize the most relevant parts of a program

- Can we codify and automate this knowledge?

building blocks

What features do we (humans) notice?

- Expert-driven system, not AI

Are the results easy to explain to a human?

- Tool must always be ready to “show its work”

How can we make this flexible and extendable?

result

With capa, we claim that some analysis conclusions are easy

Encodes patterns that have been recognized for decades

- Look for API calls, look for strings, …, and look for anomalies
- “When you see this and that, then we know other is happening”

Provides framework for

- Experts to express these patterns
- Analysts to recognize these patterns

03

capa tool

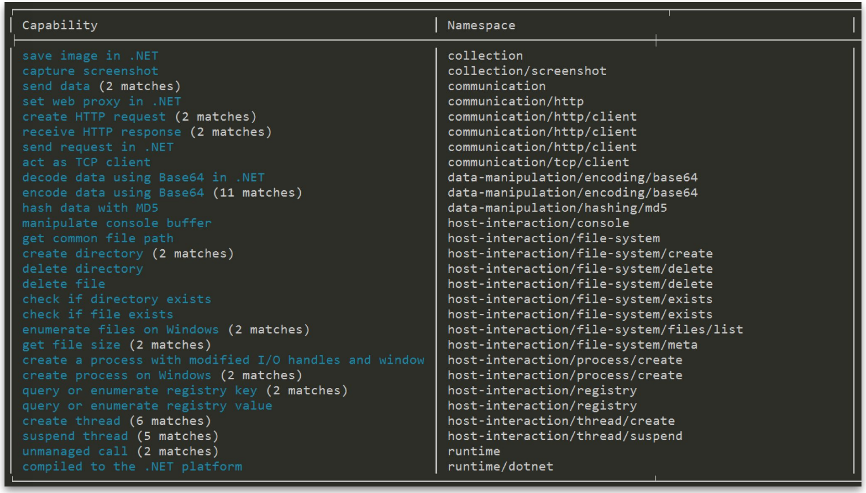
what is capa?

Tool to detect capabilities in executable files and shellcode

Powered by a collection of **over 800** rules matching features extracted from PE, ELF, .NET, and shellcode files

Two main components

- Code analysis engine
  - Extracts features from files, such as strings, disassembly, and control flow
- Logic engine
  - Finds combinations of features that are expressed in a common rule format



```
| Capability                                              | Namespace                                   |
|                                                         |                                             |
| save image in .NET                                      | collection                                  |
| capture screenshot                                      | collection/screenshot                       |
| send data (2 matches)                                   | communication                               |
| set web proxy in .NET                                   | communication/http                          |
| create HTTP request (2 matches)                         | communication/http/client                   |
| receive HTTP response (2 matches)                       | communication/http/client                   |
| send request in .NET                                    | communication/http/client                   |
| act as TCP client                                       | communication/tcp/client                    |
| decode data using Base64 in .NET                        | data-manipulation/encoding/base64           |
| encode data using Base64 (11 matches)                   | data-manipulation/encoding/base64           |
| hash data with MD5                                      | data-manipulation/hashing/md5               |
| manipulate console buffer                               | host-interaction/console                    |
| get common file path                                    | host-interaction/file-system                |
| create directory (2 matches)                            | host-interaction/file-system/create         |
| delete directory                                        | host-interaction/file-system/delete         |
| delete file                                             | host-interaction/file-system/delete         |
| check if directory exists                               | host-interaction/file-system/exists         |
| check if file exists                                    | host-interaction/file-system/exists         |
| enumerate files on Windows (2 matches)                  | host-interaction/file-system/files/list     |
| get file size (2 matches)                               | host-interaction/file-system/meta           |
| create a process with modified I/O handles and window   | host-interaction/process/create             |
| create process on Windows (2 matches)                   | host-interaction/process/create             |
| query or enumerate registry key (2 matches)             | host-interaction/registry                   |
| query or enumerate registry value                       | host-interaction/registry                   |
| create thread (6 matches)                               | host-interaction/thread/create              |
| suspend thread (5 matches)                              | host-interaction/thread/suspend             |
| unmanaged call (2 matches)                              | runtime                                     |
| compiled to the .NET platform                           | runtime/dotnet                              |
```

MANDIANT
NOW PART OF Google Cloud

usage

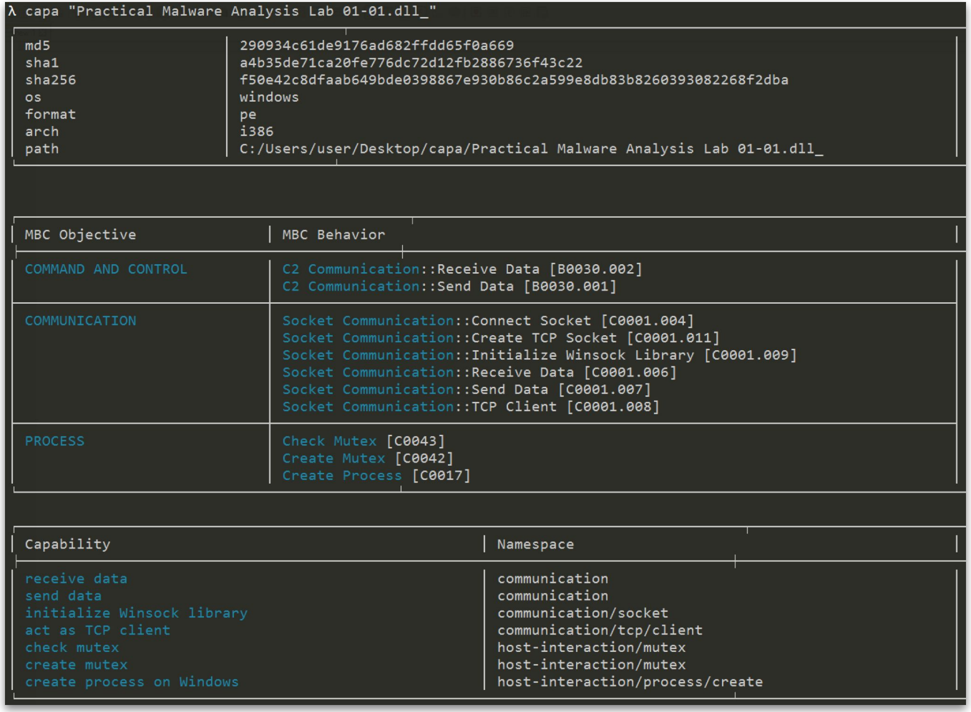
Download latest release of standalone tool from GitHub

- Windows
- Linux
- macOS

Contains all source code, Python interpreter, and associated resources (i.e. rules) needed to run capa

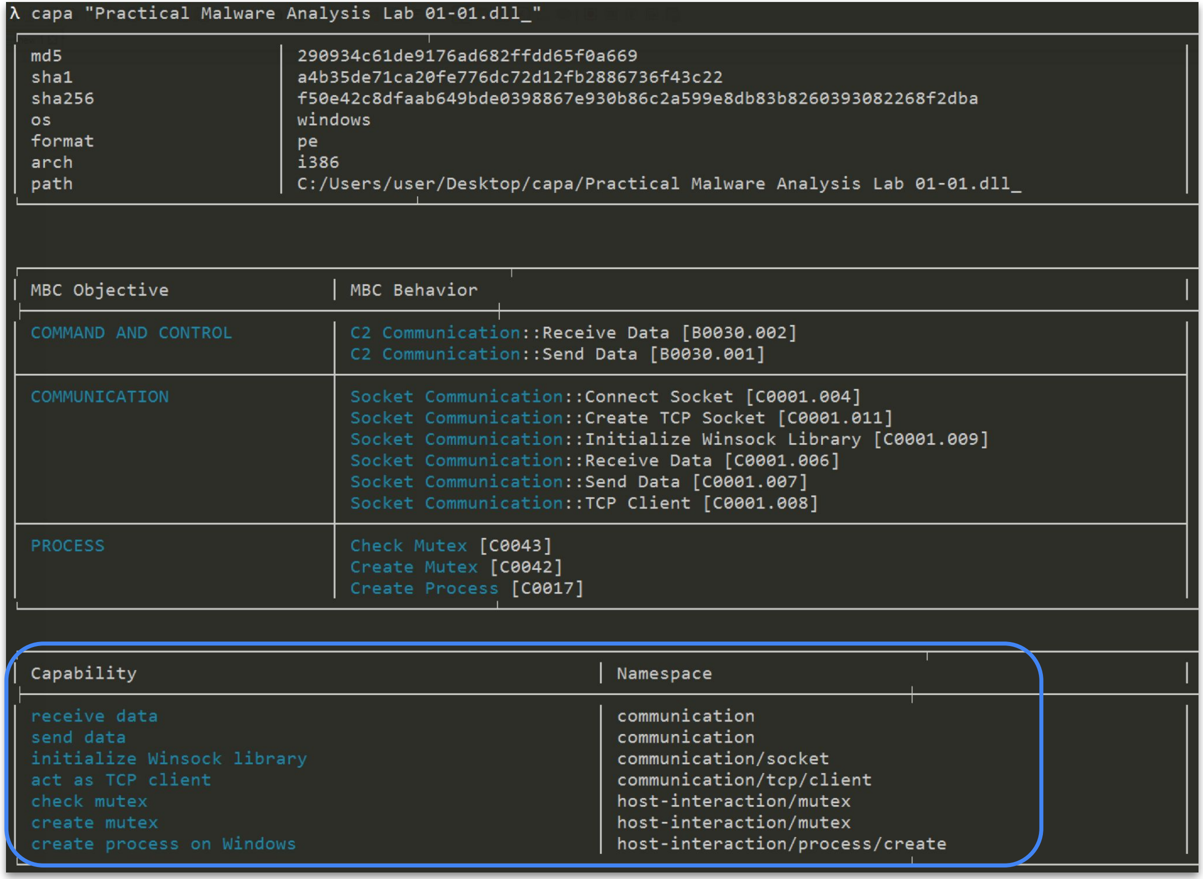
Run via command line (--help to view supported flags)

Multiple output formats



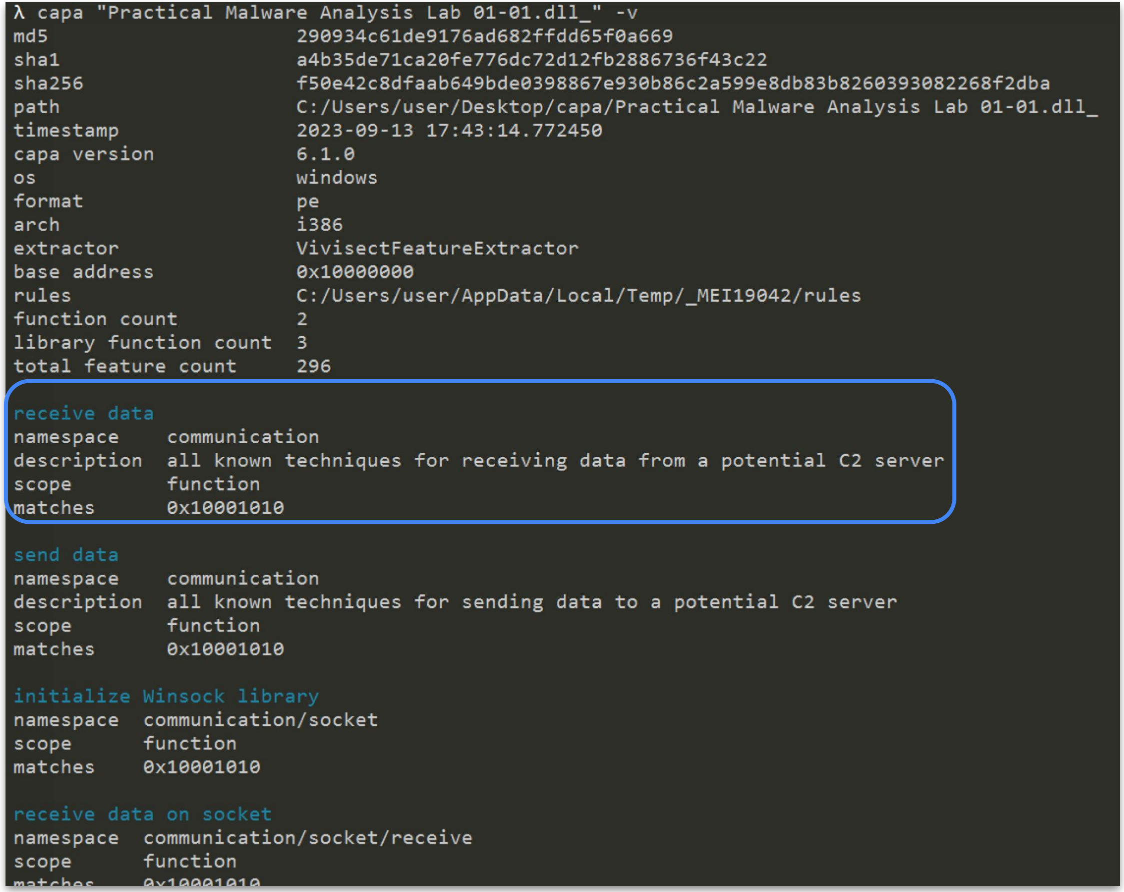
```
λ capa "Practical Malware Analysis Lab 01-01.dll_"
┌─────────────────────────────────────────────────────────────────────────────────────────────┐
│ md5                         │ 290934c61de9176ad682ffdd65f0a669                              │
│ sha1                        │ a4b35de71ca20fe776dc72d12fb2886736f43c22                      │
│ sha256                      │ f50e42c8dfaab649bde0398867e930b86c2a599e8db83b8260393082268f2dba │
│ os                          │ windows                                                       │
│ format                      │ pe                                                            │
│ arch                        │ i386                                                          │
│ path                        │ C:/Users/user/Desktop/capa/Practical Malware Analysis Lab 01-01.dll_ │
└─────────────────────────────────────────────────────────────────────────────────────────────┘

┌─────────────────────────────────────────────────────────────────────────────────────────────┐
│ MBC Objective          │ MBC Behavior                                                       │
├─────────────────────────────────────────────────────────────────────────────────────────────┤
│ COMMAND AND CONTROL    │ C2 Communication::Receive Data [B0030.002]                         │
│                        │ C2 Communication::Send Data [B0030.001]                            │
├─────────────────────────────────────────────────────────────────────────────────────────────┤
│ COMMUNICATION          │ Socket Communication::Connect Socket [C0001.004]                   │
│                        │ Socket Communication::Create TCP Socket [C0001.011]                │
│                        │ Socket Communication::Initialize Winsock Library [C0001.009]       │
│                        │ Socket Communication::Receive Data [C0001.006]                     │
│                        │ Socket Communication::Send Data [C0001.007]                        │
│                        │ Socket Communication::TCP Client [C0001.008]                       │
├─────────────────────────────────────────────────────────────────────────────────────────────┤
│ PROCESS                │ Check Mutex [C0043]                                                │
│                        │ Create Mutex [C0042]                                               │
│                        │ Create Process [C0017]                                             │
└─────────────────────────────────────────────────────────────────────────────────────────────┘

┌─────────────────────────────────────────────────────────────────────────────────────────────┐
│ Capability                                │ Namespace                                       │
├─────────────────────────────────────────────────────────────────────────────────────────────┤
│ receive data                              │ communication                                   │
│ send data                                 │ communication                                   │
│ initialize Winsock library                │ communication/socket                            │
│ act as TCP client                         │ communication/tcp/client                        │
│ check mutex                               │ host-interaction/mutex                          │
│ create mutex                              │ host-interaction/mutex                          │
│ create process on Windows                 │ host-interaction/process/create                 │
└─────────────────────────────────────────────────────────────────────────────────────────────┘
```

capa.exe /path/to/file



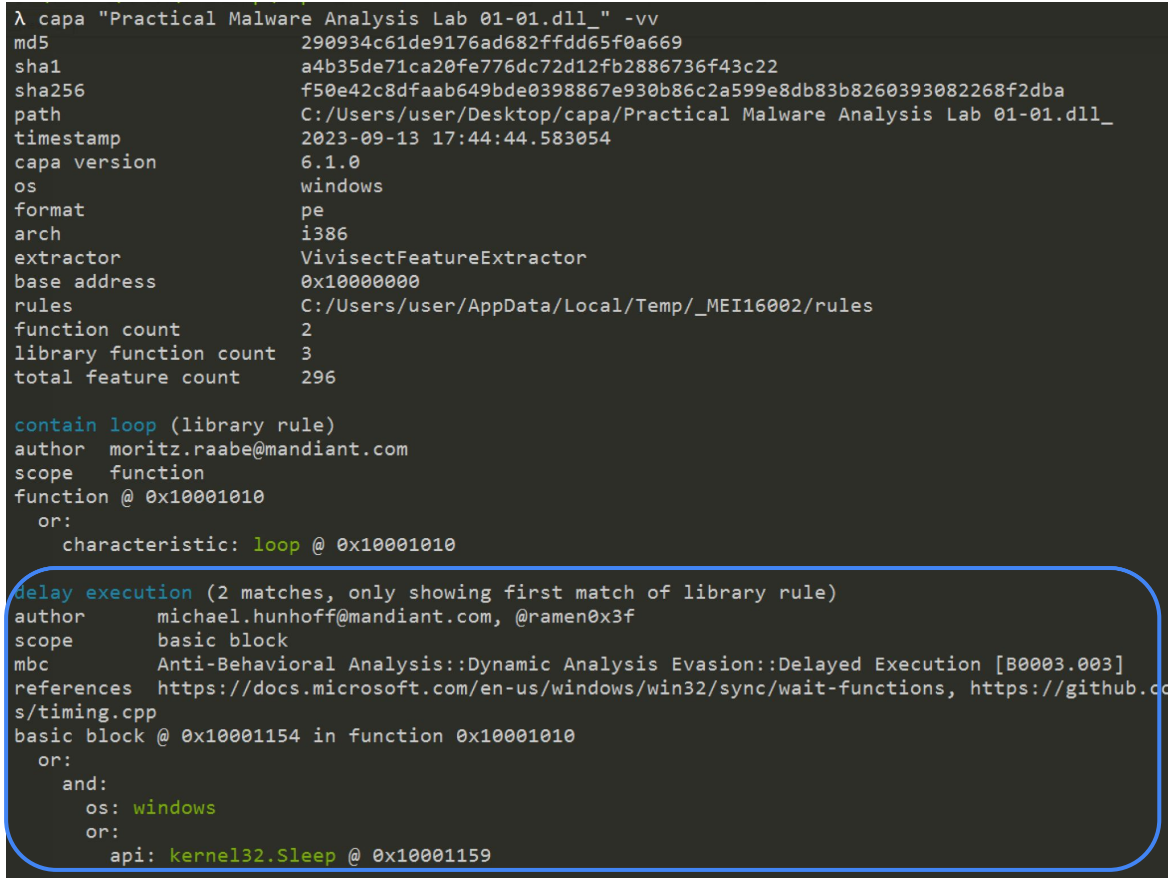
```
λ capa "Practical Malware Analysis Lab 01-01.dll_"
┌────────────────────────┬──────────────────────────────────────────────────────────────────┐
│ md5                    │ 290934c61de9176ad682ffdd65f0a669                                 │
│ sha1                   │ a4b35de71ca20fe776dc72d12fb2886736f43c22                         │
│ sha256                 │ f50e42c8dfaab649bde0398867e930b86c2a599e8db83b8260393082268f2dba │
│ os                     │ windows                                                          │
│ format                 │ pe                                                               │
│ arch                   │ i386                                                             │
│ path                   │ C:/Users/user/Desktop/capa/Practical Malware Analysis Lab 01-01.dll_ │
└────────────────────────┴──────────────────────────────────────────────────────────────────┘

┌─────────────────────────────┬────────────────────────────────────────────────────────────┐
│ MBC Objective               │ MBC Behavior                                               │
├─────────────────────────────┼────────────────────────────────────────────────────────────┤
│ COMMAND AND CONTROL         │ C2 Communication::Receive Data [B0030.002]                 │
│                             │ C2 Communication::Send Data [B0030.001]                    │
├─────────────────────────────┼────────────────────────────────────────────────────────────┤
│ COMMUNICATION               │ Socket Communication::Connect Socket [C0001.004]           │
│                             │ Socket Communication::Create TCP Socket [C0001.011]        │
│                             │ Socket Communication::Initialize Winsock Library [C0001.009] │
│                             │ Socket Communication::Receive Data [C0001.006]             │
│                             │ Socket Communication::Send Data [C0001.007]                │
│                             │ Socket Communication::TCP Client [C0001.008]               │
├─────────────────────────────┼────────────────────────────────────────────────────────────┤
│ PROCESS                     │ Check Mutex [C0043]                                        │
│                             │ Create Mutex [C0042]                                       │
│                             │ Create Process [C0017]                                     │
└─────────────────────────────┴────────────────────────────────────────────────────────────┘

┌──────────────────────────────────────────────────┬───────────────────────────────────────┐
│ Capability                                       │ Namespace                             │
├──────────────────────────────────────────────────┼───────────────────────────────────────┤
│ receive data                                     │ communication                         │
│ send data                                        │ communication                         │
│ initialize Winsock library                       │ communication/socket                  │
│ act as TCP client                                │ communication/tcp/client              │
│ check mutex                                      │ host-interaction/mutex                │
│ create mutex                                     │ host-interaction/mutex                │
│ create process on Windows                        │ host-interaction/process/create       │
└──────────────────────────────────────────────────┴───────────────────────────────────────┘
```

capa.exe /path/to/file -v



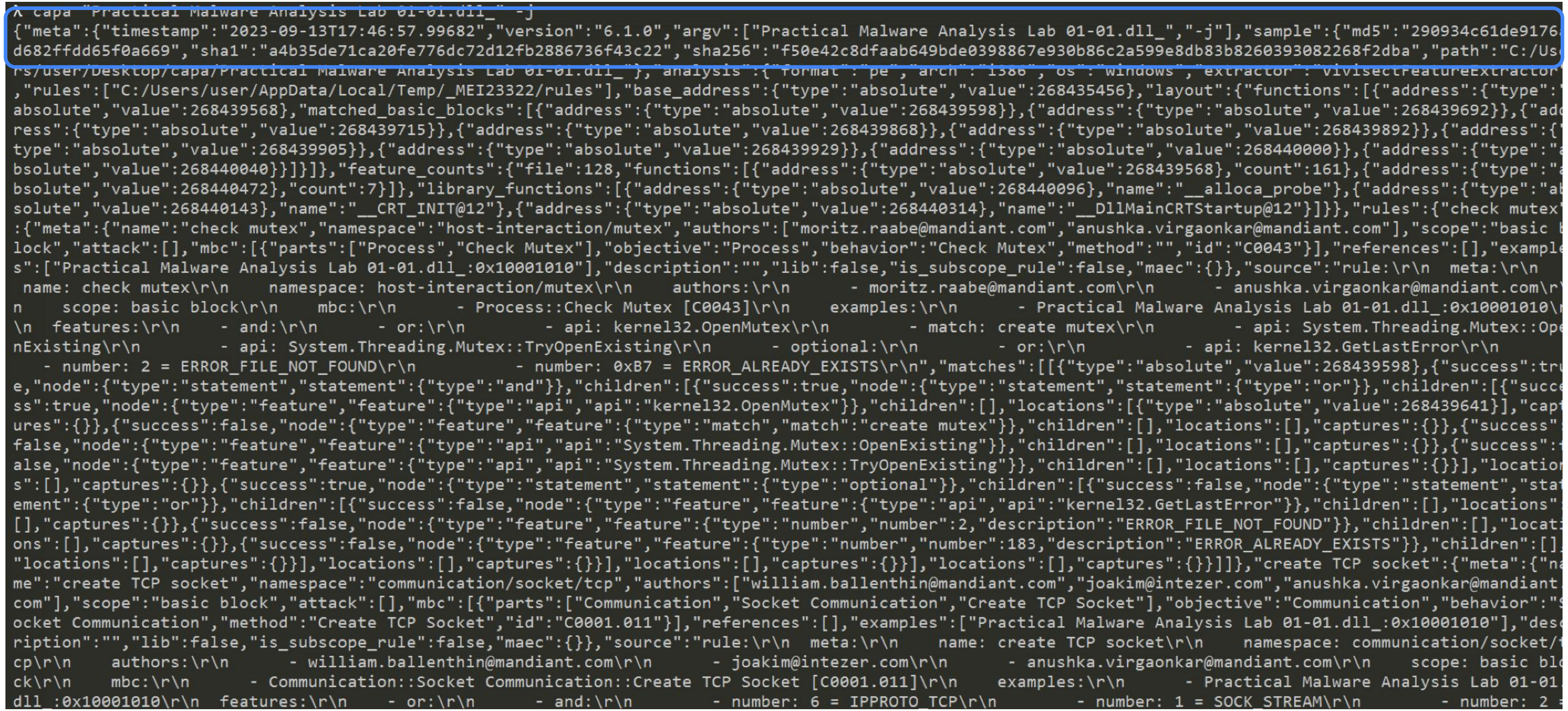
```
λ capa "Practical Malware Analysis Lab 01-01.dll_" -v
md5                     290934c61de9176ad682ffdd65f0a669
sha1                    a4b35de71ca20fe776dc72d12fb2886736f43c22
sha256                  f50e42c8dfaab649bde0398867e930b86c2a599e8db83b8260393082268f2dba
path                    C:/Users/user/Desktop/capa/Practical Malware Analysis Lab 01-01.dll_
timestamp               2023-09-13 17:43:14.772450
capa version            6.1.0
os                      windows
format                  pe
arch                    i386
extractor               VivisectFeatureExtractor
base address            0x10000000
rules                   C:/Users/user/AppData/Local/Temp/_MEI19042/rules
function count          2
library function count  3
total feature count     296

receive data
namespace    communication
description  all known techniques for receiving data from a potential C2 server
scope        function
matches      0x10001010

send data
namespace    communication
description  all known techniques for sending data to a potential C2 server
scope        function
matches      0x10001010

initialize Winsock library
namespace  communication/socket
scope      function
matches    0x10001010

receive data on socket
namespace  communication/socket/receive
scope      function
matches    0x10001010
```

capa.exe /path/to/file -vv



```
λ capa "Practical Malware Analysis Lab 01-01.dll_" -vv
md5                     290934c61de9176ad682ffdd65f0a669
sha1                    a4b35de71ca20fe776dc72d12fb2886736f43c22
sha256                  f50e42c8dfaab649bde0398867e930b86c2a599e8db83b8260393082268f2dba
path                    C:/Users/user/Desktop/capa/Practical Malware Analysis Lab 01-01.dll_
timestamp               2023-09-13 17:44:44.583054
capa version            6.1.0
os                      windows
format                  pe
arch                    i386
extractor               VivisectFeatureExtractor
base address            0x10000000
rules                   C:/Users/user/AppData/Local/Temp/_MEI16002/rules
function count          2
library function count  3
total feature count     296

contain loop (library rule)
author  moritz.raabe@mandiant.com
scope   function
function @ 0x10001010
  or:
    characteristic: loop @ 0x10001010

delay execution (2 matches, only showing first match of library rule)
author      michael.hunhoff@mandiant.com, @ramen0x3f
scope       basic block
mbc         Anti-Behavioral Analysis::Dynamic Analysis Evasion::Delayed Execution [B0003.003]
references  https://docs.microsoft.com/en-us/windows/win32/sync/wait-functions, https://github.c
s/timing.cpp
basic block @ 0x10001154 in function 0x10001010
  or:
    and:
      os: windows
      or:
        api: kernel32.Sleep @ 0x10001159
```

capa.exe /path/to/file -j [> /path/to/json/file]



```
λ capa "Practical Malware Analysis Lab 01-01.dll_" -j
{"meta":{"timestamp":"2023-09-13T17:46:57.99682","version":"6.1.0","argv":["Practical Malware Analysis Lab 01-01.dll_","-j"],"sample":{"md5":"290934c61de9176
d682ffdd65f0a669","sha1":"a4b35de71ca20fe776dc72d12fb2886736f43c22","sha256":"f50e42c8dfaab649bde0398867e930b86c2a599e8db83b8260393082268f2dba","path":"C:/Us
rs/user/Desktop/capa/Practical Malware Analysis Lab 01-01.dll_"},"analysis":{"format":"pe","arch":"i386","os":"windows","extractor":"VivisectFeatureExtractor"
,"rules":["C:/Users/user/AppData/Local/Temp/_MEI23322/rules"],"base_address":{"type":"absolute","value":268435456},"layout":{"functions":[{"address":{"type":"
absolute","value":268439568},"matched_basic_blocks":[{"address":{"type":"absolute","value":268439598}},{"address":{"type":"absolute","value":268439692}},{"add
ress":{"type":"absolute","value":268439715}},{"address":{"type":"absolute","value":268439868}},{"address":{"type":"absolute","value":268439892}},{"address":{"
type":"absolute","value":268439905}},{"address":{"type":"absolute","value":268439929}},{"address":{"type":"absolute","value":268440000}},{"address":{"type":"a
bsolute","value":268440040}}]}]},"feature_counts":{"file":128,"functions":[{"address":{"type":"absolute","value":268439568},"count":161},{"address":{"type":"a
bsolute","value":268440472},"count":7}]},"library_functions":[{"address":{"type":"absolute","value":268440096},"name":"__alloca_probe"},{"address":{"type":"ab
solute","value":268440143},"name":"__CRT_INIT@12"},{"address":{"type":"absolute","value":268440314},"name":"__DllMainCRTStartup@12"}]}},"rules":{"check mutex"
:{"meta":{"name":"check mutex","namespace":"host-interaction/mutex","authors":["moritz.raabe@mandiant.com","anushka.virgaonkar@mandiant.com"],"scope":"basic b
lock","attack":[],"mbc":[{"parts":["Process","Check Mutex"],"objective":"Process","behavior":"Check Mutex","method":"","id":"C0043"}],"references":[],"example
s":["Practical Malware Analysis Lab 01-01.dll_:0x10001010"],"description":"","lib":false,"is_subscope_rule":false,"maec":{}},"source":"rule:\r\n  meta:\r\n
 name: check mutex\r\n    namespace: host-interaction/mutex\r\n    authors:\r\n      - moritz.raabe@mandiant.com\r\n      - anushka.virgaonkar@mandiant.com\r\
n    scope: basic block\r\n    mbc:\r\n      - Process::Check Mutex [C0043]\r\n    examples:\r\n      - Practical Malware Analysis Lab 01-01.dll_:0x10001010\r
\n  features:\r\n    - and:\r\n      - or:\r\n        - api: kernel32.OpenMutex\r\n        - match: create mutex\r\n        - api: System.Threading.Mutex::Ope
nExisting\r\n        - api: System.Threading.Mutex::TryOpenExisting\r\n      - optional:\r\n        - or:\r\n          - api: kernel32.GetLastError\r\n
  - number: 2 = ERROR_FILE_NOT_FOUND\r\n          - number: 0xB7 = ERROR_ALREADY_EXISTS\r\n","matches":[[{"type":"absolute","value":268439598},{"success":tru
e,"node":{"type":"statement","statement":{"type":"and"}},"children":[{"success":true,"node":{"type":"statement","statement":{"type":"or"}},"children":[{"succe
ss":true,"node":{"type":"feature","feature":{"type":"api","api":"kernel32.OpenMutex"}},"children":[],"locations":[{"type":"absolute","value":268439641}],"capt
ures":{}},{"success":false,"node":{"type":"feature","feature":{"type":"match","match":"create mutex"}},"children":[],"locations":[],"captures":{}},{"success":
false,"node":{"type":"feature","feature":{"type":"api","api":"System.Threading.Mutex::OpenExisting"}},"children":[],"locations":[],"captures":{}},{"success":f
alse,"node":{"type":"feature","feature":{"type":"api","api":"System.Threading.Mutex::TryOpenExisting"}},"children":[],"locations":[],"captures":{}}],"locatio
s":[],"captures":{}},{"success":true,"node":{"type":"statement","statement":{"type":"optional"}},"children":[{"success":false,"node":{"type":"statement","stat
ement":{"type":"or"}},"children":[{"success":false,"node":{"type":"feature","feature":{"type":"api","api":"kernel32.GetLastError"}},"children":[],"locations":
[],"captures":{}},{"success":false,"node":{"type":"feature","feature":{"type":"number","number":2,"description":"ERROR_FILE_NOT_FOUND"}},"children":[],"locati
ons":[],"captures":{}},{"success":false,"node":{"type":"feature","feature":{"type":"number","number":183,"description":"ERROR_ALREADY_EXISTS"}},"children":[],
"locations":[],"captures":{}}],"locations":[],"captures":{}}],"locations":[],"captures":{}}]]},"create TCP socket":{"meta":{"na
me":"create TCP socket","namespace":"communication/socket/tcp","authors":["william.ballenthin@mandiant.com","joakim@intezer.com","anushka.virgaonkar@mandiant.
com"],"scope":"basic block","attack":[],"mbc":[{"parts":["Communication","Socket Communication","Create TCP Socket"],"objective":"Communication","behavior":"S
ocket Communication","method":"Create TCP Socket","id":"C0001.011"}],"references":[],"examples":["Practical Malware Analysis Lab 01-01.dll_:0x10001010"],"desc
ription":"","lib":false,"is_subscope_rule":false,"maec":{}},"source":"rule:\r\n  meta:\r\n    name: create TCP socket\r\n    namespace: communication/socket/t
cp\r\n    authors:\r\n      - william.ballenthin@mandiant.com\r\n      - joakim@intezer.com\r\n      - anushka.virgaonkar@mandiant.com\r\n    scope: basic blo
ck\r\n    mbc:\r\n      - Communication::Socket Communication::Create TCP Socket [C0001.011]\r\n    examples:\r\n      - Practical Malware Analysis Lab 01-01.
dll_:0x10001010\r\n  features:\r\n    - or:\r\n      - and:\r\n        - number: 6 = IPPROTO_TCP\r\n        - number: 1 = SOCK_STREAM\r\n        - number: 2 =
```

why use capa?

Triage malware samples without deep, manual analysis

Identify malware samples via “capability signature”

Compute similarity among samples

Guide advanced reverse engineering

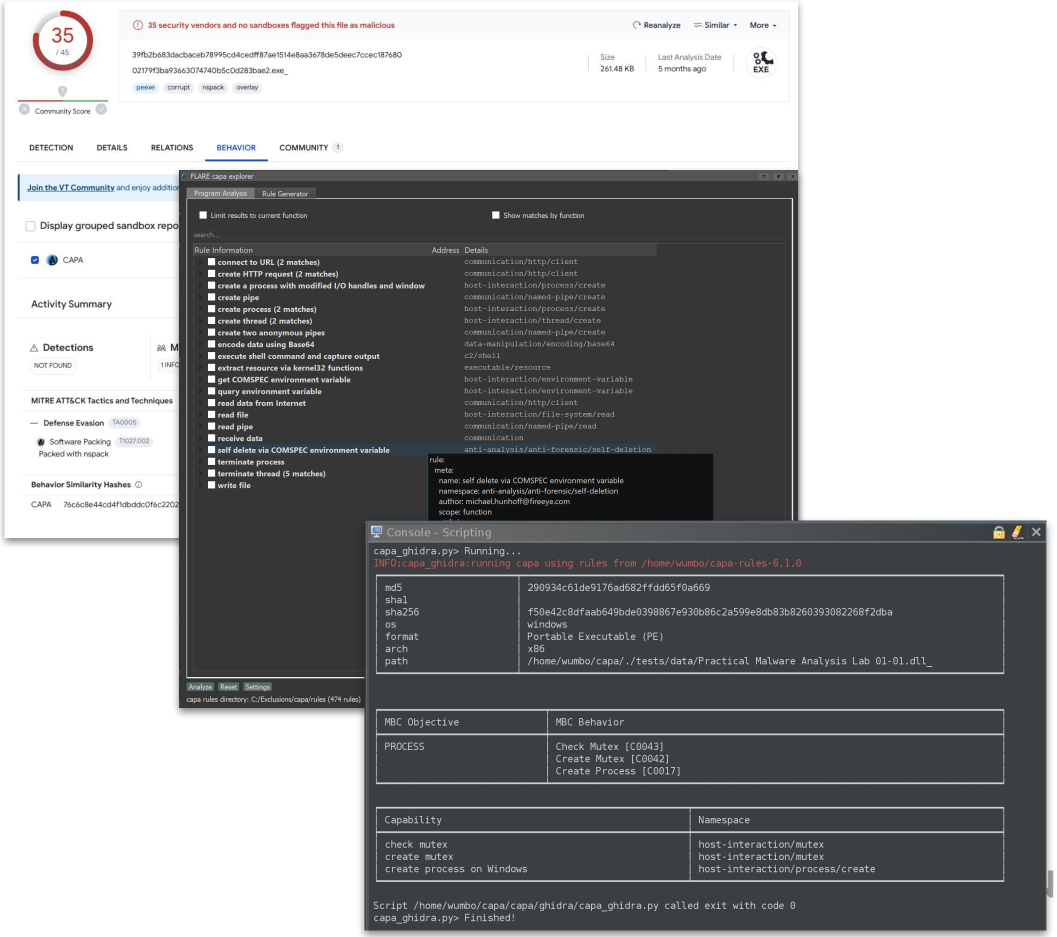
- Pivot to most interesting areas of code

integrations

Available in VirusTotal

Integrated with popular analysis tools including

- IDA Pro
- Binary Ninja
- Ghidra



35 / 45

Community Score

35 security vendors and no sandboxes flagged this file as malicious

39fb2b683dacbaceb78995cd4cedff87ae1514e8aa3678de5deec7ccec187680

02179f3ba93663074740b5c0d283bae2.exe_

peexe corrupt nspack overlay

Size 261.48 KB

Last Analysis Date 5 months ago

Join the VT Community and enjoy additio

□ Display grouped sandbox repo

☑ CAPA

Activity Summary

Detections

NOT FOUND

M

1 INFO

MITRE ATT&CK Tactics and Techniques

— Defense Evasion TA0005

Software Packing T1027.002

Packed with nspack

Behavior Similarity Hashes

CAPA 76c6c8e44cd4f1dbddc0f6c2202

FLARE capa explorer

□ Limit results to current function

□ Show matches by function

search...

Rule Information	Address	Details
□ connect to URL (2 matches)		communication/http/client
□ create HTTP request (2 matches)		communication/http/client
□ create a process with modified I/O handles and window		host-interaction/process/create
□ create pipe		communication/named-pipe/create
□ create process (2 matches)		host-interaction/process/create
□ create thread (2 matches)		host-interaction/thread/create
□ create two anonymous pipes		communication/named-pipe/create
□ encode data using Base64		data-manipulation/encoding/base64
□ execute shell command and capture output		c2/shell
□ extract resource via kernel32 functions		executable/resource
□ get COMSPEC environment variable		host-interaction/environment-variable
□ query environment variable		host-interaction/environment-variable
□ read data from Internet		communication/http/client
□ read file		host-interaction/file-system/read
□ read pipe		communication/named-pipe/read
□ receive data		communication
□ self delete via COMSPEC environment variable		anti-analysis/anti-forensic/self-deletion
□ terminate process		
□ terminate thread (5 matches)		
□ write file		

```
rule:
  meta:
    name: self delete via COMSPEC environment variable
    namespace: anti-analysis/anti-forensic/self-deletion
    author: michael.hunhoff@fireeye.com
    scope: function
```

capa rules directory: C:/Exclusions/capa/rules (474 rules)

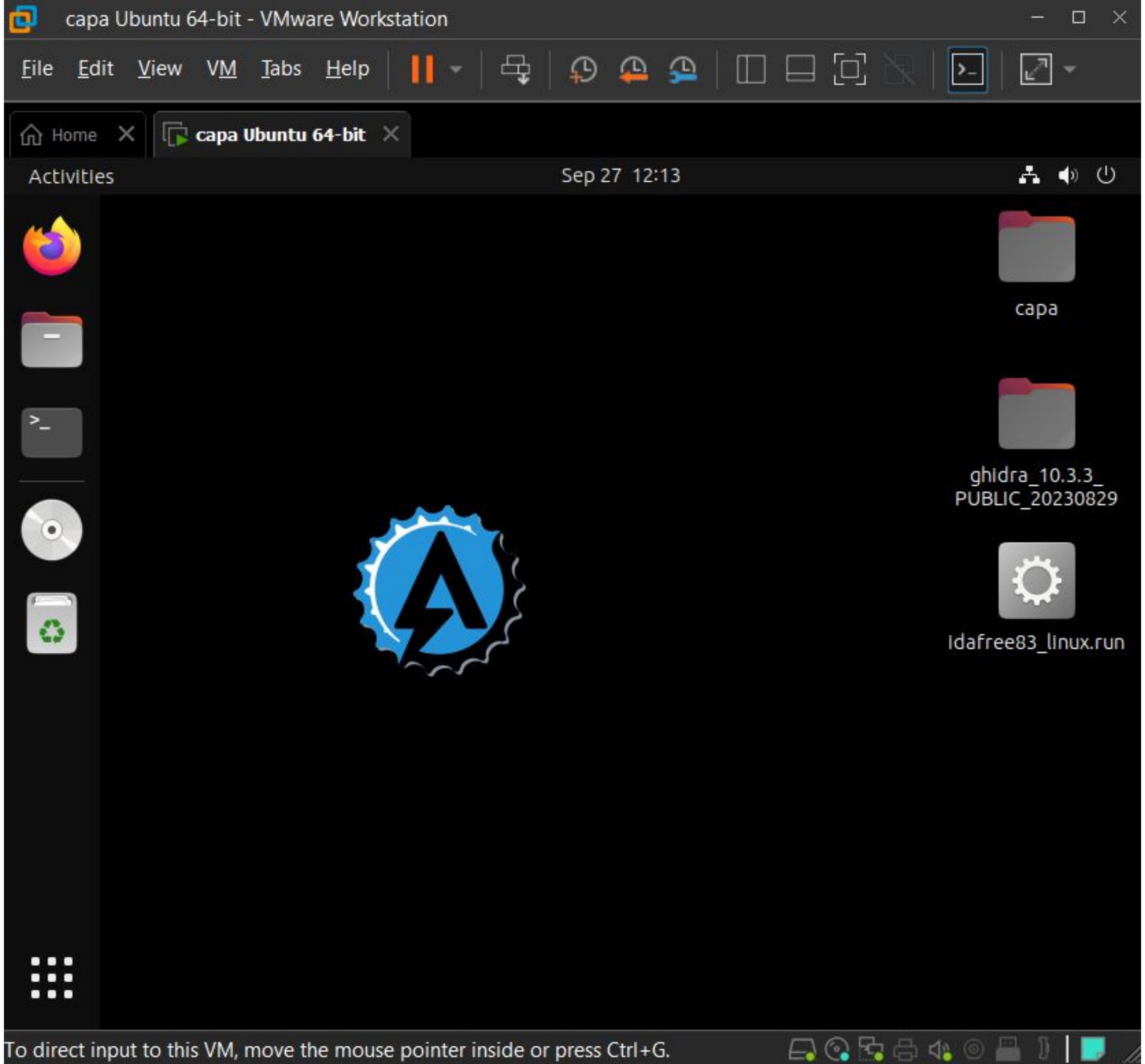
Console - Scripting

```
capa_ghidra.py> Running...
INFO:capa_ghidra:running capa using rules from /home/wumbo/capa-rules-6.1.0

 md5       290934c61de9176ad682ffdd65f0a669
 sha1
 sha256    f50e42c8dfaab649bde0398867e930b86c2a599e8db83b8260393082268f2dba
 os        windows
 format    Portable Executable (PE)
 arch      x86
 path      /home/wumbo/capa/./tests/data/Practical Malware Analysis Lab 01-01.dll_


 MBC Objective          MBC Behavior

 PROCESS                Check Mutex [C0043]
                        Create Mutex [C0042]
                        Create Process [C0017]


 Capability                          Namespace

 check mutex                         host-interaction/mutex
 create mutex                        host-interaction/mutex
 create process on Windows           host-interaction/process/create

Script /home/wumbo/capa/capa/ghidra/capa_ghidra.py called exit with code 0
capa_ghidra.py> Finished!
```

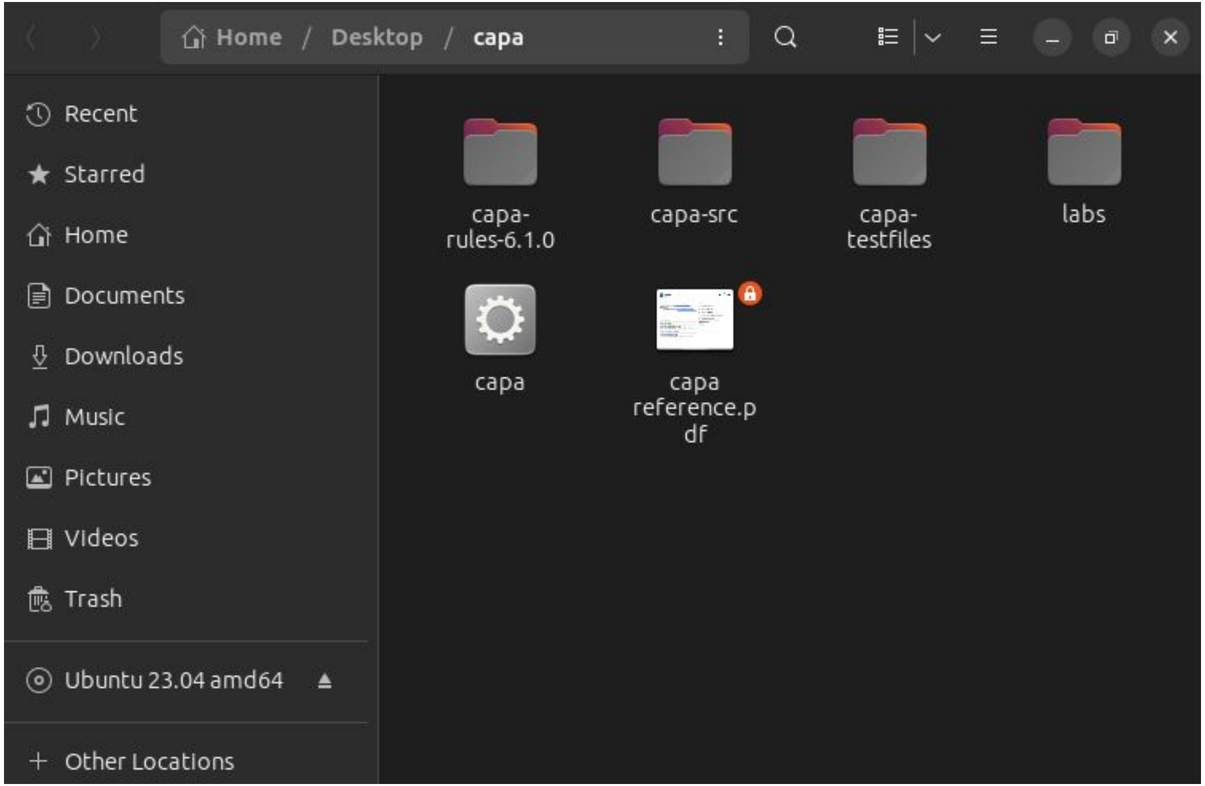
lab setup

1. Google Drive.
    a. https://drive.google.com/drive/folders/1vRkj4nJ6SZuFnOANHD06V416keUN5_sC
    b. contains all the following content.

2. .zip archive. password: infected.
    a. capa.exe, capa-rules, capa-testfiles, labs
    b. *use this if you have your own dev/analysis environment.*
3. VMware Workstation 14.x compatible virtual machine. Ubuntu 23.04 guest OS. user/password.
    a. capa.exe, capa-rules, capa-testfiles, lab
    b. analysis tools: ghidra, ida-free
    c. development tools: vscode
    d. *use this if you need a pre-built dev/analysis environment.*



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


Home / Desktop / capa
Recent
Starred
Home
Documents
Downloads
Music
Pictures
Videos
Trash
Ubuntu 23.04 amd64
+ Other Locations
capa-rules-6.1.0
capa-src
capa-testfiles
labs
capa
capa reference.pdf

lab one
using capa

lab one using capa

Use capa to answer the following questions

a)

1. Which of the file(s) is a Windows PE? Linux ELF? Windows .NET PE?
2. Which of the file(s) is packed? Using what packer?

b)

1. How many functions does capa identify in the packed file? How many features?
2. How many functions does capa identify in the unpacked file? How many features? (Hint: unpack the file using **upx -d**)

c)

1. Which file(s) use MITRE ATT&CK persistence tactics? What is the specific persistence technique(s)?
2. Which file(s) create a mutex? Which function address is responsible for creating the mutex?

bonus)

1. Execute capa to generate JSON-formatted output for the unpacked file and use **jq** to display the address of any function that has a match (Hint: all of the data that you need is stored in the **meta** field. Use the command **jq ".meta" /path/to/json** to display the contents of the **meta** field).

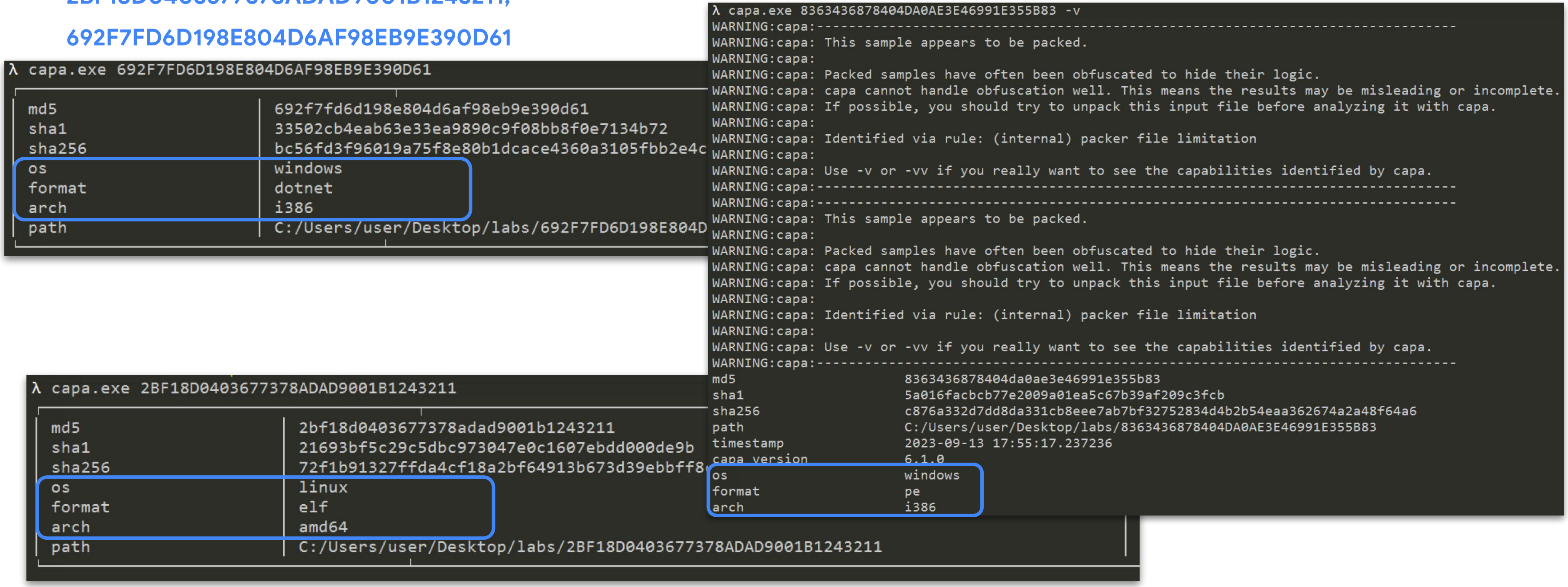
lab one answers (a)

Which of the files is a Windows PE? Linux ELF? Windows .NET PE?

8363436878404DA0AE3E46991E355B83,

2BF18D0403677378ADAD9001B1243211,

692F7FD6D198E804D6AF98EB9E390D61



```
λ capa.exe 692F7FD6D198E804D6AF98EB9E390D61

 md5                       692f7fd6d198e804d6af98eb9e390d61
 sha1                      33502cb4eab63e33ea9890c9f08bb8f0e7134b72
 sha256                    bc56fd3f96019a75f8e80b1dcace4360a3105fbb2e4c
 os                        windows
 format                    dotnet
 arch                      i386
 path                      C:/Users/user/Desktop/labs/692F7FD6D198E804D
```

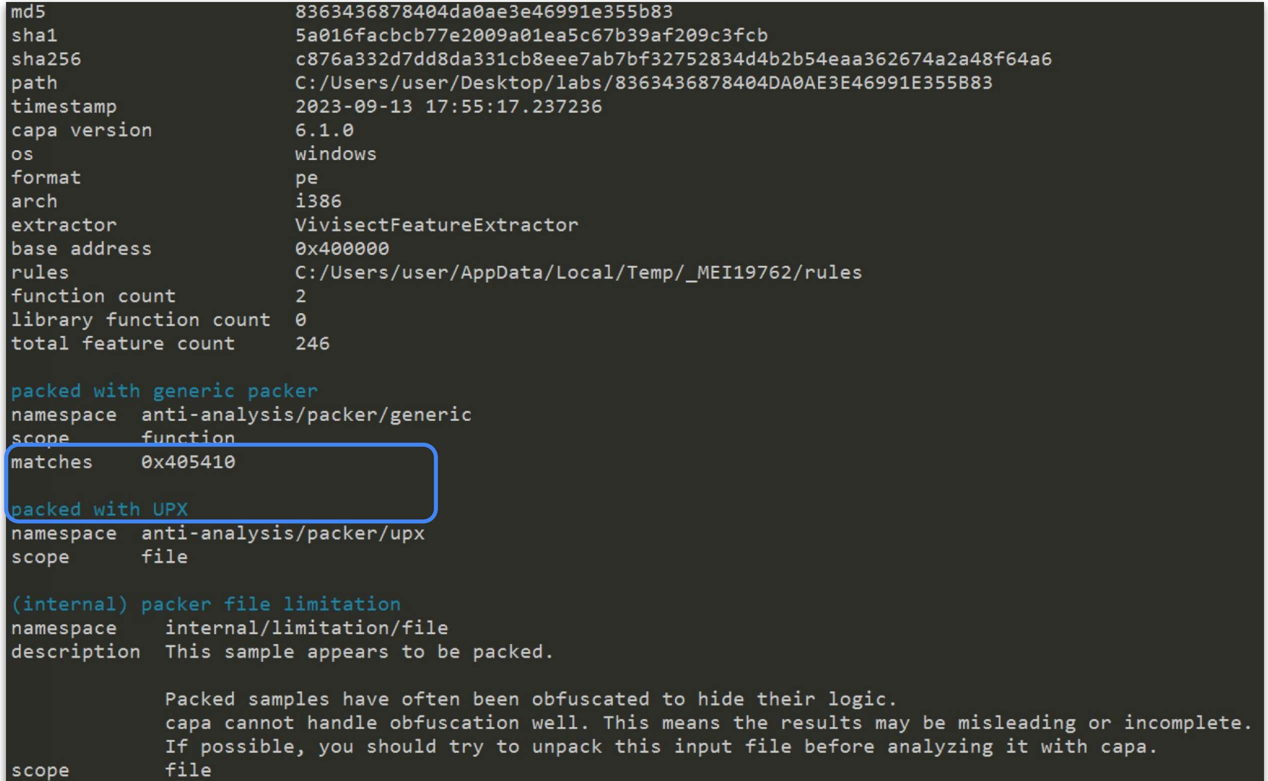
```
λ capa.exe 2BF18D0403677378ADAD9001B1243211

 md5                       2bf18d0403677378adad9001b1243211
 sha1                      21693bf5c29c5dbc973047e0c1607ebdd000de9b
 sha256                    72f1b91327ffda4cf18a2bf64913b673d39ebbff8
 os                        linux
 format                    elf
 arch                      amd64
 path                      C:/Users/user/Desktop/labs/2BF18D0403677378ADAD9001B1243211
```

```
λ capa.exe 8363436878404DA0AE3E46991E355B83 -v
WARNING:capa:--------------------------------------------------------------------------------
WARNING:capa: This sample appears to be packed.
WARNING:capa:
WARNING:capa: Packed samples have often been obfuscated to hide their logic.
WARNING:capa: capa cannot handle obfuscation well. This means the results may be misleading or incomplete.
WARNING:capa: If possible, you should try to unpack this input file before analyzing it with capa.
WARNING:capa:
WARNING:capa: Identified via rule: (internal) packer file limitation
WARNING:capa:
WARNING:capa: Use -v or -vv if you really want to see the capabilities identified by capa.
WARNING:capa:--------------------------------------------------------------------------------
WARNING:capa:--------------------------------------------------------------------------------
WARNING:capa: This sample appears to be packed.
WARNING:capa:
WARNING:capa: Packed samples have often been obfuscated to hide their logic.
WARNING:capa: capa cannot handle obfuscation well. This means the results may be misleading or incomplete.
WARNING:capa: If possible, you should try to unpack this input file before analyzing it with capa.
WARNING:capa:
WARNING:capa: Identified via rule: (internal) packer file limitation
WARNING:capa:
WARNING:capa: Use -v or -vv if you really want to see the capabilities identified by capa.
WARNING:capa:--------------------------------------------------------------------------------
md5                     8363436878404da0ae3e46991e355b83
sha1                    5a016facbcb77e2009a01ea5c67b39af209c3fcb
sha256                  c876a332d7dd8da331cb8eee7ab7bf32752834d4b2b54eaa362674a2a48f64a6
path                    C:/Users/user/Desktop/labs/8363436878404DA0AE3E46991E355B83
timestamp               2023-09-13 17:55:17.237236
capa version            6.1.0
os                      windows
format                  pe
arch                    i386
```

lab one answers (a)

Which of the files is packed? 8363436878404DA0AE3E46991E355B83

Using what packer? UPX

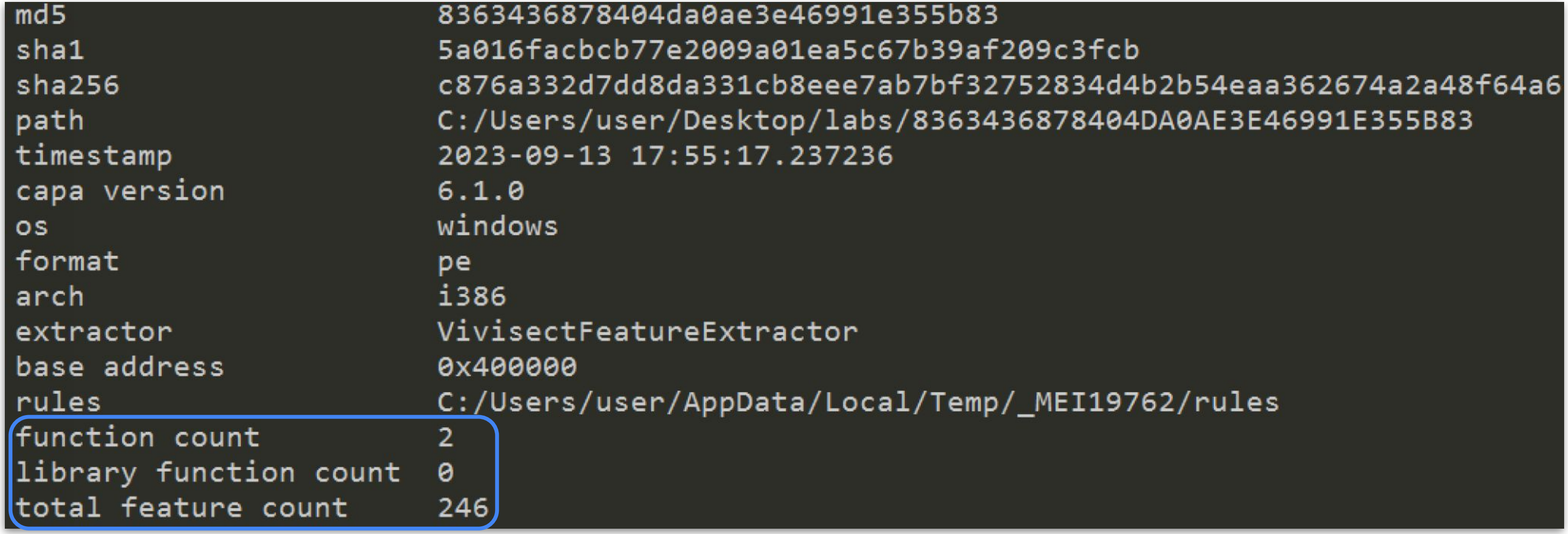


```
md5                     8363436878404da0ae3e46991e355b83
sha1                    5a016facbcb77e2009a01ea5c67b39af209c3fcb
sha256                  c876a332d7dd8da331cb8eee7ab7bf32752834d4b2b54eaa362674a2a48f64a6
path                    C:/Users/user/Desktop/labs/8363436878404DA0AE3E46991E355B83
timestamp               2023-09-13 17:55:17.237236
capa version            6.1.0
os                      windows
format                  pe
arch                    i386
extractor               VivisectFeatureExtractor
base address            0x400000
rules                   C:/Users/user/AppData/Local/Temp/_MEI19762/rules
function count          2
library function count  0
total feature count     246

packed with generic packer
namespace  anti-analysis/packer/generic
scope      function
matches    0x405410

packed with UPX
namespace  anti-analysis/packer/upx
scope      file

(internal) packer file limitation
namespace    internal/limitation/file
description  This sample appears to be packed.

             Packed samples have often been obfuscated to hide their logic.
             capa cannot handle obfuscation well. This means the results may be misleading or incomplete.
             If possible, you should try to unpack this input file before analyzing it with capa.
scope        file
```

lab one answers (b)

How many functions does capa identify in the packed file? **2**

How many features? **246**

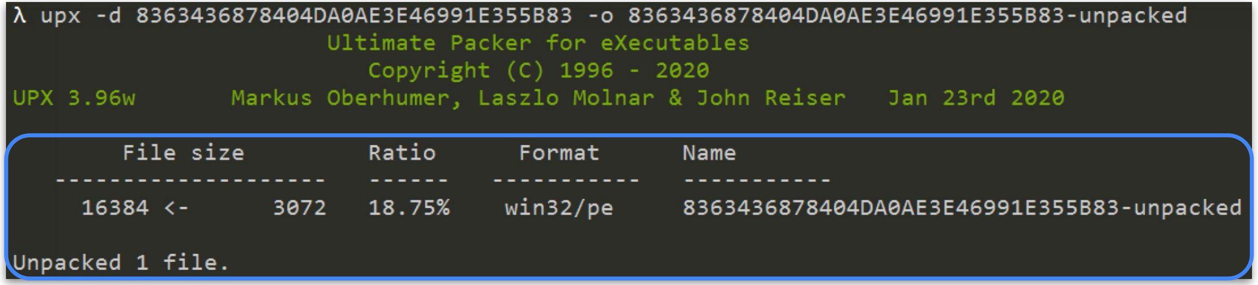


```
md5                     8363436878404da0ae3e46991e355b83
sha1                    5a016facbcb77e2009a01ea5c67b39af209c3fcb
sha256                  c876a332d7dd8da331cb8eee7ab7bf32752834d4b2b54eaa362674a2a48f64a6
path                    C:/Users/user/Desktop/labs/8363436878404DA0AE3E46991E355B83
timestamp               2023-09-13 17:55:17.237236
capa version            6.1.0
os                      windows
format                  pe
arch                    i386
extractor               VivisectFeatureExtractor
base address            0x400000
rules                   C:/Users/user/AppData/Local/Temp/_MEI19762/rules
function count          2
library function count  0
total feature count     246
```

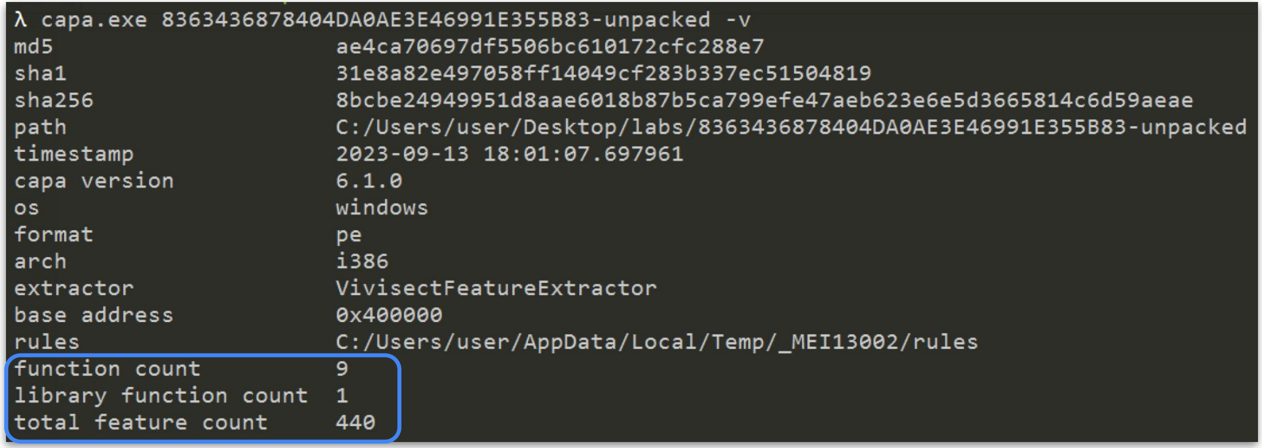
lab one answers (b)

How many functions does capa identify in the unpacked file? **9**

How many features? **440**



```
λ upx -d 8363436878404DA0AE3E46991E355B83 -o 8363436878404DA0AE3E46991E355B83-unpacked
                       Ultimate Packer for eXecutables
                          Copyright (C) 1996 - 2020
UPX 3.96w       Markus Oberhumer, Laszlo Molnar & John Reiser   Jan 23rd 2020

        File size         Ratio      Format      Name
   --------------------   ------   -----------   -----------
     16384 <-      3072   18.75%    win32/pe     8363436878404DA0AE3E46991E355B83-unpacked

Unpacked 1 file.
```

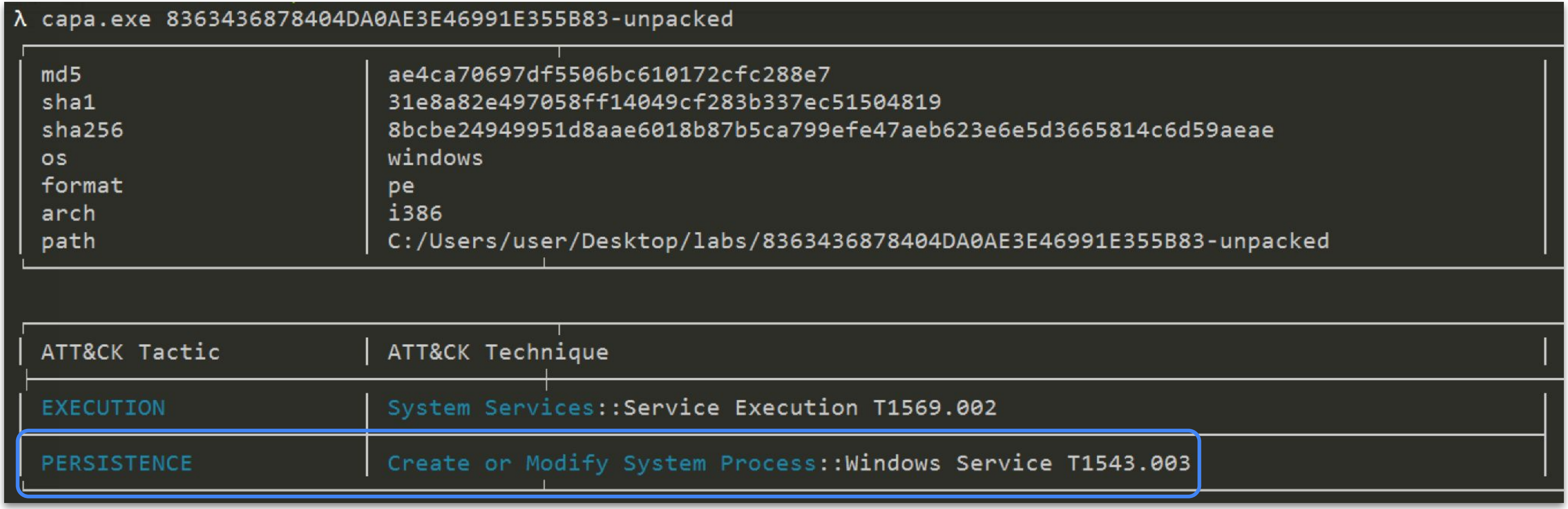


```
λ capa.exe 8363436878404DA0AE3E46991E355B83-unpacked -v
md5                     ae4ca70697df5506bc610172cfc288e7
sha1                    31e8a82e497058ff14049cf283b337ec51504819
sha256                  8bcbe24949951d8aae6018b87b5ca799efe47aeb623e6e5d3665814c6d59aeae
path                    C:/Users/user/Desktop/labs/8363436878404DA0AE3E46991E355B83-unpacked
timestamp               2023-09-13 18:01:07.697961
capa version            6.1.0
os                      windows
format                  pe
arch                    i386
extractor               VivisectFeatureExtractor
base address            0x400000
rules                   C:/Users/user/AppData/Local/Temp/_MEI13002/rules
function count          9
library function count  1
total feature count     440
```

lab one answers (c)

Which file(s) use MITRE ATT&CK persistence tactics? 8363436878404da0ae3e46991e355b83-unpacked

What is the specific persistence technique(s)? persist via Windows service

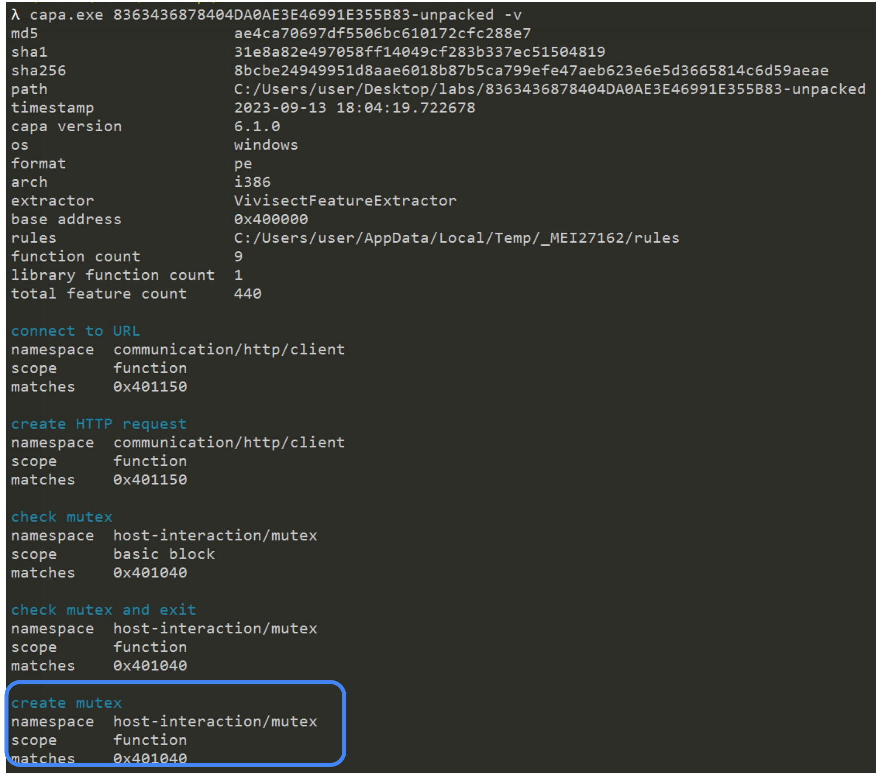


```
λ capa.exe 8363436878404DA0AE3E46991E355B83-unpacked

  md5                      ae4ca70697df5506bc610172cfc288e7
  sha1                     31e8a82e497058ff14049cf283b337ec51504819
  sha256                   8bcbe24949951d8aae6018b87b5ca799efe47aeb623e6e5d3665814c6d59aeae
  os                       windows
  format                   pe
  arch                     i386
  path                     C:/Users/user/Desktop/labs/8363436878404DA0AE3E46991E355B83-unpacked


  ATT&CK Tactic            ATT&CK Technique

  EXECUTION                System Services::Service Execution T1569.002

  PERSISTENCE              Create or Modify System Process::Windows Service T1543.003
```

lab one answers (c)

Which file(s) create a mutex? 8363436878404da0ae3e46991e355b83-unpacked

Which function address is responsible for creating the mutex? 0x401040



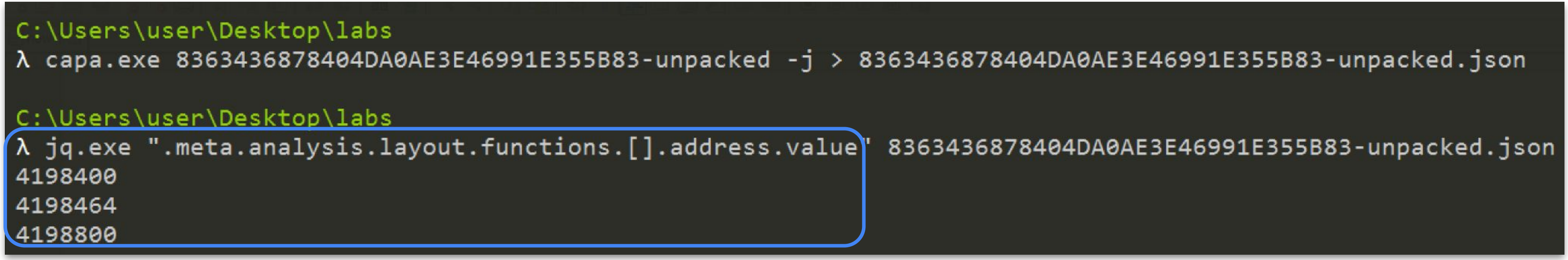
```
λ capa.exe 8363436878404DA0AE3E46991E355B83-unpacked -v
md5                     ae4ca70697df5506bc610172cfc288e7
sha1                    31e8a82e497058ff14049cf283b337ec51504819
sha256                  8bcbe24949951d8aae6018b87b5ca799efe47aeb623e6e5d3665814c6d59aeae
path                    C:/Users/user/Desktop/labs/8363436878404DA0AE3E46991E355B83-unpacked
timestamp               2023-09-13 18:04:19.722678
capa version            6.1.0
os                      windows
format                  pe
arch                    i386
extractor               VivisectFeatureExtractor
base address            0x400000
rules                   C:/Users/user/AppData/Local/Temp/_MEI27162/rules
function count          9
library function count  1
total feature count     440

connect to URL
namespace   communication/http/client
scope       function
matches     0x401150

create HTTP request
namespace   communication/http/client
scope       function
matches     0x401150

check mutex
namespace   host-interaction/mutex
scope       basic block
matches     0x401040

check mutex and exit
namespace   host-interaction/mutex
scope       function
matches     0x401040

create mutex
namespace   host-interaction/mutex
scope       function
matches     0x401040
```

MANDIANT
NOW PART OF Google Cloud

lab one answers (bonus)

Execute capa to generate JSON-formatted output to a file for the unpacked Windows PE file and use jq to display the address of each matched function

```
> jq ".meta.analysis.layout.functions.[].address.value" /path/to/json
4198400
4198464
4198800
```



```
C:\Users\user\Desktop\labs
λ capa.exe 8363436878404DA0AE3E46991E355B83-unpacked -j > 8363436878404DA0AE3E46991E355B83-unpacked.json

C:\Users\user\Desktop\labs
λ jq.exe ".meta.analysis.layout.functions.[].address.value" 8363436878404DA0AE3E46991E355B83-unpacked.json
4198400
4198464
4198800
```

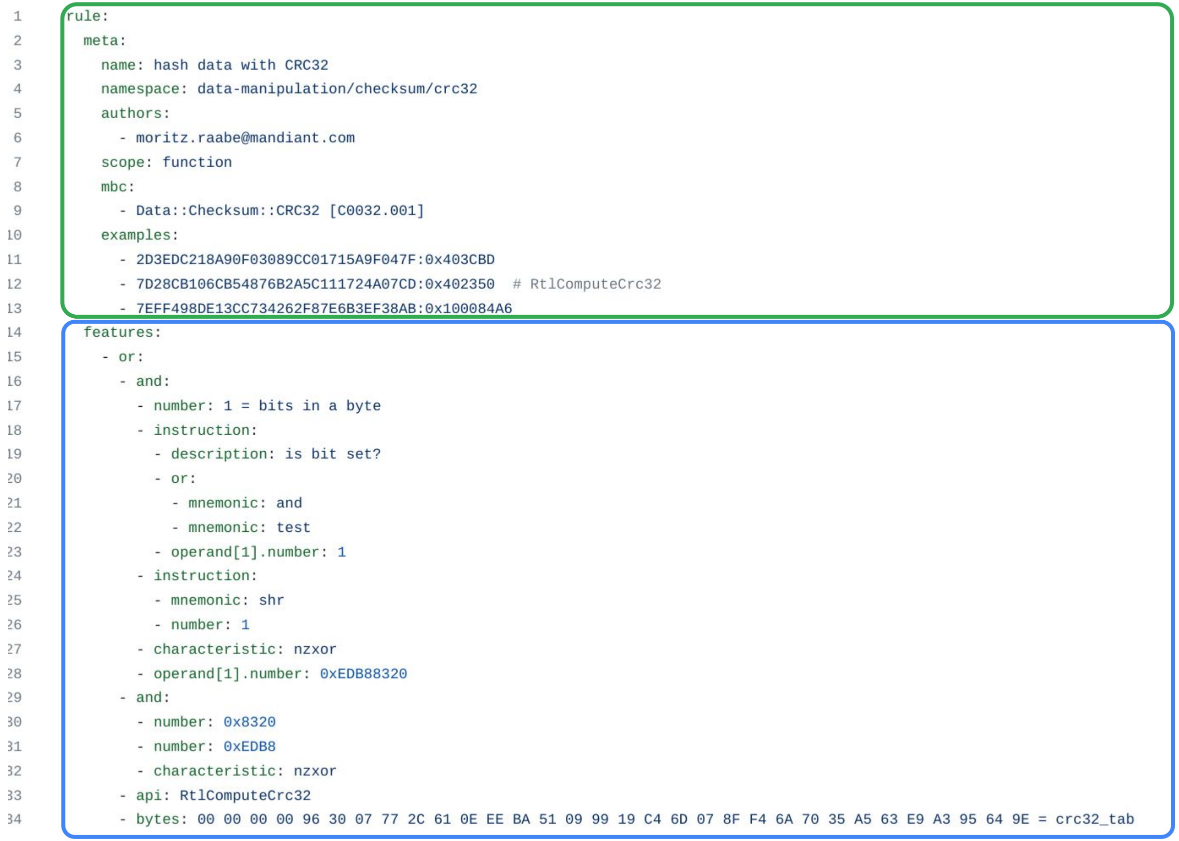
04

capa rules

rule format

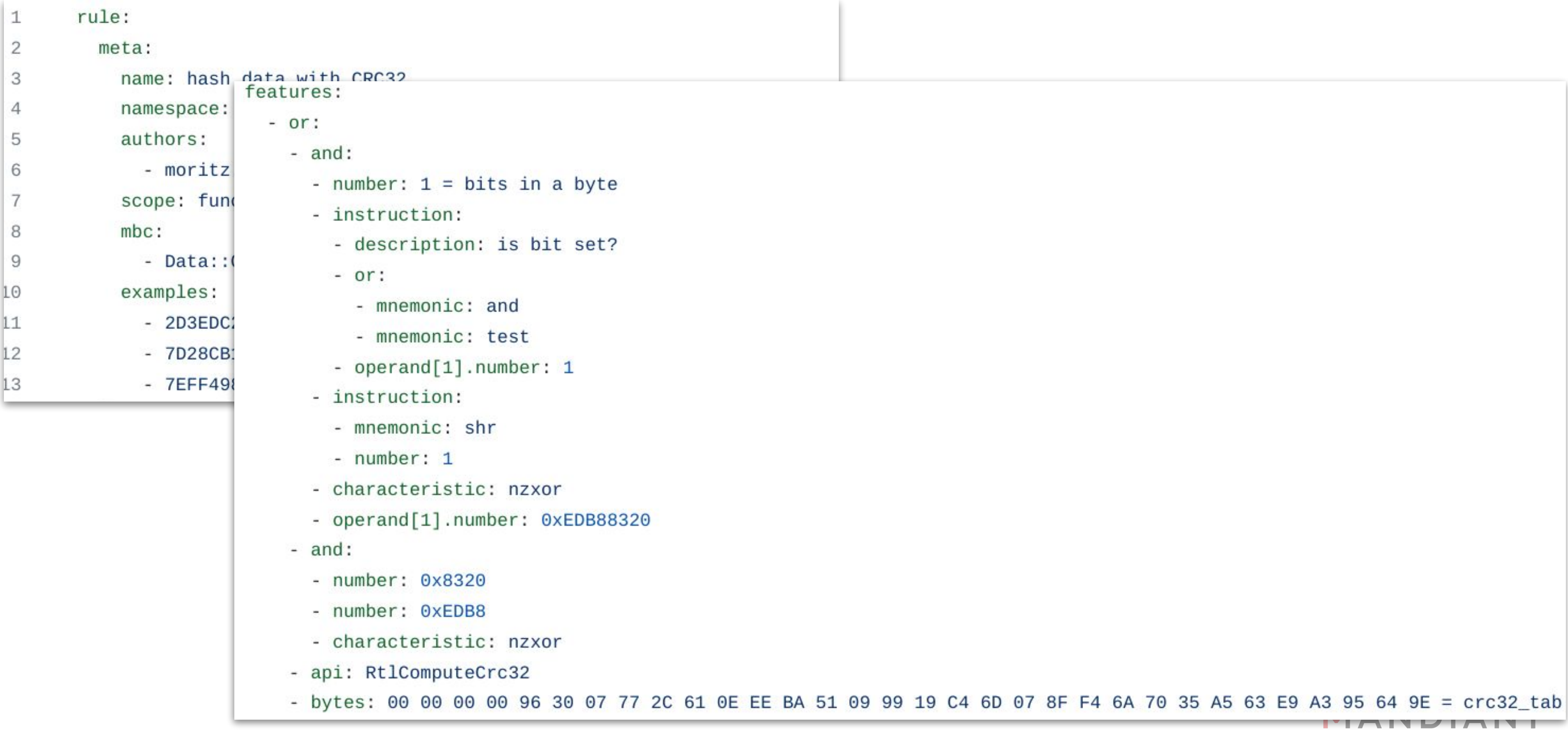
YAML-based format that contains two main blocks

- **meta**
- **features**



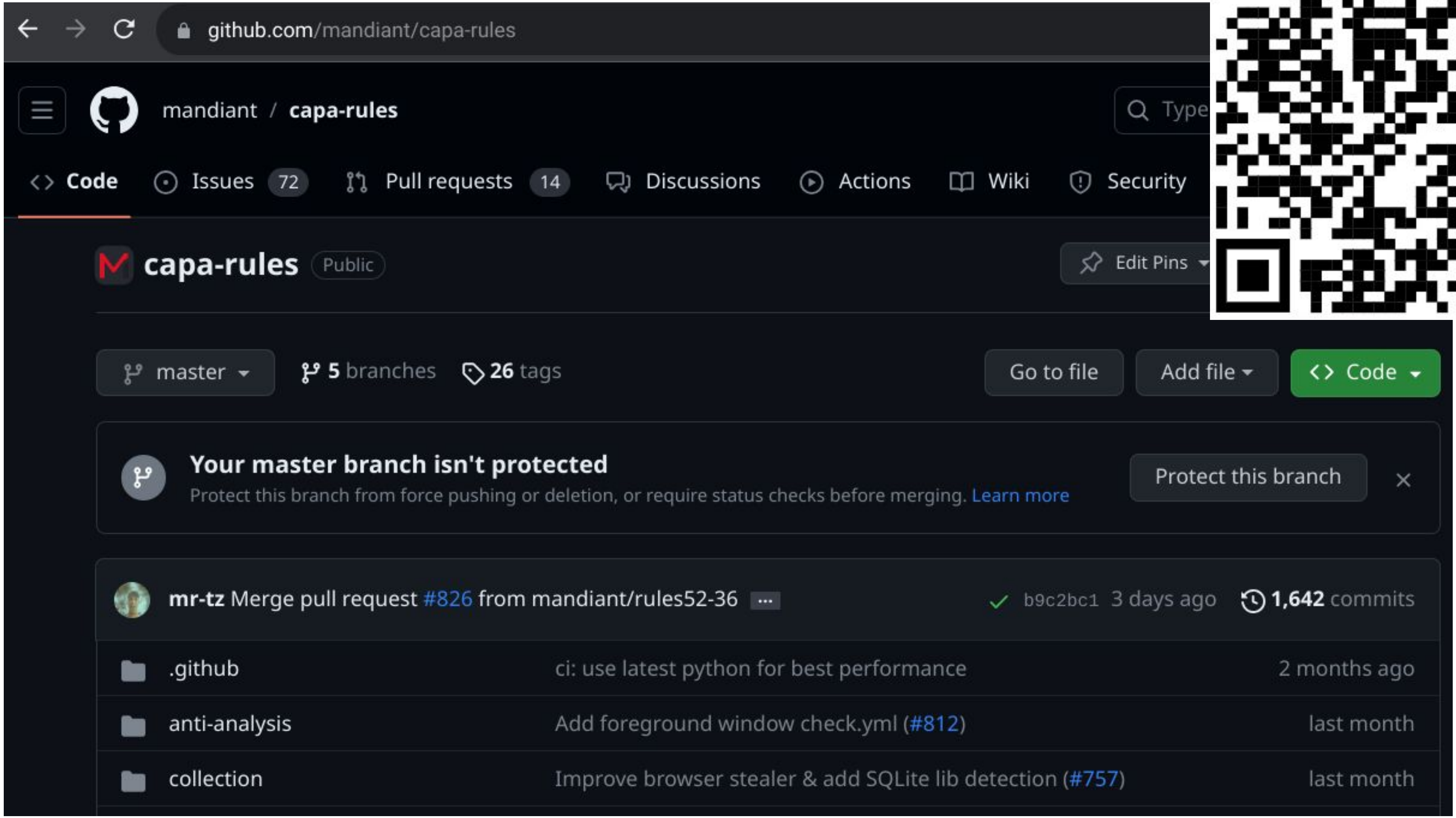
```
rule:
  meta:
    name: hash data with CRC32
    namespace: data-manipulation/checksum/crc32
    authors:
      - moritz.raabe@mandiant.com
    scope: function
    mbc:
      - Data::Checksum::CRC32 [C0032.001]
    examples:
      - 2D3EDC218A90F03089CC01715A9F047F:0x403CBD
      - 7D28CB106CB54876B2A5C111724A07CD:0x402350  # RtlComputeCrc32
      - 7EFF498DE13CC734262F87E6B3EF38AB:0x100084A6
  features:
    - or:
      - and:
        - number: 1 = bits in a byte
        - instruction:
          - description: is bit set?
          - or:
            - mnemonic: and
            - mnemonic: test
          - operand[1].number: 1
        - instruction:
          - mnemonic: shr
          - number: 1
        - characteristic: nzxor
        - operand[1].number: 0xEDB88320
      - and:
        - number: 0x8320
        - number: 0xEDB8
        - characteristic: nzxor
      - api: RtlComputeCrc32
      - bytes: 00 00 00 00 96 30 07 77 2C 61 0E EE BA 51 09 99 19 C4 6D 07 8F F4 6A 70 35 A5 63 E9 A3 95 64 9E = crc32_tab
```

rule format



```
rule:
  meta:
    name: hash data with CRC32
    namespace:
    authors:
      - moritz
    scope: func
    mbc:
      - Data::
    examples:
      - 2D3EDC
      - 7D28CB
      - 7EFF49
```

```
features:
  - or:
    - and:
      - number: 1 = bits in a byte
      - instruction:
        - description: is bit set?
        - or:
          - mnemonic: and
          - mnemonic: test
        - operand[1].number: 1
      - instruction:
        - mnemonic: shr
        - number: 1
      - characteristic: nzxor
      - operand[1].number: 0xEDB88320
    - and:
      - number: 0x8320
      - number: 0xEDB8
      - characteristic: nzxor
    - api: RtlComputeCrc32
    - bytes: 00 00 00 00 96 30 07 77 2C 61 0E EE BA 51 09 99 19 C4 6D 07 8F F4 6A 70 35 A5 63 E9 A3 95 64 9E = crc32_tab
```



Your master branch isn't protected

Protect this branch from force pushing or deletion, or require status checks before merging. Learn more

mr-tz Merge pull request #826 from mandiant/rules52-36 … ✓ b9c2bc1 3 days ago 1,642 commits

.github	ci: use latest python for best performance	2 months ago
anti-analysis	Add foreground window check.yml (#812)	last month
collection	Improve browser stealer & add SQLite lib detection (#757)	last month

capa statistics

830 rules

Written and vetted by experts

75 contributors

Security practitioners

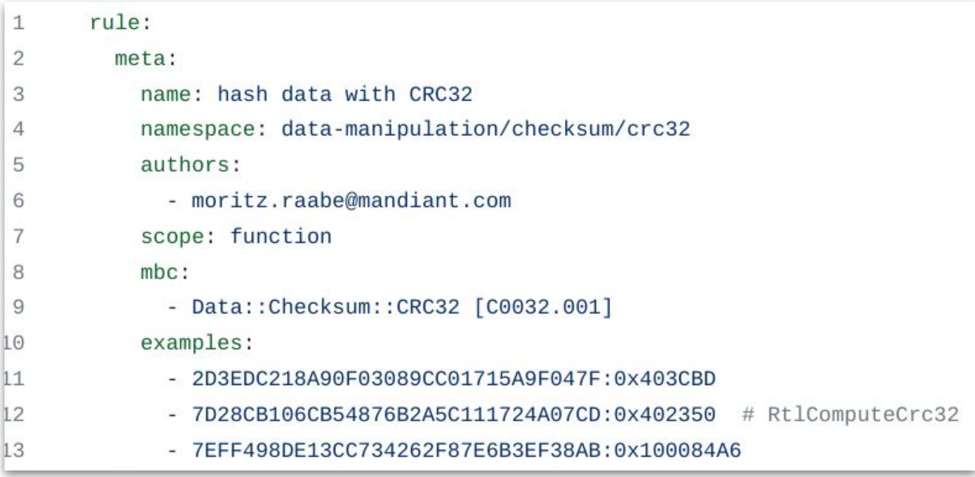
180,000 downloads

Since first release in 2020

meta block

Identifies the rule, groups the technique, and provides references to documentation

Mix of required and optional fields



```
rule:
  meta:
    name: hash data with CRC32
    namespace: data-manipulation/checksum/crc32
    authors:
      - moritz.raabe@mandiant.com
    scope: function
    mbc:
      - Data::Checksum::CRC32 [C0032.001]
    examples:
      - 2D3EDC218A90F03089CC01715A9F047F:0x403CBD
      - 7D28CB106CB54876B2A5C111724A07CD:0x402350  # RtlComputeCrc32
      - 7EFF498DE13CC734262F87E6B3EF38AB:0x100084A6
```

required fields

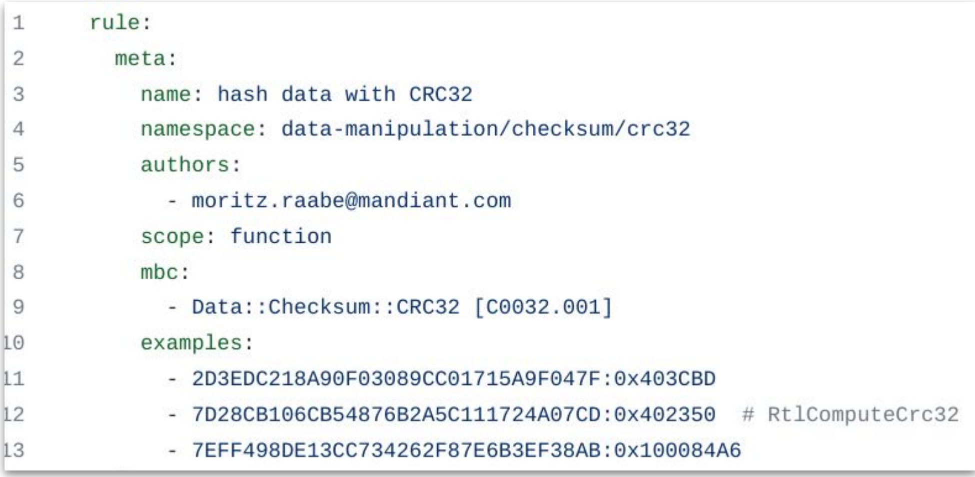
name: Uniquely identifies rule

namespace: Groups related rules

authors: Lists rule author(s) name or handle

scope: Specifies feature set applied to rule

- **instruction** (most specific)
- **basic block**
- **function**
- **file** (most general)



```
rule:
  meta:
    name: hash data with CRC32
    namespace: data-manipulation/checksum/crc32
    authors:
      - moritz.raabe@mandiant.com
    scope: function
    mbc:
      - Data::Checksum::CRC32 [C0032.001]
    examples:
      - 2D3EDC218A90F03089CC01715A9F047F:0x403CBD
      - 7D28CB106CB54876B2A5C111724A07CD:0x402350  # RtlComputeCrc32
      - 7EFF498DE13CC734262F87E6B3EF38AB:0x100084A6
```

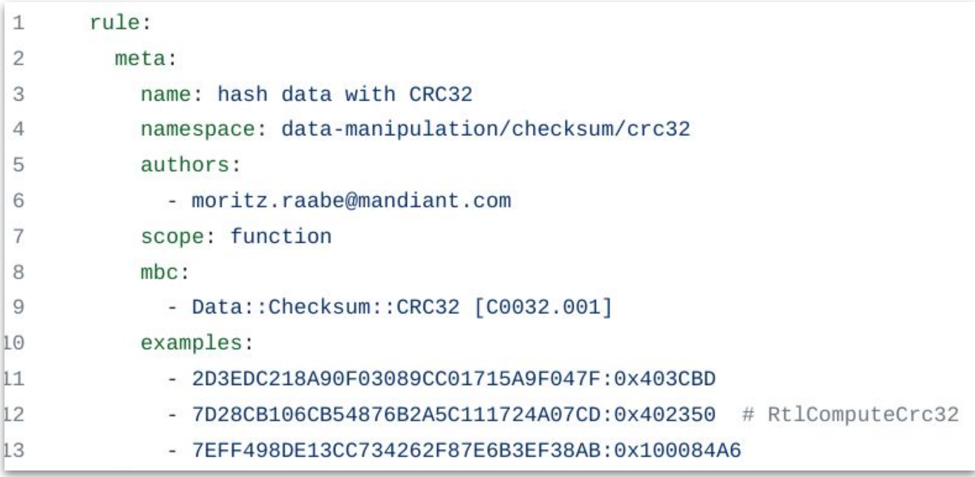
optional fields

description: Provides additional context on rule’s intent

att&ck: Specifies ATT&CK framework technique

mbc: Specifies Malware Behavior Catalog technique

examples: Lists reference samples that match rule



```
rule:
  meta:
    name: hash data with CRC32
    namespace: data-manipulation/checksum/crc32
    authors:
      - moritz.raabe@mandiant.com
    scope: function
    mbc:
      - Data::Checksum::CRC32 [C0032.001]
    examples:
      - 2D3EDC218A90F03089CC01715A9F047F:0x403CBD
      - 7D28CB106CB54876B2A5C111724A07CD:0x402350  # RtlComputeCrc32
      - 7EFF498DE13CC734262F87E6B3EF38AB:0x100084A6
```

features block

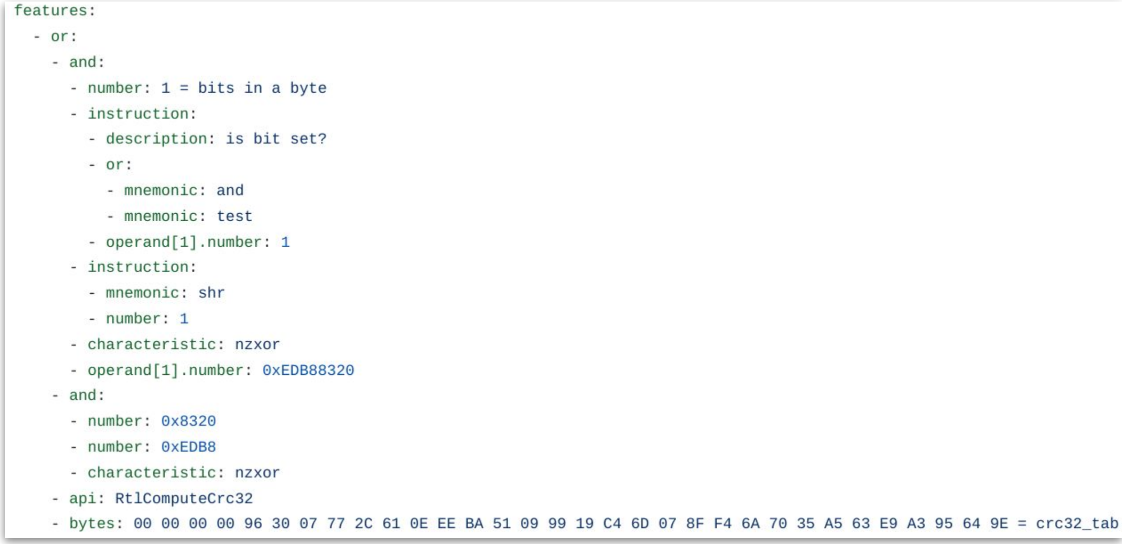
Logic tree consisting of nested combinations of structural expressions, features, and characteristics

Structural expressions

- **and**: All children must match
- **or:** Match at least one child
- **not:** Match when child expression does not
- ***n* or more:** Match at least *n* or more children
  - **optional** (0 or more)

Scopes

- **instruction** (most specific)
- **basic block**
- **function** (most general)



```
features:
  - or:
    - and:
      - number: 1 = bits in a byte
      - instruction:
        - description: is bit set?
        - or:
          - mnemonic: and
          - mnemonic: test
        - operand[1].number: 1
      - instruction:
        - mnemonic: shr
        - number: 1
      - characteristic: nzxor
      - operand[1].number: 0xEDB88320
    - and:
      - number: 0x8320
      - number: 0xEDB8
      - characteristic: nzxor
    - api: RtlComputeCrc32
    - bytes: 00 00 00 00 96 30 07 77 2C 61 0E EE BA 51 09 99 19 C4 6D 07 8F F4 6A 70 35 A5 63 E9 A3 95 64 9E = crc32_tab
```

MANDIANT

NOW PART OF Google Cloud

features and characteristics

Features are extracted from multiple scopes, starting with most specific (**instruction**), and working towards most general (**file**)

Characteristics are one-off features that represent unique or interesting functionality

file

(sub)string
export
import
section
forwarded export

function-name
namespace
class
embedded pe
mixed mode

function

loop
recursive call

calls from
calls to

basic block

tight loop

stack string

instruction

namespace
class
api
property
number
nzxor
peb access
fs access
gs access

(sub)string
bytes
offset
mnemonic
operand
cross section flow
indirect call
call $+5
unmanaged call

(global)

os
arch

format

features and characteristics

Features are extracted from multiple scopes, starting with most specific (**instruction**), and working towards most general (**file**)

Characteristics are one-off features that represent unique or interesting functionality

file

(sub)string
export
import
section
forwarded export

function-name
namespace
class
embedded pe
mixed mode

function

loop
recursive call

calls from
calls to

basic block

tight loop

stack string

instruction

namespace
class
api
property
number
nzxor
peb access
fs access
gs access

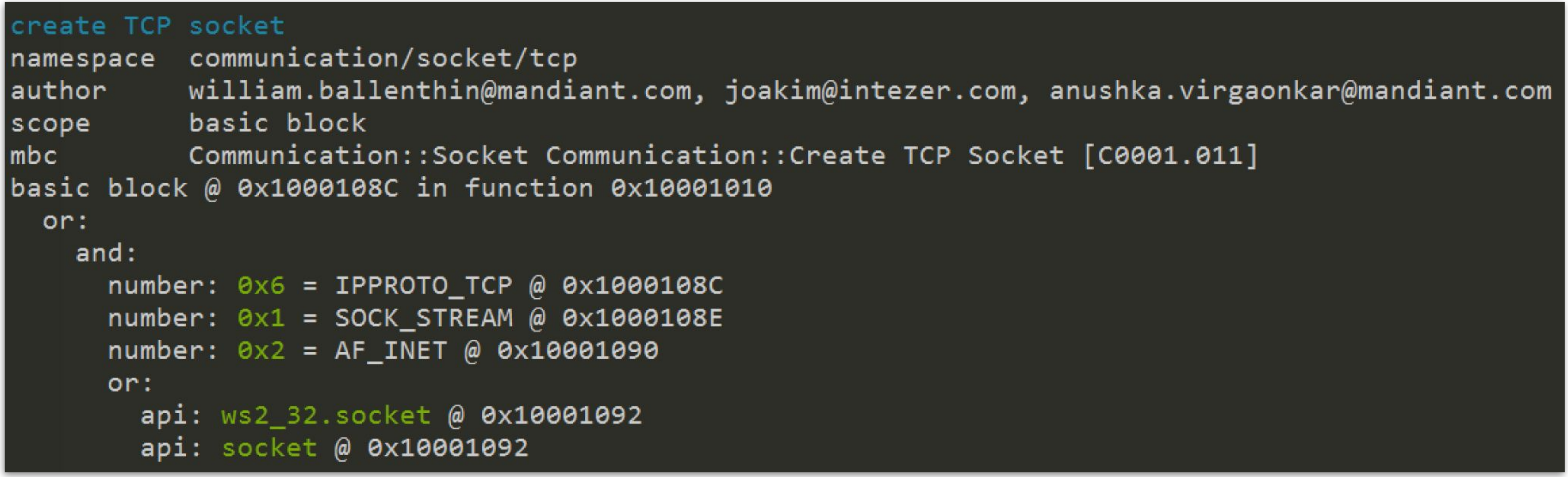
(sub)string
bytes
offset
mnemonic
operand
cross section flow
indirect call
call $+5
unmanaged call

(global)

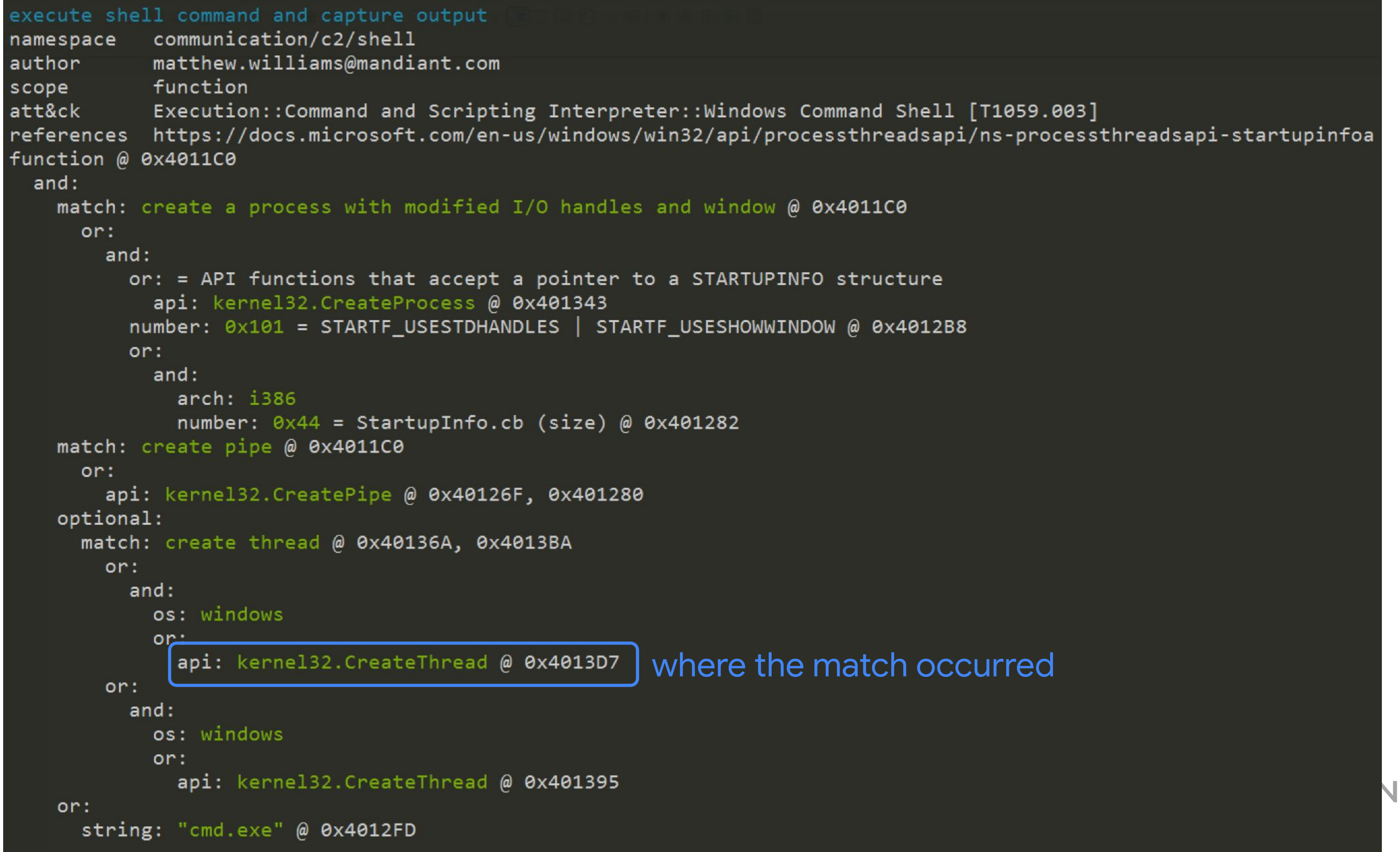
os
arch

format

capa.exe /path/to/file -vv

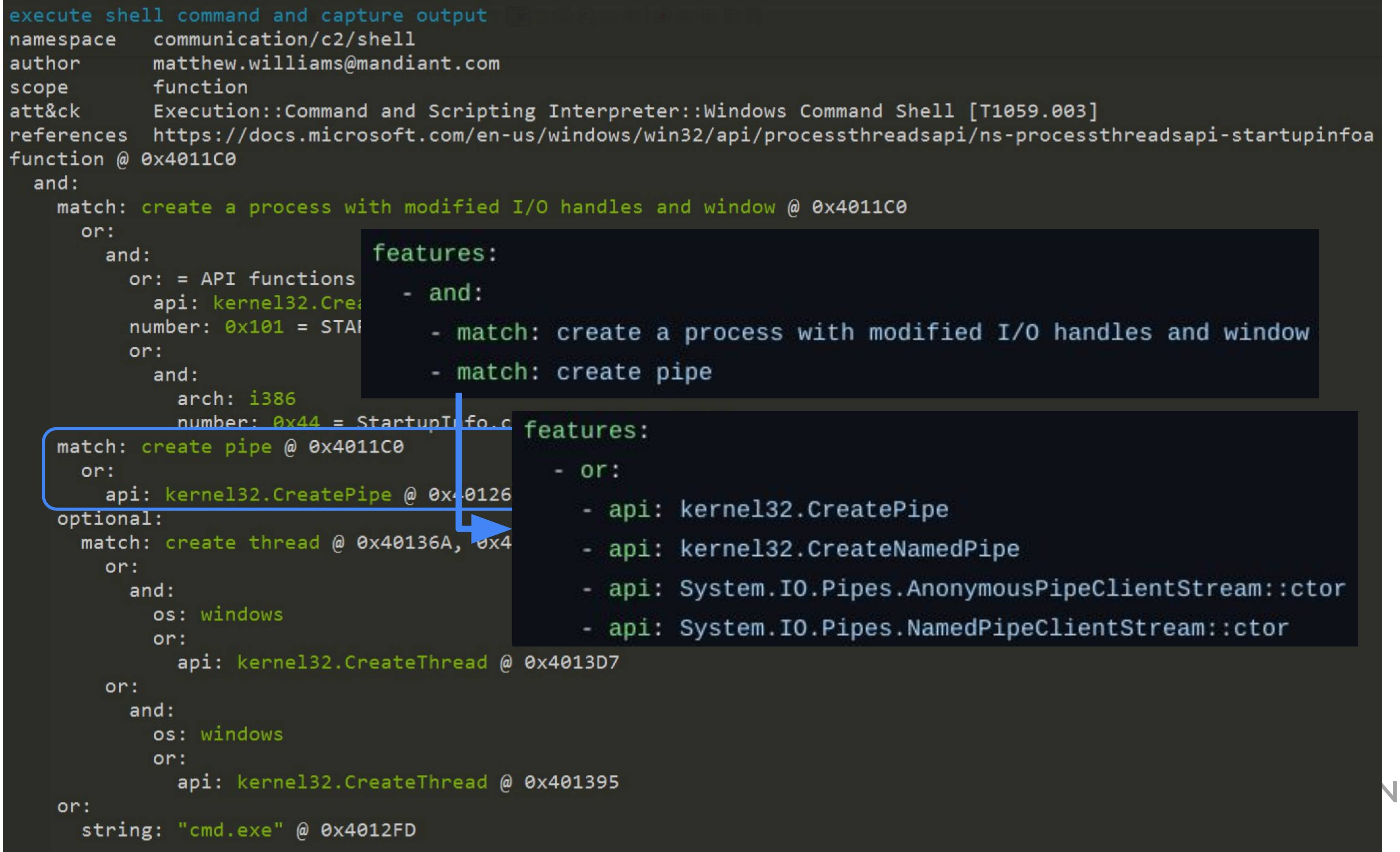


```
create TCP socket
namespace  communication/socket/tcp
author     william.ballenthin@mandiant.com, joakim@intezer.com, anushka.virgaonkar@mandiant.com
scope      basic block
mbc        Communication::Socket Communication::Create TCP Socket [C0001.011]
basic block @ 0x1000108C in function 0x10001010
  or:
    and:
      number: 0x6 = IPPROTO_TCP @ 0x1000108C
      number: 0x1 = SOCK_STREAM @ 0x1000108E
      number: 0x2 = AF_INET @ 0x10001090
      or:
        api: ws2_32.socket @ 0x10001092
        api: socket @ 0x10001092
```



```
execute shell command and capture output
namespace   communication/c2/shell
author      matthew.williams@mandiant.com
scope       function
att&ck      Execution::Command and Scripting Interpreter::Windows Command Shell [T1059.003]
references  https://docs.microsoft.com/en-us/windows/win32/api/processthreadsapi/ns-processthreadsapi-startupinfoa
function @ 0x4011C0
  and:
    match: create a process with modified I/O handles and window @ 0x4011C0
      or:
        and:
          or: = API functions that accept a pointer to a STARTUPINFO structure
            api: kernel32.CreateProcess @ 0x401343
          number: 0x101 = STARTF_USESTDHANDLES | STARTF_USESHOWWINDOW @ 0x4012B8
          or:
            and:
              arch: i386
              number: 0x44 = StartupInfo.cb (size) @ 0x401282
    match: create pipe @ 0x4011C0
      or:
        api: kernel32.CreatePipe @ 0x40126F, 0x401280
    optional:
      match: create thread @ 0x40136A, 0x4013BA
        or:
          and:
            os: windows
            or:
              api: kernel32.CreateThread @ 0x4013D7
        or:
          and:
            os: windows
            or:
              api: kernel32.CreateThread @ 0x401395
    or:
      string: "cmd.exe" @ 0x4012FD
```

where the match occurred



```
execute shell command and capture output
namespace   communication/c2/shell
author      matthew.williams@mandiant.com
scope       function
att&ck      Execution::Command and Scripting Interpreter::Windows Command Shell [T1059.003]
references  https://docs.microsoft.com/en-us/windows/win32/api/processthreadsapi/ns-processthreadsapi-startupinfoa
function @ 0x4011C0
  and:
    match: create a process with modified I/O handles and window @ 0x4011C0
      or:
        and:
          or: = API functions
            api: kernel32.Crea
          number: 0x101 = STAR
          or:
            and:
              arch: i386
              number: 0x44 = StartupInfo.c
    match: create pipe @ 0x4011C0
      or:
        api: kernel32.CreatePipe @ 0x40126
    optional:
      match: create thread @ 0x40136A, 0x4
        or:
          and:
            os: windows
            or:
              api: kernel32.CreateThread @ 0x4013D7
        or:
          and:
            os: windows
            or:
              api: kernel32.CreateThread @ 0x401395
    or:
      string: "cmd.exe" @ 0x4012FD
```

```
features:
  - and:
    - match: create a process with modified I/O handles and window
    - match: create pipe
```

```
features:
  - or:
    - api: kernel32.CreatePipe
    - api: kernel32.CreateNamedPipe
    - api: System.IO.Pipes.AnonymousPipeClientStream::ctor
    - api: System.IO.Pipes.NamedPipeClientStream::ctor
```

lab two
reading capa rules

lab two reading capa rules

Use capa to answer the following questions using sample 9976ff9292264c5e58318e6b785fd13b:

a)

1. Based on which feature categories does capa recognize the **check for sandbox username or hostname** capability?
2. How many functions implement this capability?
3. List the sandbox usernames/hostname values that capa recognizes.

b)

1. How many features does capa use to detect the **reference anti-VM strings targeting VMWare** capability?
2. How many functions implement this capability?

c)

1. Which function **sends and receives data**?
2. Which APIs does the sample use to send and receive data?
3. How many submatches are identified in the function?

d)

1. How many **library rule matches** does capa identify in the sample?

lab two: reading capa rules

Sample: 9976ff9292264c5e58318e6b785fd13b

A)

1. Based on which feature categories does capa recognize the **check for sandbox username or hostname** capability?
2. How many functions implement this capability?
3. List the sandbox usernames/hostname values that capa recognizes.

B)

1. How many features does capa use to detect the **reference anti-VM strings targeting VirtualBox** capability?
2. How many functions implement this capability?

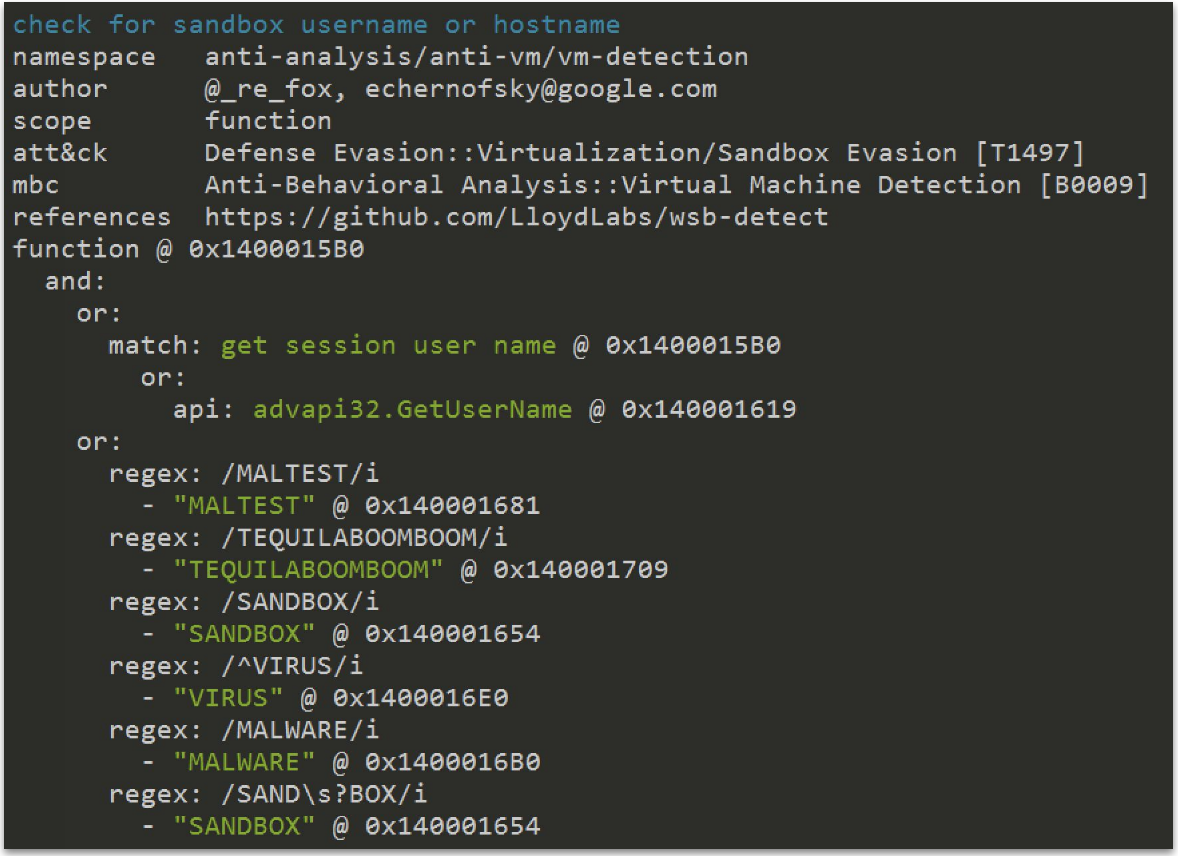
C)

1. Which function **sends and receives data**?
2. Which APIs does the sample use to send and receive data?
3. How many submatches are identified in the function?

D)

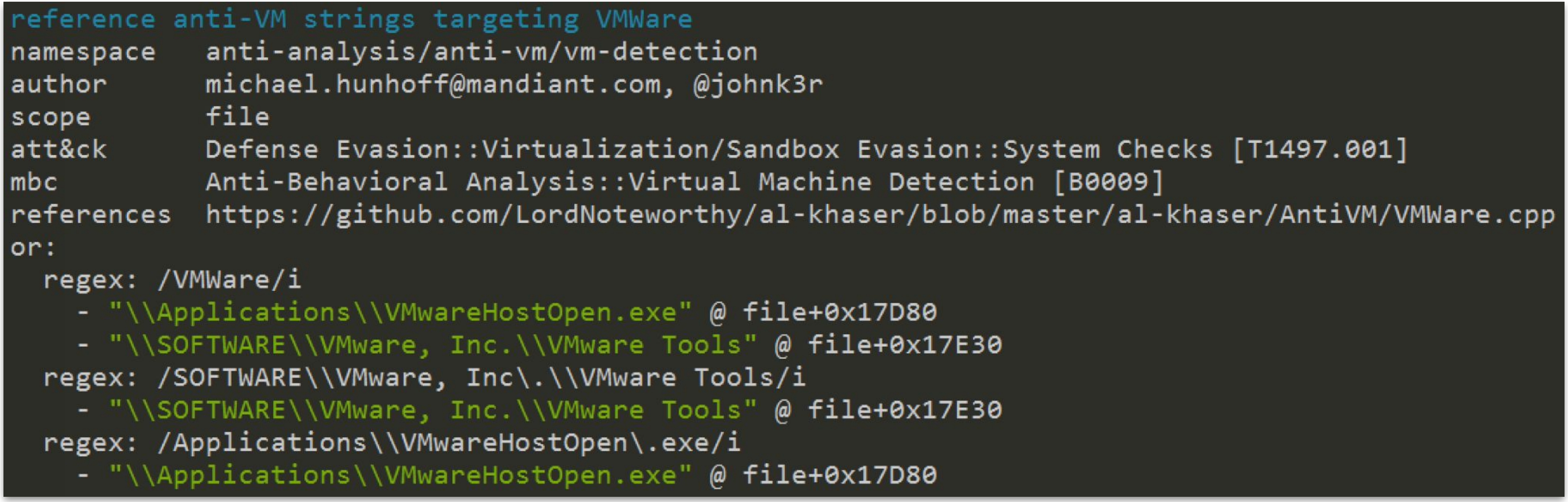
1. How many **library rule matches** does capa identify in the sample?

lab two answers (a)



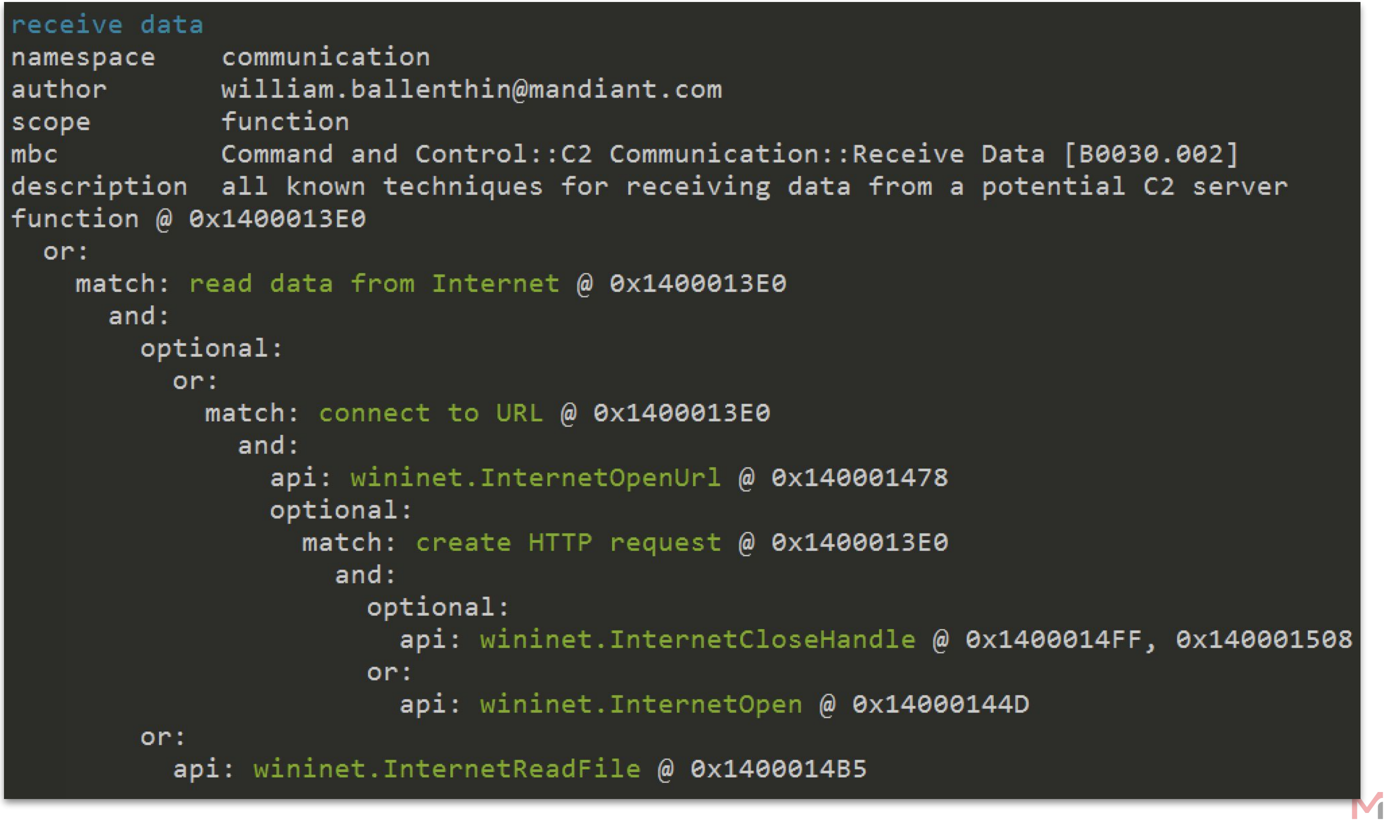
```
check for sandbox username or hostname
namespace   anti-analysis/anti-vm/vm-detection
author      @_re_fox, echernofsky@google.com
scope       function
att&ck      Defense Evasion::Virtualization/Sandbox Evasion [T1497]
mbc         Anti-Behavioral Analysis::Virtual Machine Detection [B0009]
references  https://github.com/LloydLabs/wsb-detect
function @ 0x1400015B0
  and:
    or:
      match: get session user name @ 0x1400015B0
        or:
          api: advapi32.GetUserName @ 0x140001619
    or:
      regex: /MALTEST/i
        - "MALTEST" @ 0x140001681
      regex: /TEQUILABOOMBOOM/i
        - "TEQUILABOOMBOOM" @ 0x140001709
      regex: /SANDBOX/i
        - "SANDBOX" @ 0x140001654
      regex: /^VIRUS/i
        - "VIRUS" @ 0x1400016E0
      regex: /MALWARE/i
        - "MALWARE" @ 0x1400016B0
      regex: /SAND\s?BOX/i
        - "SANDBOX" @ 0x140001654
```

lab two answers (b)



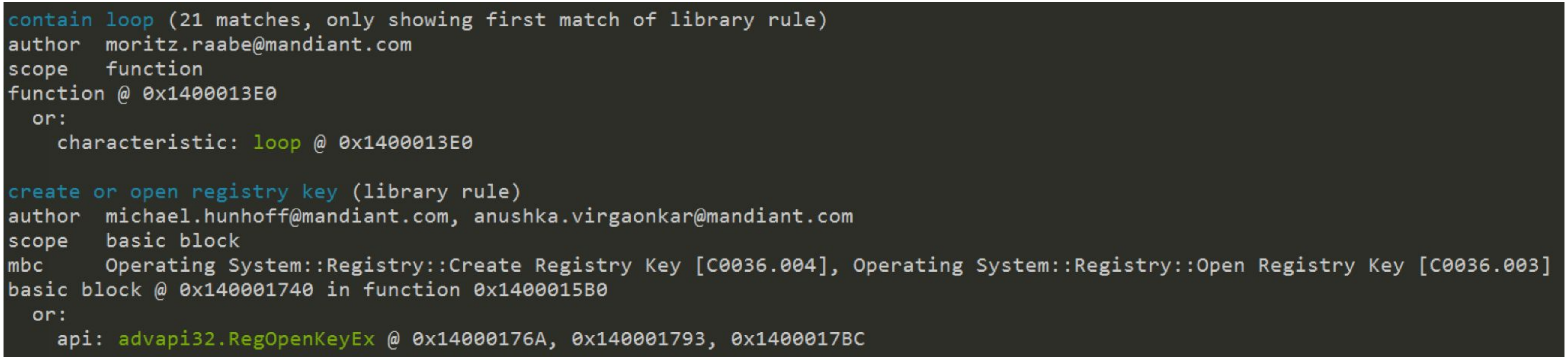
```
reference anti-VM strings targeting VMWare
namespace   anti-analysis/anti-vm/vm-detection
author      michael.hunhoff@mandiant.com, @johnk3r
scope       file
att&ck      Defense Evasion::Virtualization/Sandbox Evasion::System Checks [T1497.001]
mbc         Anti-Behavioral Analysis::Virtual Machine Detection [B0009]
references  https://github.com/LordNoteworthy/al-khaser/blob/master/al-khaser/AntiVM/VMWare.cpp
or:
  regex: /VMWare/i
    - "\\Applications\\VMwareHostOpen.exe" @ file+0x17D80
    - "\\SOFTWARE\\VMware, Inc.\\VMware Tools" @ file+0x17E30
  regex: /SOFTWARE\\VMware, Inc\.\\VMware Tools/i
    - "\\SOFTWARE\\VMware, Inc.\\VMware Tools" @ file+0x17E30
  regex: /Applications\\VMwareHostOpen\.exe/i
    - "\\Applications\\VMwareHostOpen.exe" @ file+0x17D80
```

lab two answers (c)



```
receive data
namespace    communication
author       william.ballenthin@mandiant.com
scope        function
mbc          Command and Control::C2 Communication::Receive Data [B0030.002]
description  all known techniques for receiving data from a potential C2 server
function @ 0x1400013E0
  or:
    match: read data from Internet @ 0x1400013E0
      and:
        optional:
          or:
            match: connect to URL @ 0x1400013E0
              and:
                api: wininet.InternetOpenUrl @ 0x140001478
                optional:
                  match: create HTTP request @ 0x1400013E0
                    and:
                      optional:
                        api: wininet.InternetCloseHandle @ 0x1400014FF, 0x140001508
                      or:
                        api: wininet.InternetOpen @ 0x14000144D
        or:
          api: wininet.InternetReadFile @ 0x1400014B5
```

lab two answers (d)



```
contain loop (21 matches, only showing first match of library rule)
author  moritz.raabe@mandiant.com
scope   function
function @ 0x1400013E0
  or:
    characteristic: loop @ 0x1400013E0

create or open registry key (library rule)
author  michael.hunhoff@mandiant.com, anushka.virgaonkar@mandiant.com
scope   basic block
mbc     Operating System::Registry::Create Registry Key [C0036.004], Operating System::Registry::Open Registry Key [C0036.003]
basic block @ 0x140001740 in function 0x1400015B0
  or:
    api: advapi32.RegOpenKeyEx @ 0x14000176A, 0x140001793, 0x1400017BC
```

05

writing rules

why are capa rules important?

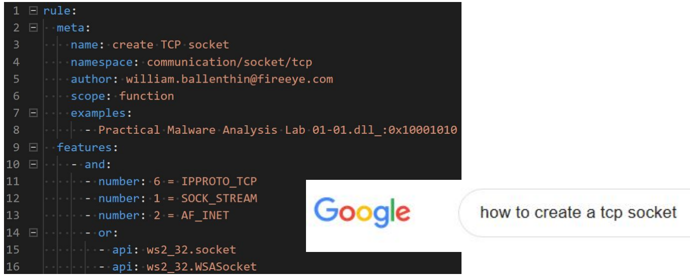
Foundation of capa's analysis

- Over 800 rules in official rule repository on GitHub

Extend capa to recognize new behaviors

Have huge reach through capa integrations like VirusTotal

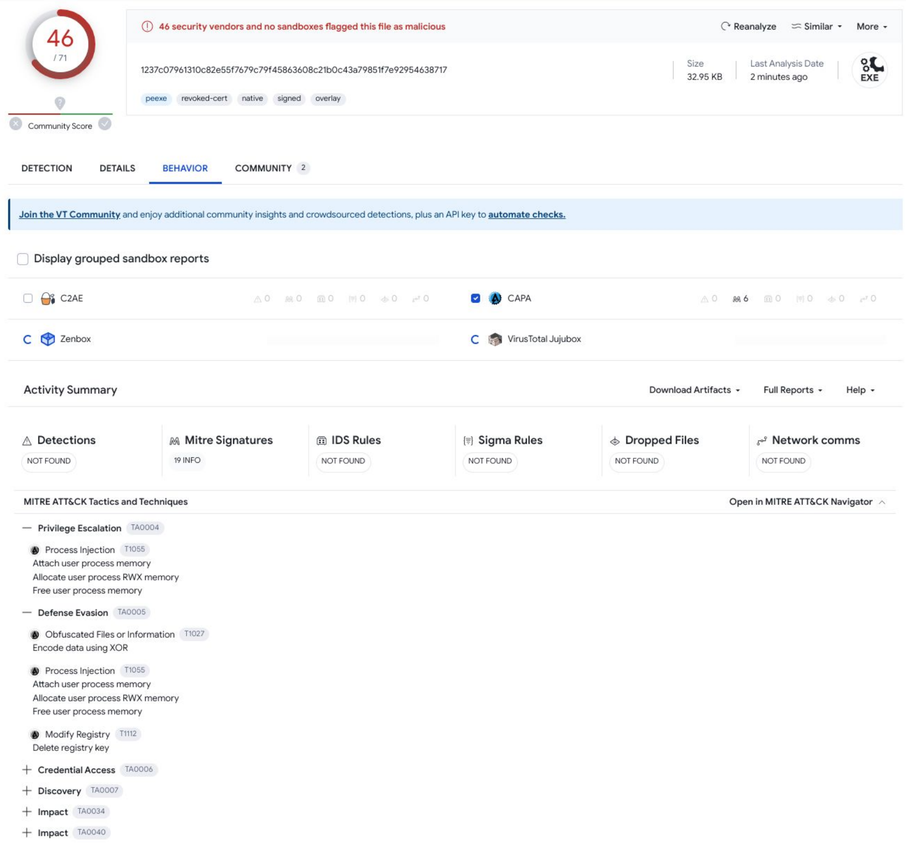
Serve as documentation of common malware techniques



```
rule:
  meta:
    name: create TCP socket
    namespace: communication/socket/tcp
    author: william.ballenthin@fireeye.com
    scope: function
    examples:
      - Practical Malware Analysis Lab 01-01.dll_:0x10001010
  features:
    - and:
      - number: 6 = IPPROTO_TCP
      - number: 1 = SOCK_STREAM
      - number: 2 = AF_INET
      - or:
        - api: ws2_32.socket
        - api: ws2_32.WSASocket
```

Google

how to create a tcp socket



46 / 71

Community Score

46 security vendors and no sandboxes flagged this file as malicious

Reanalyze · Similar · More

1237c07961310c82e55f7679c79f45863608c21b0c43a79851f7e92954638717

Size 32.95 KB

Last Analysis Date 2 minutes ago

EXE

peexe · revoked-cert · native · signed · overlay

DETECTION · DETAILS · BEHAVIOR · COMMUNITY 2

Join the VT Community and enjoy additional community insights and crowdsourced detections, plus an API key to automate checks.

□ Display grouped sandbox reports

□ C2AE 0 0 0 0 0 0

☑ CAPA 0 6 0 0 0 0

Zenbox

VirusTotal Jujubox

Activity Summary

Download Artifacts · Full Reports · Help

Detections	Mitre Signatures	IDS Rules	Sigma Rules	Dropped Files	Network comms
NOT FOUND	19 INFO	NOT FOUND	NOT FOUND	NOT FOUND	NOT FOUND

MITRE ATT&CK Tactics and Techniques

Open in MITRE ATT&CK Navigator

— Privilege Escalation TA0004

Process Injection T1055
Attach user process memory
Allocate user process RWX memory
Free user process memory

— Defense Evasion TA0005

Obfuscated Files or Information T1027
Encode data using XOR

Process Injection T1055
Attach user process memory
Allocate user process RWX memory
Free user process memory

Modify Registry T1112
Delete registry key

+ Credential Access TA0006

+ Discovery TA0007

+ Impact TA0034

+ Impact TA0040

writing a rule

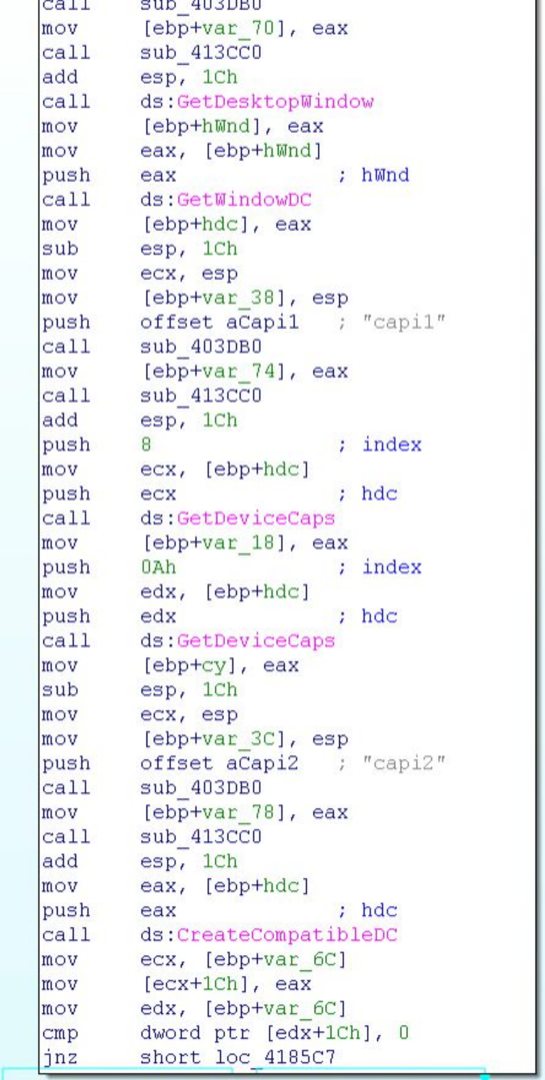
How to find a behavior to describe?

- You are reverse engineering and you notice a technique, so you encode it for your future self (and everyone else)
- You browse #good-first-issue and/or #help-wanted on github

What do you need to get started?

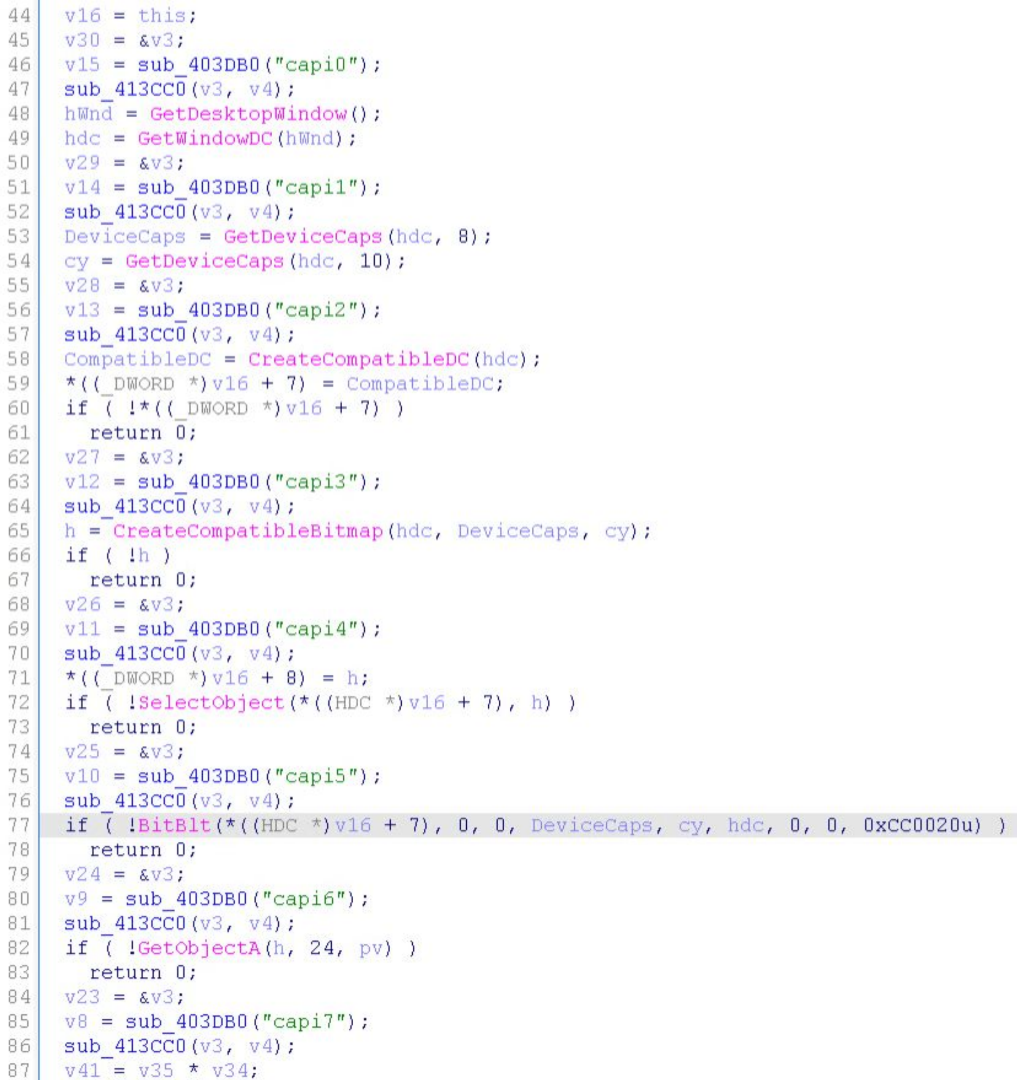
- Some idea of the features and logic that describe the behavior. API names, constants
- You may see this in your disassembler
- You may find a StackOverflow post or Github repository with a code snippet

example: disassembly



```
call    sub_403DB0
mov     [ebp+var_70], eax
call    sub_413CC0
add     esp, 1Ch
call    ds:GetDesktopWindow
mov     [ebp+hWnd], eax
mov     eax, [ebp+hWnd]
push    eax             ; hWnd
call    ds:GetWindowDC
mov     [ebp+hdc], eax
sub     esp, 1Ch
mov     ecx, esp
mov     [ebp+var_38], esp
push    offset aCapi1   ; "capi1"
call    sub_403DB0
mov     [ebp+var_74], eax
call    sub_413CC0
add     esp, 1Ch
push    8               ; index
mov     ecx, [ebp+hdc]
push    ecx             ; hdc
call    ds:GetDeviceCaps
mov     [ebp+var_18], eax
push    0Ah             ; index
mov     edx, [ebp+hdc]
push    edx             ; hdc
call    ds:GetDeviceCaps
mov     [ebp+cy], eax
sub     esp, 1Ch
mov     ecx, esp
mov     [ebp+var_3C], esp
push    offset aCapi2   ; "capi2"
call    sub_403DB0
mov     [ebp+var_78], eax
call    sub_413CC0
add     esp, 1Ch
mov     eax, [ebp+hdc]
push    eax             ; hdc
call    ds:CreateCompatibleDC
mov     ecx, [ebp+var_6C]
mov     [ecx+1Ch], eax
mov     edx, [ebp+var_6C]
cmp     dword ptr [edx+1Ch], 0
jnz     short loc_4185C7
```

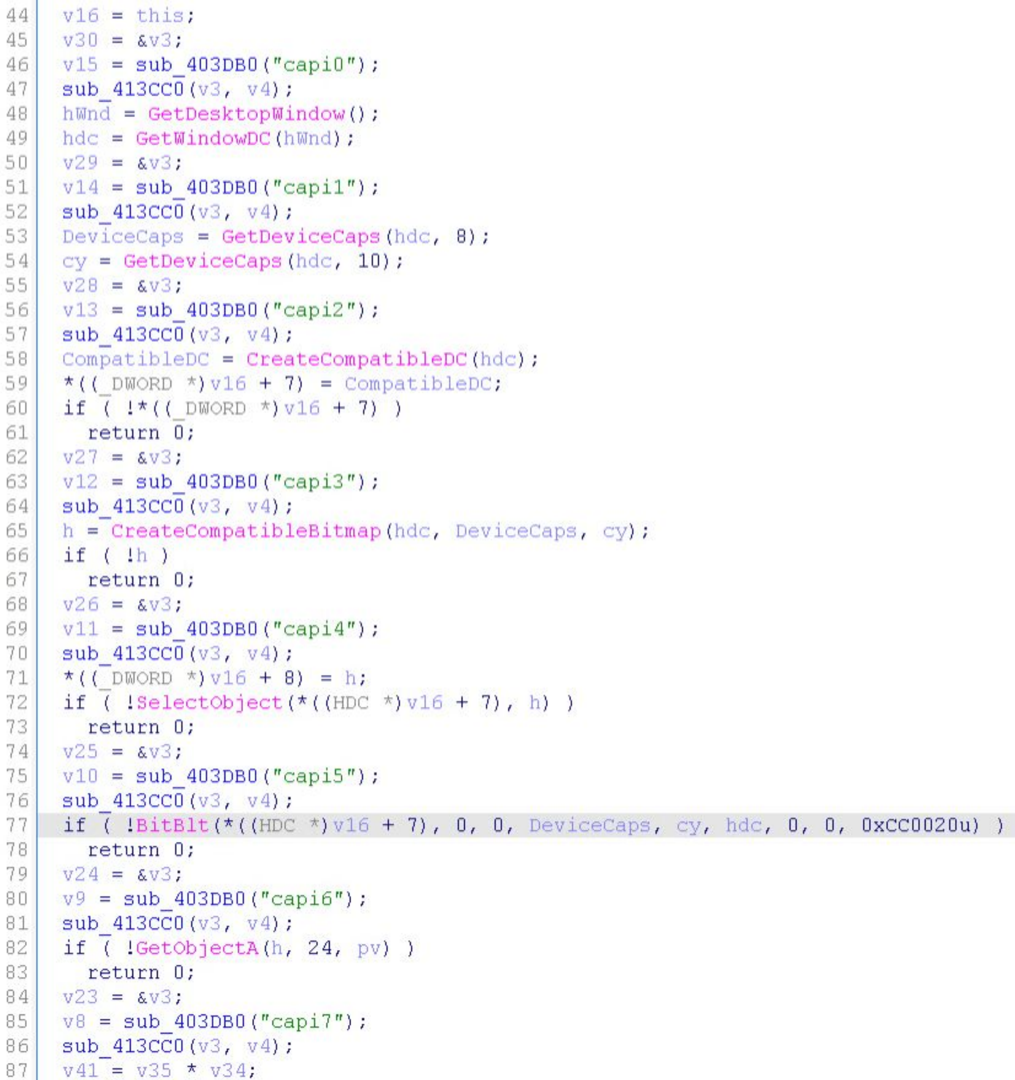
example: decompilation



```
v16 = this;
v30 = &v3;
v15 = sub_403DB0("capi0");
sub_413CC0(v3, v4);
hWnd = GetDesktopWindow();
hdc = GetWindowDC(hWnd);
v29 = &v3;
v14 = sub_403DB0("capi1");
sub_413CC0(v3, v4);
DeviceCaps = GetDeviceCaps(hdc, 8);
cy = GetDeviceCaps(hdc, 10);
v28 = &v3;
v13 = sub_403DB0("capi2");
sub_413CC0(v3, v4);
CompatibleDC = CreateCompatibleDC(hdc);
*((_DWORD *)v16 + 7) = CompatibleDC;
if ( !*((_DWORD *)v16 + 7) )
  return 0;
v27 = &v3;
v12 = sub_403DB0("capi3");
sub_413CC0(v3, v4);
h = CreateCompatibleBitmap(hdc, DeviceCaps, cy);
if ( !h )
  return 0;
v26 = &v3;
v11 = sub_403DB0("capi4");
sub_413CC0(v3, v4);
*((_DWORD *)v16 + 8) = h;
if ( !SelectObject(*((HDC *)v16 + 7), h) )
  return 0;
v25 = &v3;
v10 = sub_403DB0("capi5");
sub_413CC0(v3, v4);
if ( !BitBlt(*((HDC *)v16 + 7), 0, 0, DeviceCaps, cy, hdc, 0, 0, 0xCC0020u) )
  return 0;
v24 = &v3;
v9 = sub_403DB0("capi6");
sub_413CC0(v3, v4);
if ( !GetObjectA(h, 24, pv) )
  return 0;
v23 = &v3;
v8 = sub_403DB0("capi7");
sub_413CC0(v3, v4);
v41 = v35 * v34;
```

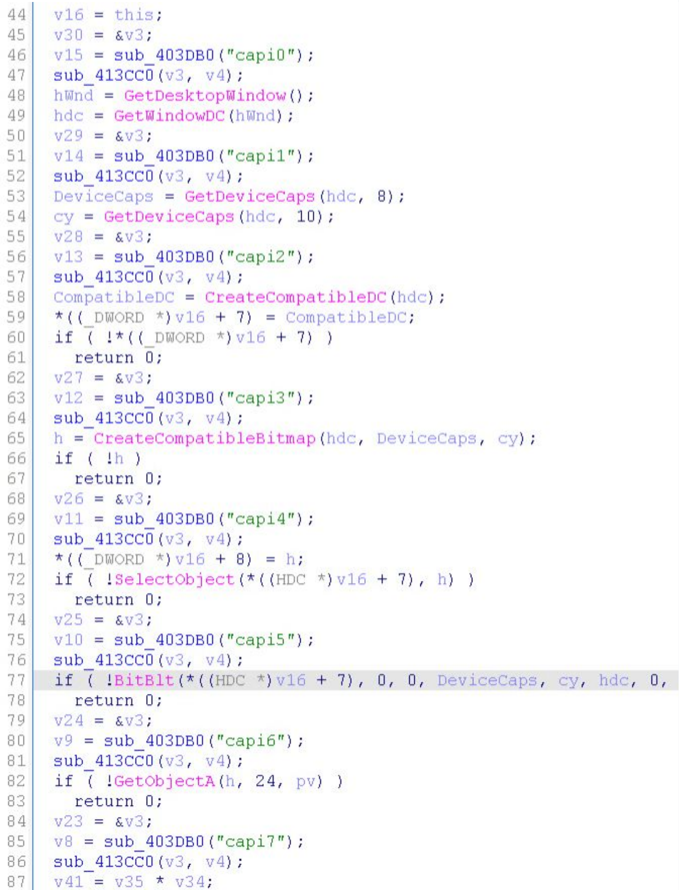
screenshot: candidate features

```
GetDesktopWindow(...)
GetWindowDC(...)
GetDeviceCaps(hdc, 8)    // HORZRES
GetDeviceCaps(hdc, 10)   // VERTRES
CreateCompatibleDC(...)
CreateCompatibleBitmap(...)
SelectObject(...)
BitBlt(...., 0xCC0020) //SRCCOPY
GetObject(...)
```



```
v16 = this;
v30 = &v3;
v15 = sub_403DB0("capi0");
sub_413CC0(v3, v4);
hWnd = GetDesktopWindow();
hdc = GetWindowDC(hWnd);
v29 = &v3;
v14 = sub_403DB0("capi1");
sub_413CC0(v3, v4);
DeviceCaps = GetDeviceCaps(hdc, 8);
cy = GetDeviceCaps(hdc, 10);
v28 = &v3;
v13 = sub_403DB0("capi2");
sub_413CC0(v3, v4);
CompatibleDC = CreateCompatibleDC(hdc);
*((_DWORD *)v16 + 7) = CompatibleDC;
if ( !*((_DWORD *)v16 + 7) )
  return 0;
v27 = &v3;
v12 = sub_403DB0("capi3");
sub_413CC0(v3, v4);
h = CreateCompatibleBitmap(hdc, DeviceCaps, cy);
if ( !h )
  return 0;
v26 = &v3;
v11 = sub_403DB0("capi4");
sub_413CC0(v3, v4);
*((_DWORD *)v16 + 8) = h;
if ( !SelectObject(*((HDC *)v16 + 7), h) )
  return 0;
v25 = &v3;
v10 = sub_403DB0("capi5");
sub_413CC0(v3, v4);
if ( !BitBlt(*((HDC *)v16 + 7), 0, 0, DeviceCaps, cy, hdc, 0, 0, 0xCC0020u) )
  return 0;
v24 = &v3;
v9 = sub_403DB0("capi6");
sub_413CC0(v3, v4);
if ( !GetObjectA(h, 24, pv) )
  return 0;
v23 = &v3;
v8 = sub_403DB0("capi7");
sub_413CC0(v3, v4);
v41 = v35 * v34;
```

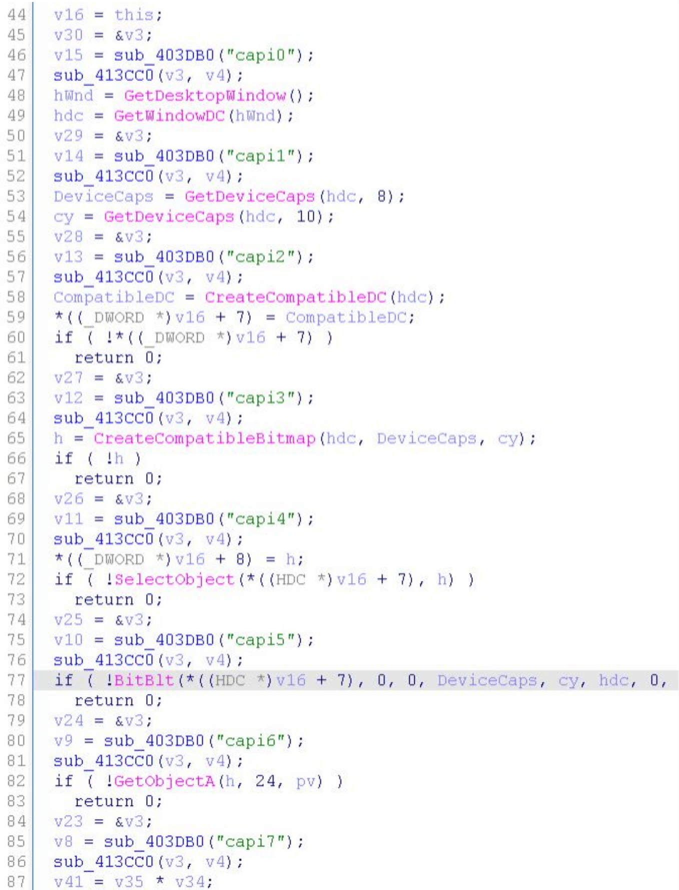
candidate features



```
v16 = this;
v30 = &v3;
v15 = sub_403DB0("capi0");
sub_413CC0(v3, v4);
hWnd = GetDesktopWindow();
hdc = GetWindowDC(hWnd);
v29 = &v3;
v14 = sub_403DB0("capi1");
sub_413CC0(v3, v4);
DeviceCaps = GetDeviceCaps(hdc, 8);
cy = GetDeviceCaps(hdc, 10);
v28 = &v3;
v13 = sub_403DB0("capi2");
sub_413CC0(v3, v4);
CompatibleDC = CreateCompatibleDC(hdc);
*((_DWORD *)v16 + 7) = CompatibleDC;
if ( !*((_DWORD *)v16 + 7) )
  return 0;
v27 = &v3;
v12 = sub_403DB0("capi3");
sub_413CC0(v3, v4);
h = CreateCompatibleBitmap(hdc, DeviceCaps, cy);
if ( !h )
  return 0;
v26 = &v3;
v11 = sub_403DB0("capi4");
sub_413CC0(v3, v4);
*((_DWORD *)v16 + 8) = h;
if ( !SelectObject(*((HDC *)v16 + 7), h) )
  return 0;
v25 = &v3;
v10 = sub_403DB0("capi5");
sub_413CC0(v3, v4);
if ( !BitBlt(*((HDC *)v16 + 7), 0, 0, DeviceCaps, cy, hdc, 0,
  return 0;
v24 = &v3;
v9 = sub_403DB0("capi6");
sub_413CC0(v3, v4);
if ( !GetObjectA(h, 24, pv) )
  return 0;
v23 = &v3;
v8 = sub_403DB0("capi7");
sub_413CC0(v3, v4);
v41 = v35 * v34;
```

```
and:
  - api: GetDesktopWindow
  - api: GetWindowDC
  - api: GetDeviceCaps
  - api: GetDeviceCaps
  - api: CreateCompatibleDC
  - api: CreateCompatibleBitmap
  - api: SelectObject
  - api: BitBlt
  - api: GetObject
```

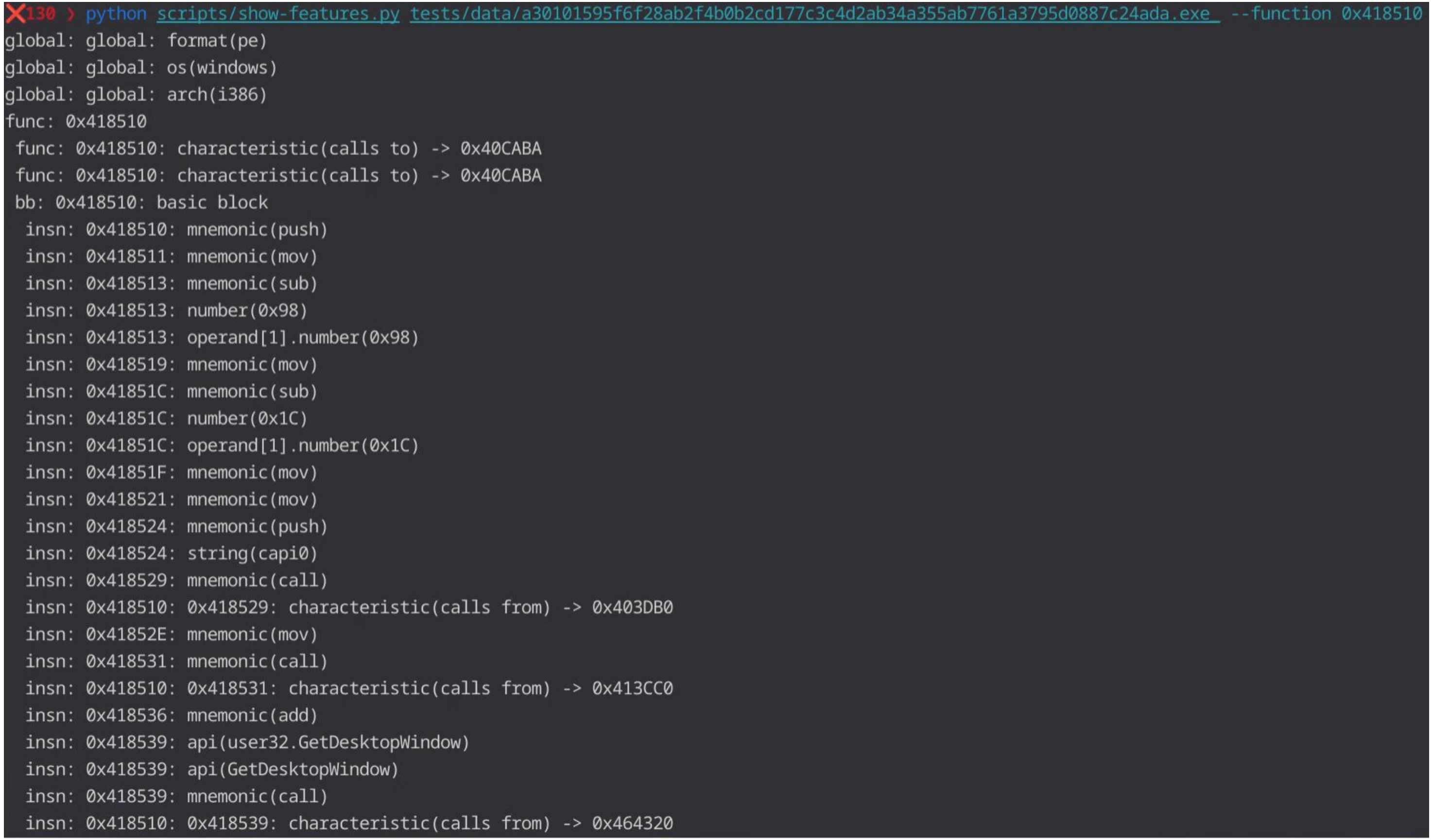
logic nodes



```
v16 = this;
v30 = &v3;
v15 = sub_403DB0("capi0");
sub_413CC0(v3, v4);
hWnd = GetDesktopWindow();
hdc = GetWindowDC(hWnd);
v29 = &v3;
v14 = sub_403DB0("capi1");
sub_413CC0(v3, v4);
DeviceCaps = GetDeviceCaps(hdc, 8);
cy = GetDeviceCaps(hdc, 10);
v28 = &v3;
v13 = sub_403DB0("capi2");
sub_413CC0(v3, v4);
CompatibleDC = CreateCompatibleDC(hdc);
*((_DWORD *)v16 + 7) = CompatibleDC;
if ( !*((_DWORD *)v16 + 7) )
  return 0;
v27 = &v3;
v12 = sub_403DB0("capi3");
sub_413CC0(v3, v4);
h = CreateCompatibleBitmap(hdc, DeviceCaps, cy);
if ( !h )
  return 0;
v26 = &v3;
v11 = sub_403DB0("capi4");
sub_413CC0(v3, v4);
*((_DWORD *)v16 + 8) = h;
if ( !SelectObject(*((HDC *)v16 + 7), h) )
  return 0;
v25 = &v3;
v10 = sub_403DB0("capi5");
sub_413CC0(v3, v4);
if ( !BitBlt(*((HDC *)v16 + 7), 0, 0, DeviceCaps, cy, hdc, 0,
  return 0;
v24 = &v3;
v9 = sub_403DB0("capi6");
sub_413CC0(v3, v4);
if ( !GetObjectA(h, 24, pv) )
  return 0;
v23 = &v3;
v8 = sub_403DB0("capi7");
sub_413CC0(v3, v4);
v41 = v35 * v34;
```

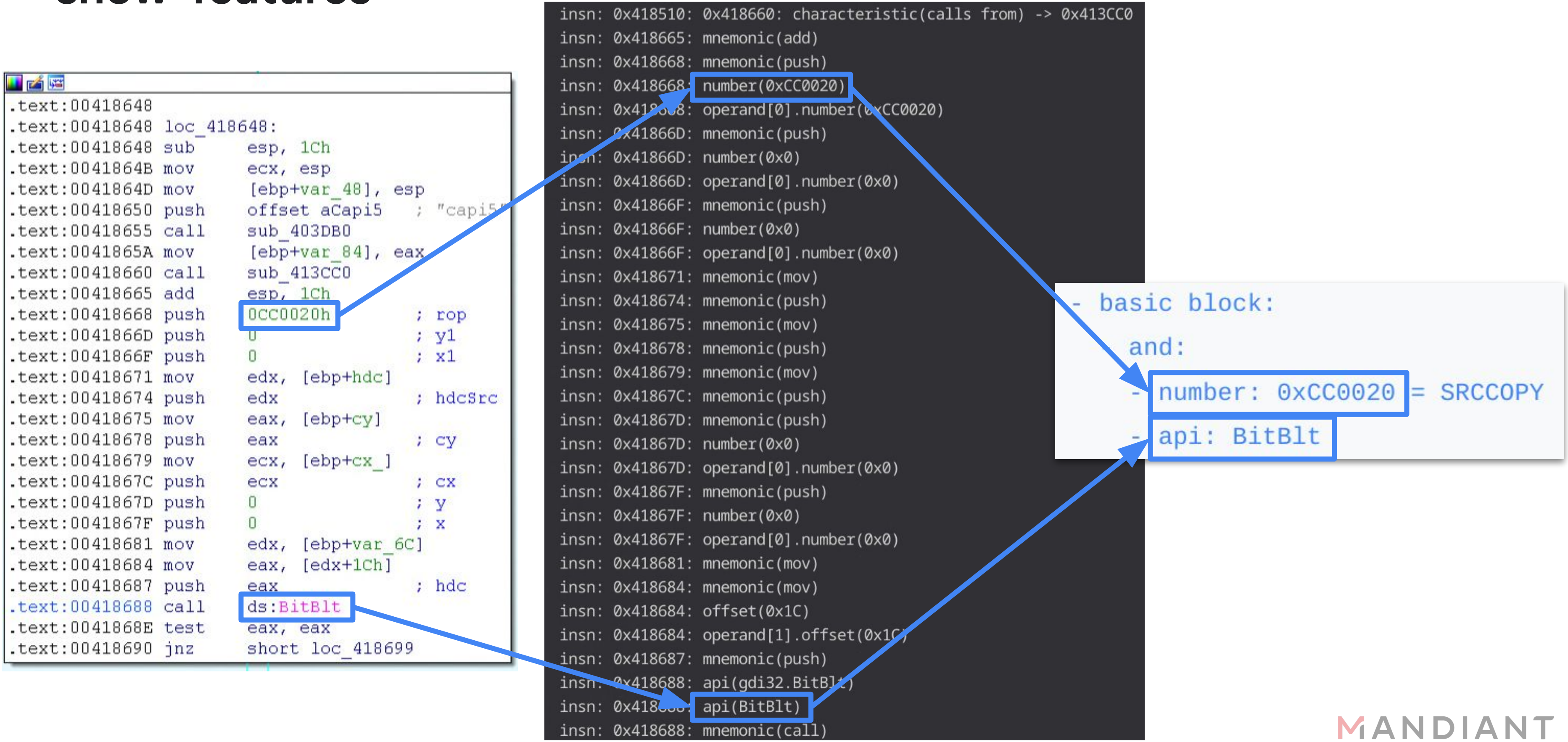
```
and:
  - api: GetDesktopWindow
  - api: GetWindowDC
  - and:
    - api: GetDeviceCaps
    - number: 8
  - and:
    - api: GetDeviceCaps
    - number: 10
  - api: CreateCompatibleDC
  - api: CreateCompatibleBitmap
  - api: SelectObject
  - and:
    - api: BitBlt
    - number: 0xCC0020
  - api: GetObject
```

show-features



```
✕130 ❯ python scripts/show-features.py tests/data/a30101595f6f28ab2f4b0b2cd177c3c4d2ab34a355ab7761a3795d0887c24ada.exe_ --function 0x418510
global: global: format(pe)
global: global: os(windows)
global: global: arch(i386)
func: 0x418510
 func: 0x418510: characteristic(calls to) -> 0x40CABA
 func: 0x418510: characteristic(calls to) -> 0x40CABA
 bb: 0x418510: basic block
  insn: 0x418510: mnemonic(push)
  insn: 0x418511: mnemonic(mov)
  insn: 0x418513: mnemonic(sub)
  insn: 0x418513: number(0x98)
  insn: 0x418513: operand[1].number(0x98)
  insn: 0x418519: mnemonic(mov)
  insn: 0x41851C: mnemonic(sub)
  insn: 0x41851C: number(0x1C)
  insn: 0x41851C: operand[1].number(0x1C)
  insn: 0x41851F: mnemonic(mov)
  insn: 0x418521: mnemonic(mov)
  insn: 0x418524: mnemonic(push)
  insn: 0x418524: string(capi0)
  insn: 0x418529: mnemonic(call)
  insn: 0x418510: 0x418529: characteristic(calls from) -> 0x403DB0
  insn: 0x41852E: mnemonic(mov)
  insn: 0x418531: mnemonic(call)
  insn: 0x418510: 0x418531: characteristic(calls from) -> 0x413CC0
  insn: 0x418536: mnemonic(add)
  insn: 0x418539: api(user32.GetDesktopWindow)
  insn: 0x418539: api(GetDesktopWindow)
  insn: 0x418539: mnemonic(call)
  insn: 0x418510: 0x418539: characteristic(calls from) -> 0x464320
```

show-features

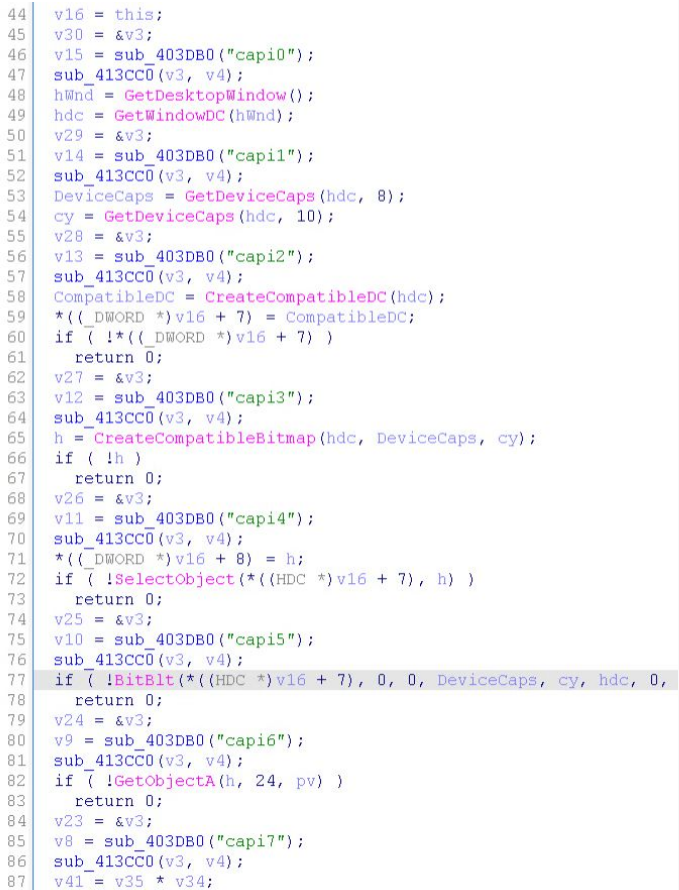


```
.text:00418648
.text:00418648 loc_418648:
.text:00418648 sub     esp, 1Ch
.text:0041864B mov     ecx, esp
.text:0041864D mov     [ebp+var_48], esp
.text:00418650 push    offset aCapi5    ; "capi5
.text:00418655 call    sub_403DB0
.text:0041865A mov     [ebp+var_84], eax
.text:00418660 call    sub_413CC0
.text:00418665 add     esp, 1Ch
.text:00418668 push    0CC0020h         ; rop
.text:0041866D push    0                ; y1
.text:0041866F push    0                ; x1
.text:00418671 mov     edx, [ebp+hdc]
.text:00418674 push    edx              ; hdcSrc
.text:00418675 mov     eax, [ebp+cy]
.text:00418678 push    eax              ; cy
.text:00418679 mov     ecx, [ebp+cx_]
.text:0041867C push    ecx              ; cx
.text:0041867D push    0                ; y
.text:0041867F push    0                ; x
.text:00418681 mov     edx, [ebp+var_6C]
.text:00418684 mov     eax, [edx+1Ch]
.text:00418687 push    eax              ; hdc
.text:00418688 call    ds:BitBlt
.text:0041868E test    eax, eax
.text:00418690 jnz     short loc_418699
```

```
insn: 0x418510: 0x418660: characteristic(calls from) -> 0x413CC0
insn: 0x418665: mnemonic(add)
insn: 0x418668: mnemonic(push)
insn: 0x418668: number(0xCC0020)
insn: 0x418668: operand[0].number(0xCC0020)
insn: 0x41866D: mnemonic(push)
insn: 0x41866D: number(0x0)
insn: 0x41866D: operand[0].number(0x0)
insn: 0x41866F: mnemonic(push)
insn: 0x41866F: number(0x0)
insn: 0x41866F: operand[0].number(0x0)
insn: 0x418671: mnemonic(mov)
insn: 0x418674: mnemonic(push)
insn: 0x418675: mnemonic(mov)
insn: 0x418678: mnemonic(push)
insn: 0x418679: mnemonic(mov)
insn: 0x41867C: mnemonic(push)
insn: 0x41867D: mnemonic(push)
insn: 0x41867D: number(0x0)
insn: 0x41867D: operand[0].number(0x0)
insn: 0x41867F: mnemonic(push)
insn: 0x41867F: number(0x0)
insn: 0x41867F: operand[0].number(0x0)
insn: 0x418681: mnemonic(mov)
insn: 0x418684: mnemonic(mov)
insn: 0x418684: offset(0x1C)
insn: 0x418684: operand[1].offset(0x1C)
insn: 0x418687: mnemonic(push)
insn: 0x418688: api(gdi32.BitBlt)
insn: 0x418688: api(BitBlt)
insn: 0x418688: mnemonic(call)
```

```
- basic block:
  - and:
    - number: 0xCC0020 = SRCCOPY
    - api: BitBlt
```

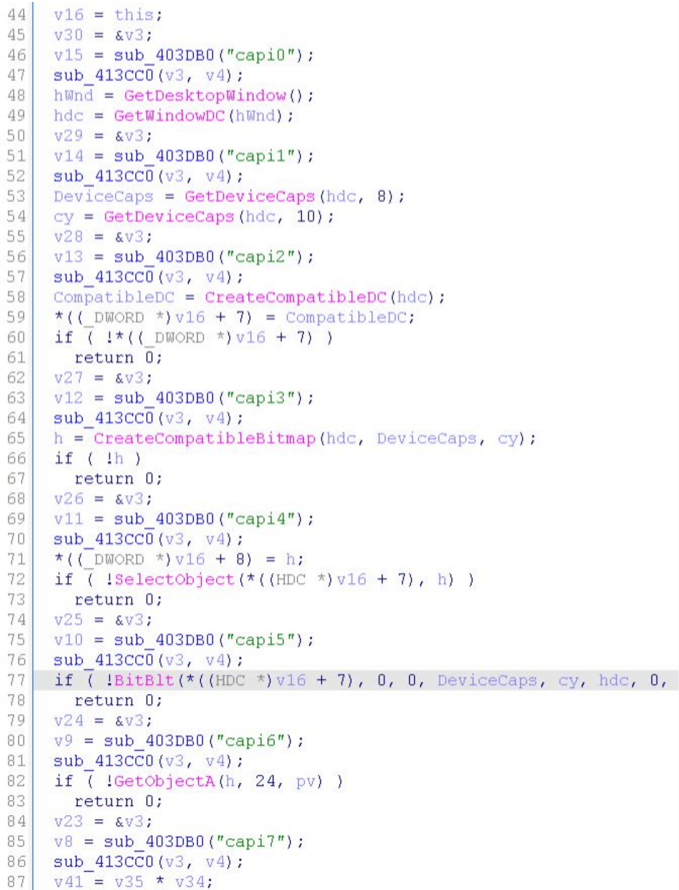
comments & symbols



```
v16 = this;
v30 = &v3;
v15 = sub_403DB0("capi0");
sub_413CC0(v3, v4);
hWnd = GetDesktopWindow();
hdc = GetWindowDC(hWnd);
v29 = &v3;
v14 = sub_403DB0("capi1");
sub_413CC0(v3, v4);
DeviceCaps = GetDeviceCaps(hdc, 8);
cy = GetDeviceCaps(hdc, 10);
v28 = &v3;
v13 = sub_403DB0("capi2");
sub_413CC0(v3, v4);
CompatibleDC = CreateCompatibleDC(hdc);
*((_DWORD *)v16 + 7) = CompatibleDC;
if ( !*((_DWORD *)v16 + 7) )
  return 0;
v27 = &v3;
v12 = sub_403DB0("capi3");
sub_413CC0(v3, v4);
h = CreateCompatibleBitmap(hdc, DeviceCaps, cy);
if ( !h )
  return 0;
v26 = &v3;
v11 = sub_403DB0("capi4");
sub_413CC0(v3, v4);
*((_DWORD *)v16 + 8) = h;
if ( !SelectObject(*((HDC *)v16 + 7), h) )
  return 0;
v25 = &v3;
v10 = sub_403DB0("capi5");
sub_413CC0(v3, v4);
if ( !BitBlt(*((HDC *)v16 + 7), 0, 0, DeviceCaps, cy, hdc, 0,
  return 0;
v24 = &v3;
v9 = sub_403DB0("capi6");
sub_413CC0(v3, v4);
if ( !GetObjectA(h, 24, pv) )
  return 0;
v23 = &v3;
v8 = sub_403DB0("capi7");
sub_413CC0(v3, v4);
v41 = v35 * v34;
```

```
and:
  - api: GetDesktopWindow
  - api: GetWindowDC
  - and:
    - api: GetDeviceCaps
    - number: 8 = HORZRES
  - and:
    - api: GetDeviceCaps
    - number: 10 = VERTRES
  - api: CreateCompatibleDC
  - api: CreateCompatibleBitmap
  - api: SelectObject
  - and:
    - api: BitBlt
    - number: 0xCC0020 = SRCCOPY
  - api: GetObject
```

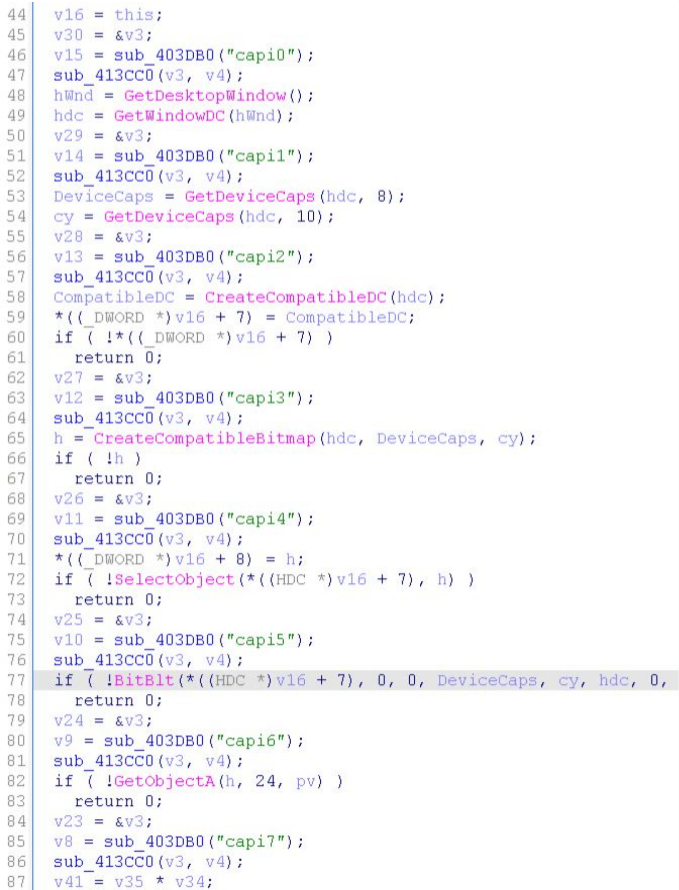
comments & symbols



```
v16 = this;
v30 = &v3;
v15 = sub_403DB0("capi0");
sub_413CC0(v3, v4);
hWnd = GetDesktopWindow();
hdc = GetWindowDC(hWnd);
v29 = &v3;
v14 = sub_403DB0("capi1");
sub_413CC0(v3, v4);
DeviceCaps = GetDeviceCaps(hdc, 8);
cy = GetDeviceCaps(hdc, 10);
v28 = &v3;
v13 = sub_403DB0("capi2");
sub_413CC0(v3, v4);
CompatibleDC = CreateCompatibleDC(hdc);
*((_DWORD *)v16 + 7) = CompatibleDC;
if ( !*((_DWORD *)v16 + 7) )
  return 0;
v27 = &v3;
v12 = sub_403DB0("capi3");
sub_413CC0(v3, v4);
h = CreateCompatibleBitmap(hdc, DeviceCaps, cy);
if ( !h )
  return 0;
v26 = &v3;
v11 = sub_403DB0("capi4");
sub_413CC0(v3, v4);
*((_DWORD *)v16 + 8) = h;
if ( !SelectObject(*((HDC *)v16 + 7), h) )
  return 0;
v25 = &v3;
v10 = sub_403DB0("capi5");
sub_413CC0(v3, v4);
if ( !BitBlt(*((HDC *)v16 + 7), 0, 0, DeviceCaps, cy, hdc, 0,
  return 0;
v24 = &v3;
v9 = sub_403DB0("capi6");
sub_413CC0(v3, v4);
if ( !GetObjectA(h, 24, pv) )
  return 0;
v23 = &v3;
v8 = sub_403DB0("capi7");
sub_413CC0(v3, v4);
v41 = v35 * v34;
```

```
and:
  - api: GetDesktopWindow
  - api: GetWindowDC
  - and:
    - api: GetDeviceCaps
    - number: 8 = HORZRES
  - and:
    - api: GetDeviceCaps
    - number: 10 = VERTRES
  - api: CreateCompatibleDC
  - api: CreateCompatibleBitmap
  - api: SelectObject
  - basic block:
    - and:
      - api: BitBlt
      - number: 0xCC0020 = SRCCOPY
  - api: GetObject
```

comments & symbols



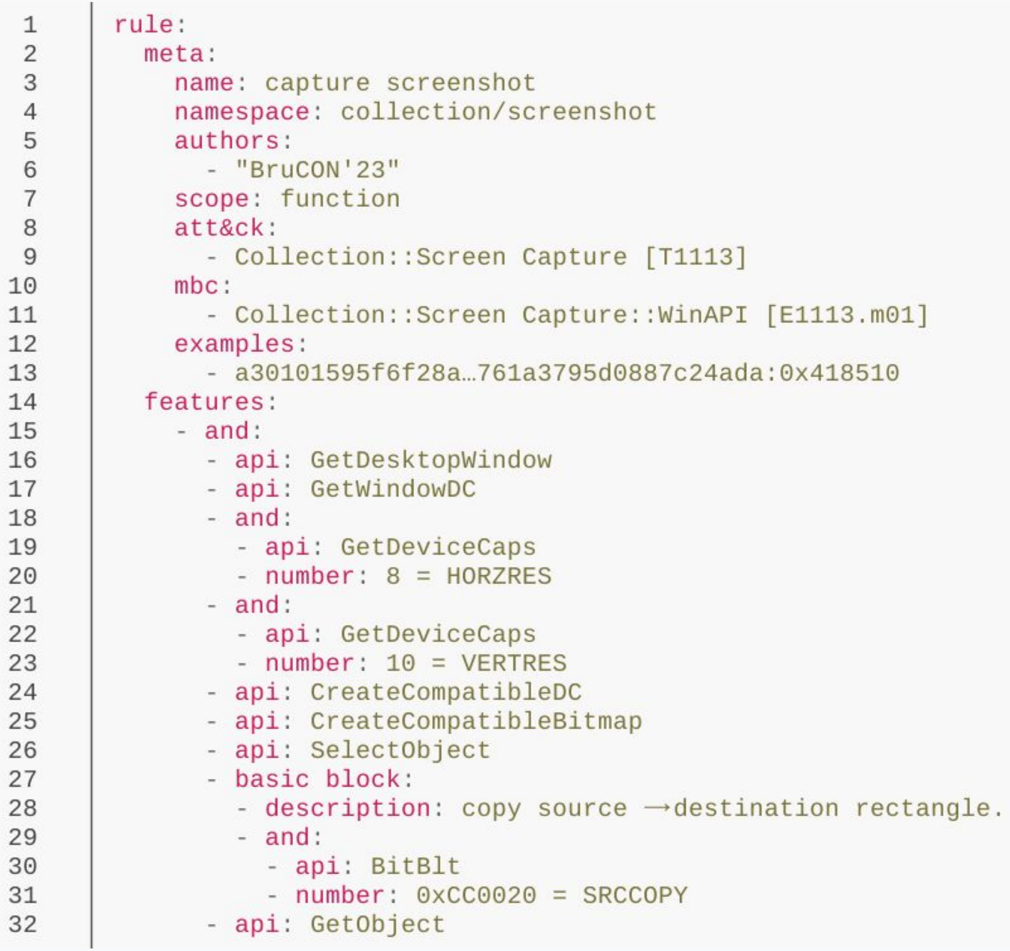
```
v16 = this;
v30 = &v3;
v15 = sub_403DB0("capi0");
sub_413CC0(v3, v4);
hWnd = GetDesktopWindow();
hdc = GetWindowDC(hWnd);
v29 = &v3;
v14 = sub_403DB0("capi1");
sub_413CC0(v3, v4);
DeviceCaps = GetDeviceCaps(hdc, 8);
cy = GetDeviceCaps(hdc, 10);
v28 = &v3;
v13 = sub_403DB0("capi2");
sub_413CC0(v3, v4);
CompatibleDC = CreateCompatibleDC(hdc);
*((_DWORD *)v16 + 7) = CompatibleDC;
if ( !*((_DWORD *)v16 + 7) )
  return 0;
v27 = &v3;
v12 = sub_403DB0("capi3");
sub_413CC0(v3, v4);
h = CreateCompatibleBitmap(hdc, DeviceCaps, cy);
if ( !h )
  return 0;
v26 = &v3;
v11 = sub_403DB0("capi4");
sub_413CC0(v3, v4);
*((_DWORD *)v16 + 8) = h;
if ( !SelectObject(*((HDC *)v16 + 7), h) )
  return 0;
v25 = &v3;
v10 = sub_403DB0("capi5");
sub_413CC0(v3, v4);
if ( !BitBlt(*((HDC *)v16 + 7), 0, 0, DeviceCaps, cy, hdc, 0,
  return 0;
v24 = &v3;
v9 = sub_403DB0("capi6");
sub_413CC0(v3, v4);
if ( !GetObjectA(h, 24, pv) )
  return 0;
v23 = &v3;
v8 = sub_403DB0("capi7");
sub_413CC0(v3, v4);
v41 = v35 * v34;
```

```
and:
  - api: GetDesktopWindow
  - api: GetWindowDC
  - and:
    - api: GetDeviceCaps
    - number: 8 = HORZRES
  - and:
    - api: GetDeviceCaps
    - number: 10 = VERTRES
  - api: CreateCompatibleDC
  - api: CreateCompatibleBitmap
  - api: SelectObject
  - basic block:
    - description: copy source → destination rectangle.
    - and:
      - api: BitBlt
      - number: 0xCC0020 = SRCCOPY
  - api: GetObject
```

rule metadata

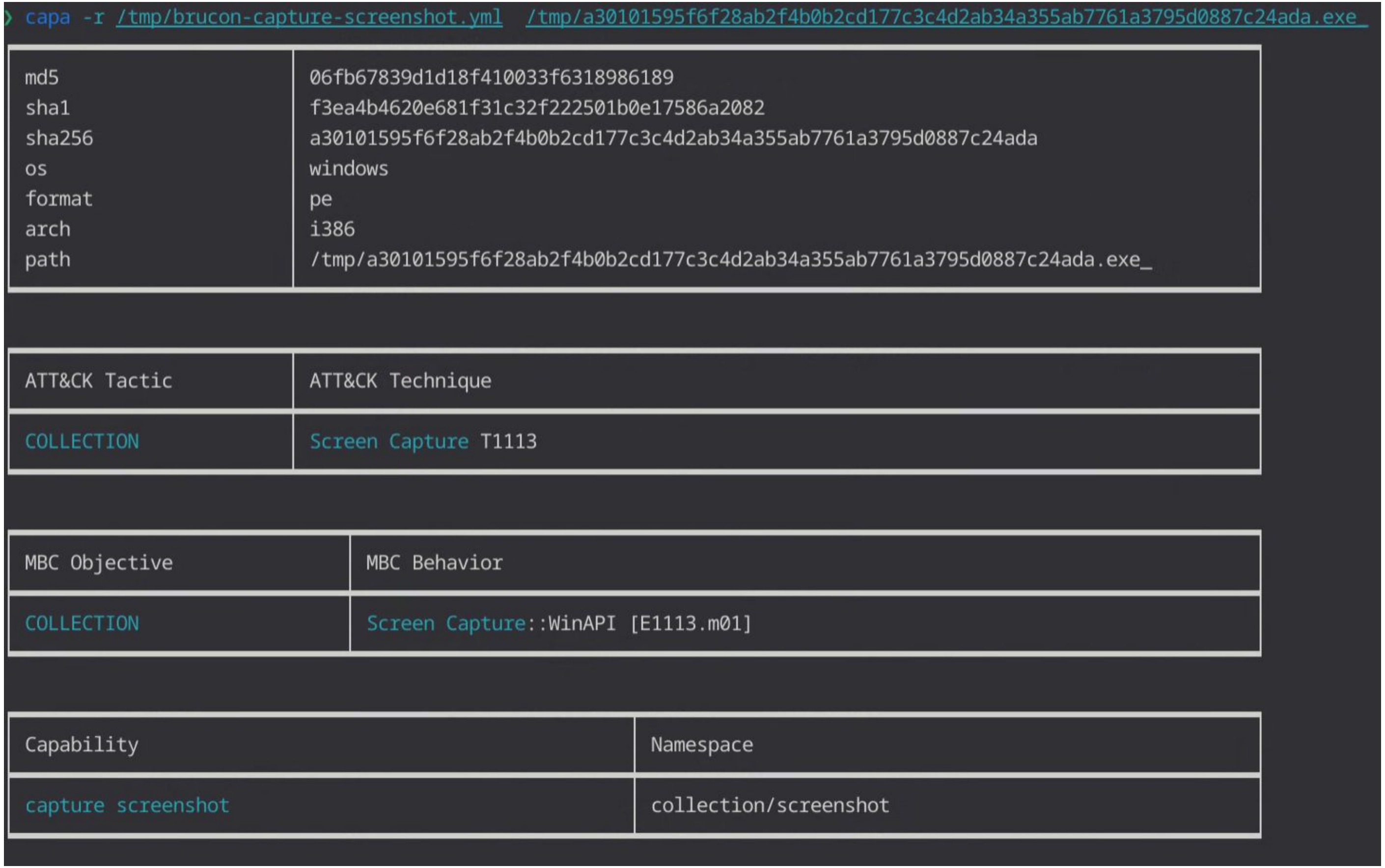
```
meta:
  name: capture screenshot
  namespace: collection/screenshot
  authors:
    - BruCON’23
  scope: function
  att&ck:
    - Collection::Screen Capture [T1113]
  mbc:
    - Collection::Screen Capture::WinAPI [E1113.m01]
  examples:
    - a30101595f6f28a…761a3795d0887c24ada:0x418510
```

final rule



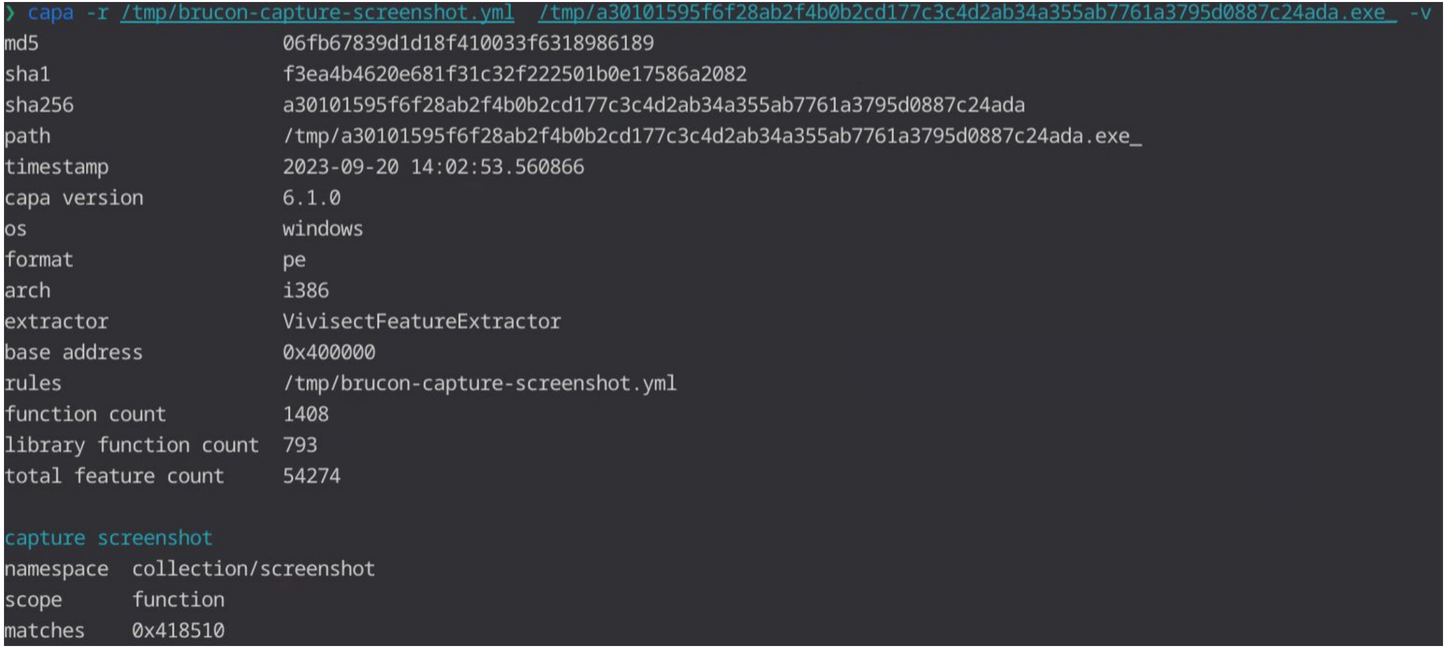
```
rule:
  meta:
    name: capture screenshot
    namespace: collection/screenshot
    authors:
      - "BruCON'23"
    scope: function
    att&ck:
      - Collection::Screen Capture [T1113]
    mbc:
      - Collection::Screen Capture::WinAPI [E1113.m01]
    examples:
      - a30101595f6f28a…761a3795d0887c24ada:0x418510
  features:
    - and:
      - api: GetDesktopWindow
      - api: GetWindowDC
      - and:
        - api: GetDeviceCaps
        - number: 8 = HORZRES
      - and:
        - api: GetDeviceCaps
        - number: 10 = VERTRES
      - api: CreateCompatibleDC
      - api: CreateCompatibleBitmap
      - api: SelectObject
      - basic block:
        - description: copy source →destination rectangle.
        - and:
          - api: BitBlt
          - number: 0xCC0020 = SRCCOPY
      - api: GetObject
```

setting the rule path



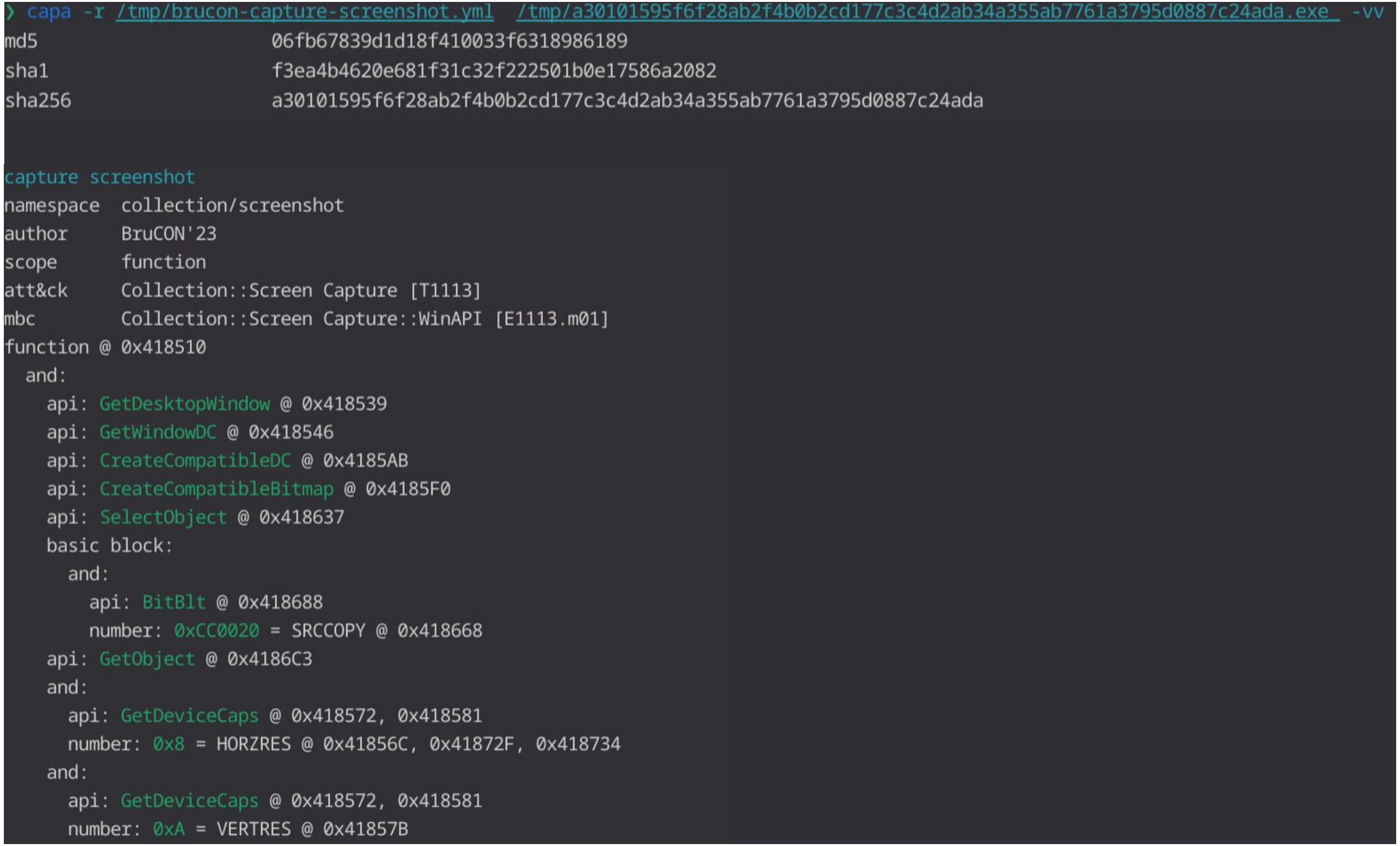
```
❯ capa -r /tmp/brucon-capture-screenshot.yml  /tmp/a30101595f6f28ab2f4b0b2cd177c3c4d2ab34a355ab7761a3795d0887c24ada.exe_

 md5                      06fb67839d1d18f410033f6318986189
 sha1                     f3ea4b4620e681f31c32f222501b0e17586a2082
 sha256                   a30101595f6f28ab2f4b0b2cd177c3c4d2ab34a355ab7761a3795d0887c24ada
 os                       windows
 format                   pe
 arch                     i386
 path                     /tmp/a30101595f6f28ab2f4b0b2cd177c3c4d2ab34a355ab7761a3795d0887c24ada.exe_


 ATT&CK Tactic            ATT&CK Technique

 COLLECTION               Screen Capture T1113


 MBC Objective                MBC Behavior

 COLLECTION                   Screen Capture::WinAPI [E1113.m01]


 Capability                                Namespace

 capture screenshot                        collection/screenshot
```

-v



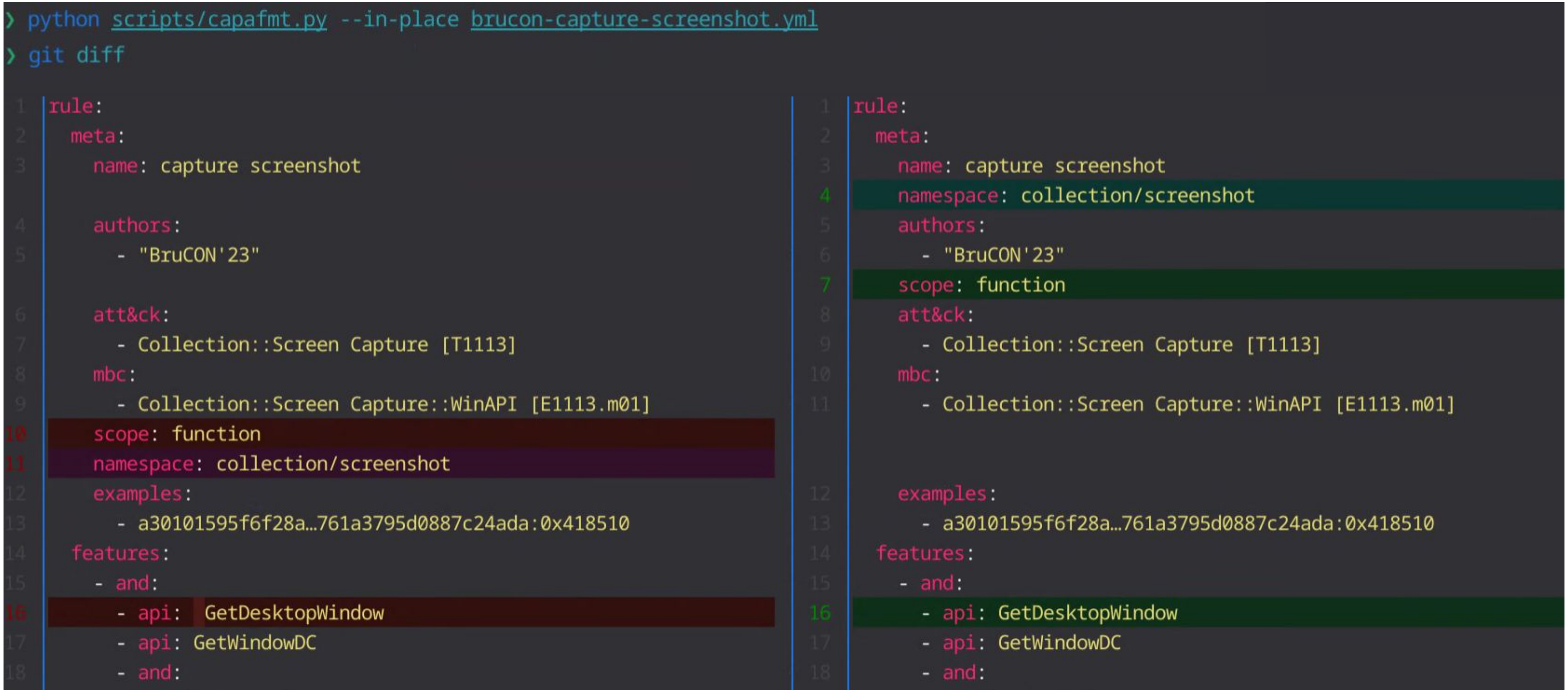
```
❯ capa -r /tmp/brucon-capture-screenshot.yml  /tmp/a30101595f6f28ab2f4b0b2cd177c3c4d2ab34a355ab7761a3795d0887c24ada.exe_  -v
md5                     06fb67839d1d18f410033f6318986189
sha1                    f3ea4b4620e681f31c32f222501b0e17586a2082
sha256                  a30101595f6f28ab2f4b0b2cd177c3c4d2ab34a355ab7761a3795d0887c24ada
path                    /tmp/a30101595f6f28ab2f4b0b2cd177c3c4d2ab34a355ab7761a3795d0887c24ada.exe_
timestamp               2023-09-20 14:02:53.560866
capa version            6.1.0
os                      windows
format                  pe
arch                    i386
extractor               VivisectFeatureExtractor
base address            0x400000
rules                   /tmp/brucon-capture-screenshot.yml
function count          1408
library function count  793
total feature count     54274

capture screenshot
namespace  collection/screenshot
scope      function
matches    0x418510
```

-vv



```
❯ capa -r /tmp/brucon-capture-screenshot.yml  /tmp/a30101595f6f28ab2f4b0b2cd177c3c4d2ab34a355ab7761a3795d0887c24ada.exe  -vv
md5                      06fb67839d1d18f410033f6318986189
sha1                     f3ea4b4620e681f31c32f222501b0e17586a2082
sha256                   a30101595f6f28ab2f4b0b2cd177c3c4d2ab34a355ab7761a3795d0887c24ada


capture screenshot
namespace  collection/screenshot
author     BruCON'23
scope      function
att&ck     Collection::Screen Capture [T1113]
mbc        Collection::Screen Capture::WinAPI [E1113.m01]
function @ 0x418510
  and:
    api: GetDesktopWindow @ 0x418539
    api: GetWindowDC @ 0x418546
    api: CreateCompatibleDC @ 0x4185AB
    api: CreateCompatibleBitmap @ 0x4185F0
    api: SelectObject @ 0x418637
    basic block:
      and:
        api: BitBlt @ 0x418688
        number: 0xCC0020 = SRCCOPY @ 0x418668
    api: GetObject @ 0x4186C3
    and:
      api: GetDeviceCaps @ 0x418572, 0x418581
      number: 0x8 = HORZRES @ 0x41856C, 0x41872F, 0x418734
    and:
      api: GetDeviceCaps @ 0x418572, 0x418581
      number: 0xA = VERTRES @ 0x41857B
```

capafmt

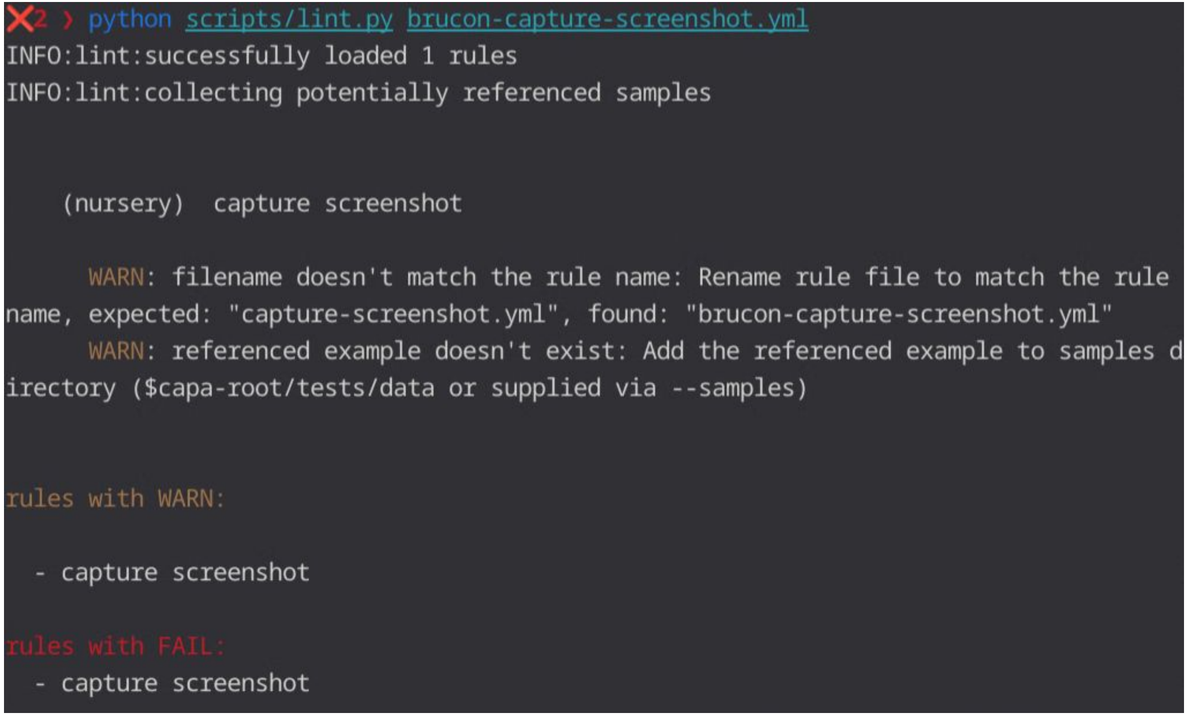


```
❯ python scripts/capafmt.py --in-place brucon-capture-screenshot.yml
❯ git diff
```

```
rule:
  meta:
    name: capture screenshot

    authors:
      - "BruCON'23"

    att&ck:
      - Collection::Screen Capture [T1113]
    mbc:
      - Collection::Screen Capture::WinAPI [E1113.m01]
    scope: function
    namespace: collection/screenshot
    examples:
      - a30101595f6f28a…761a3795d0887c24ada:0x418510
  features:
    - and:
      - api:  GetDesktopWindow
      - api: GetWindowDC
      - and:
```

```
rule:
  meta:
    name: capture screenshot
    namespace: collection/screenshot
    authors:
      - "BruCON'23"
    scope: function
    att&ck:
      - Collection::Screen Capture [T1113]
    mbc:
      - Collection::Screen Capture::WinAPI [E1113.m01]


    examples:
      - a30101595f6f28a…761a3795d0887c24ada:0x418510
  features:
    - and:
      - api: GetDesktopWindow
      - api: GetWindowDC
      - and:
```

rule linter



```
✗2 ❯ python scripts/lint.py brucon-capture-screenshot.yml
INFO:lint:successfully loaded 1 rules
INFO:lint:collecting potentially referenced samples



    (nursery)  capture screenshot

      WARN: filename doesn't match the rule name: Rename rule file to match the rule
name, expected: "capture-screenshot.yml", found: "brucon-capture-screenshot.yml"
      WARN: referenced example doesn't exist: Add the referenced example to samples d
irectory ($capa-root/tests/data or supplied via --samples)



rules with WARN:

  - capture screenshot

rules with FAIL:
  - capture screenshot
```

lab three

writing capa rules

lab three writing capa rules

For each of the following behaviors and samples

A. Persisting via a registry run key, `3f8e2b945deba235fa4888682bd0d640`
B. Writing to a file, `625ac05fd47adc3c63700c3b30de79ab`
C. Creating a TCP socket, `290934c61de9176ad682ffdd65f0a669`

Write a capa rule that matches against sample, and consider:

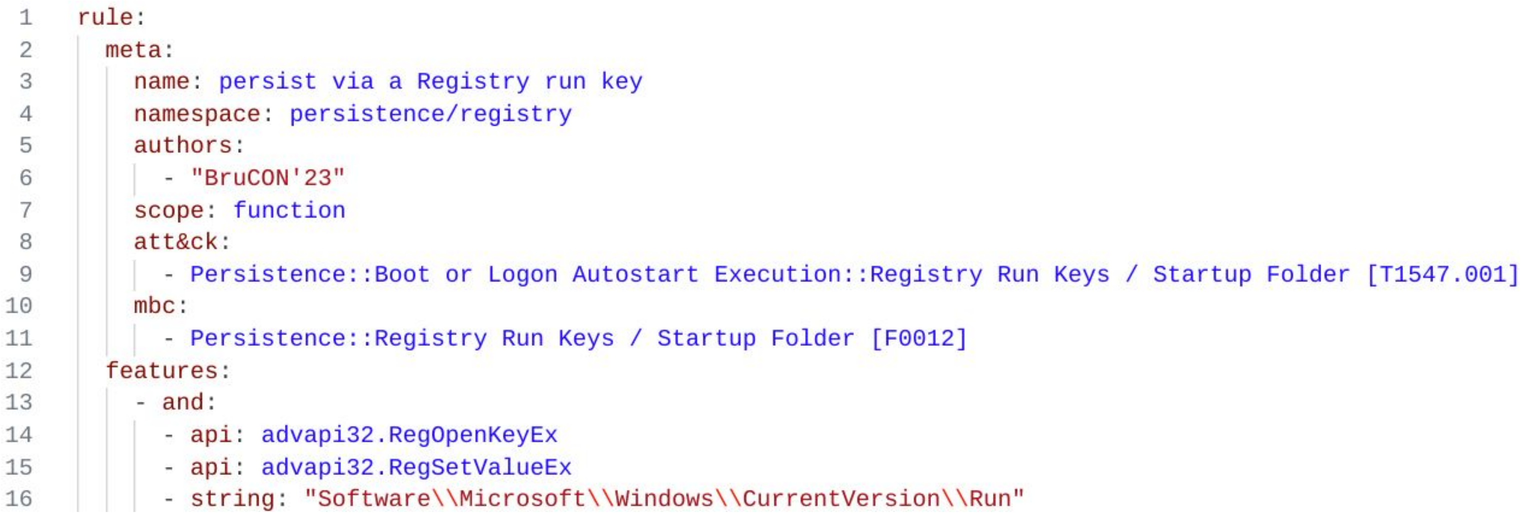
1. What features did you reference? Are there any alternatives?
2. Which scope did you use? Why?
3. Can you write a yara rule for this?

lab three answers (a)

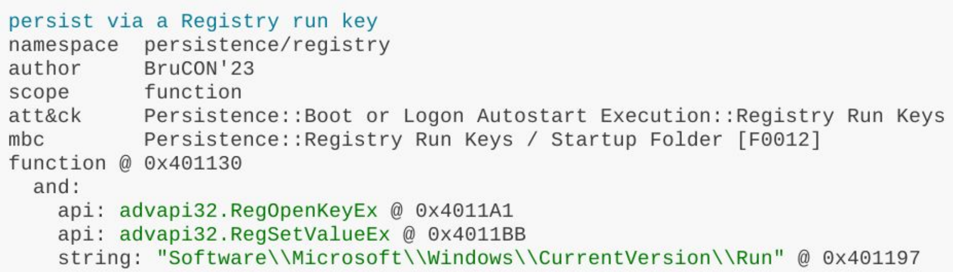
Write a capa rule that matches **persisting via a registry run key** against sample **3f8e2b945deba235fa4888682bd0d640**

1. What features did you reference? Are there any alternatives?
2. Which scope did you use? Why?
3. Can you write a yara rule for this?

lab three answers (a)



```
rule:
  meta:
    name: persist via a Registry run key
    namespace: persistence/registry
    authors:
      - "BruCON'23"
    scope: function
    att&ck:
      - Persistence::Boot or Logon Autostart Execution::Registry Run Keys / Startup Folder [T1547.001]
    mbc:
      - Persistence::Registry Run Keys / Startup Folder [F0012]
  features:
    - and:
      - api: advapi32.RegOpenKeyEx
      - api: advapi32.RegSetValueEx
      - string: "Software\\Microsoft\\Windows\\CurrentVersion\\Run"
```



```
persist via a Registry run key
namespace  persistence/registry
author     BruCON'23
scope      function
att&ck     Persistence::Boot or Logon Autostart Execution::Registry Run Keys
mbc        Persistence::Registry Run Keys / Startup Folder [F0012]
function @ 0x401130
  and:
    api: advapi32.RegOpenKeyEx @ 0x4011A1
    api: advapi32.RegSetValueEx @ 0x4011BB
    string: "Software\\Microsoft\\Windows\\CurrentVersion\\Run" @ 0x401197
```

lab three answers (a)

Write a capa rule that matches **persisting via a registry run key** against sample **3f8e2b945deba235fa4888682bd0d640**

1. What features did you reference? Are there any alternatives?

   see:

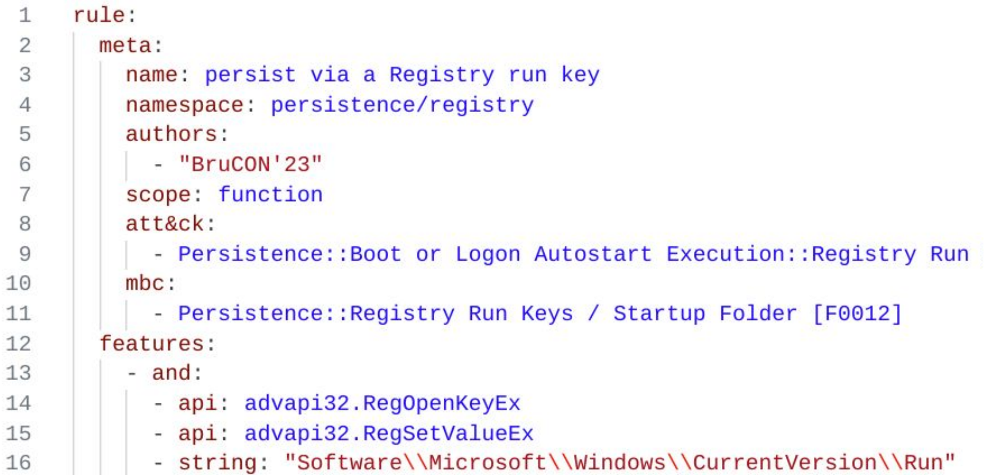
   - persistence/registry/run/persist-via-registry-run-key.yml
   - host-interaction/registry/create/set-registry-value.yml

2. Which scope did you use? Why?

   function

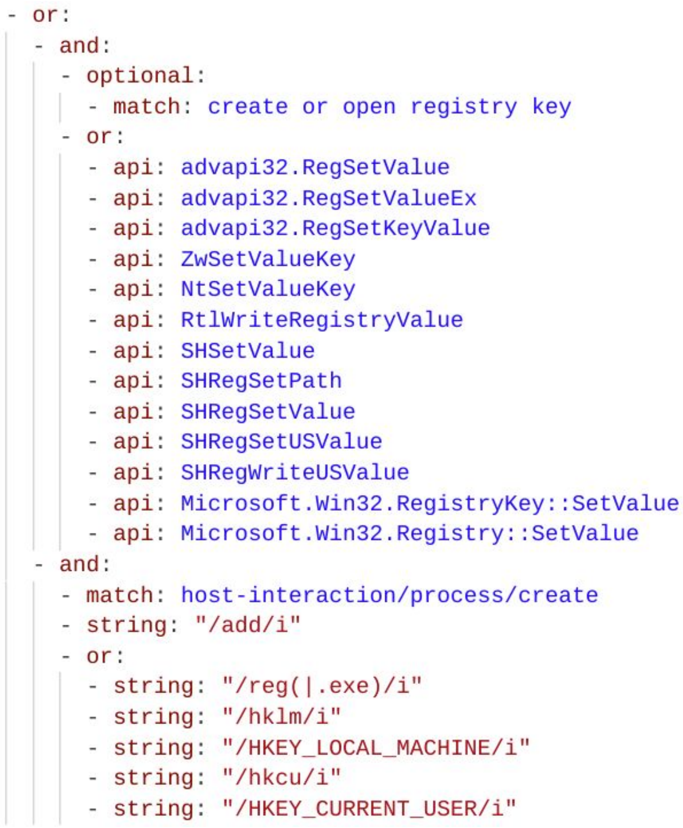
3. Can you write a yara rule for this?

   yes



```
rule:
  meta:
    name: persist via a Registry run key
    namespace: persistence/registry
    authors:
      - "BruCON'23"
    scope: function
    att&ck:
      - Persistence::Boot or Logon Autostart Execution::Registry Run
    mbc:
      - Persistence::Registry Run Keys / Startup Folder [F0012]
  features:
    - and:
      - api: advapi32.RegOpenKeyEx
      - api: advapi32.RegSetValueEx
      - string: "Software\\Microsoft\\Windows\\CurrentVersion\\Run"
```

lab three answers (a): set registry value



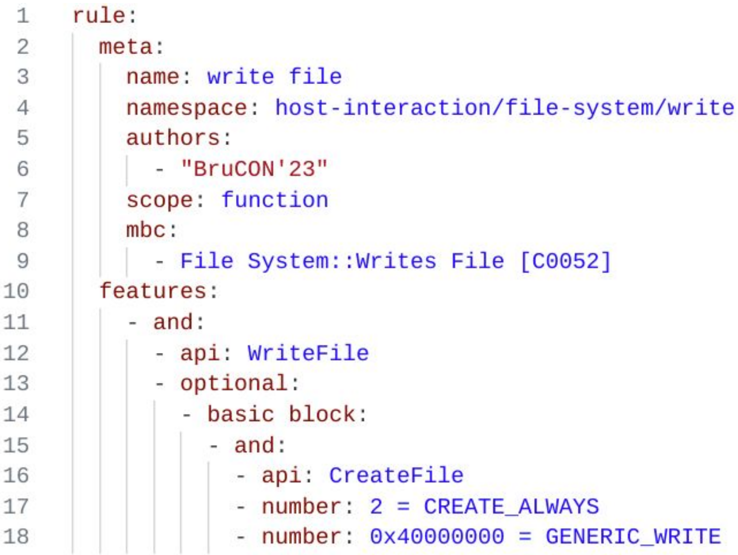
```
- or:
  - and:
    - optional:
      - match: create or open registry key
    - or:
      - api: advapi32.RegSetValue
      - api: advapi32.RegSetValueEx
      - api: advapi32.RegSetKeyValue
      - api: ZwSetValueKey
      - api: NtSetValueKey
      - api: RtlWriteRegistryValue
      - api: SHSetValue
      - api: SHRegSetPath
      - api: SHRegSetValue
      - api: SHRegSetUSValue
      - api: SHRegWriteUSValue
      - api: Microsoft.Win32.RegistryKey::SetValue
      - api: Microsoft.Win32.Registry::SetValue
  - and:
    - match: host-interaction/process/create
    - string: "/add/i"
    - or:
      - string: "/reg(|.exe)/i"
      - string: "/hklm/i"
      - string: "/HKEY_LOCAL_MACHINE/i"
      - string: "/hkcu/i"
      - string: "/HKEY_CURRENT_USER/i"
```

lab three answers (b)

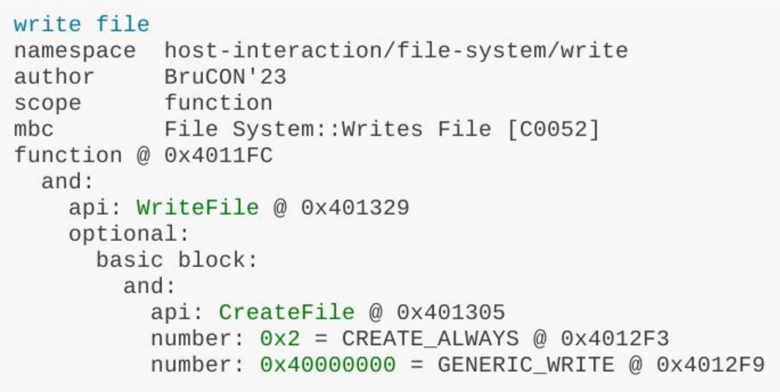
Write a capa rule that matches writing to a file against sample **625ac05fd47adc3c63700c3b30de79ab**

1. What features did you reference? Are there any alternatives?
2. Which scope did you use? Why?
3. Can you write a yara rule for this?

lab three answers (b)



```
rule:
  meta:
    name: write file
    namespace: host-interaction/file-system/write
    authors:
      - "BruCON'23"
    scope: function
    mbc:
      - File System::Writes File [C0052]
  features:
    - and:
      - api: WriteFile
      - optional:
        - basic block:
          - and:
            - api: CreateFile
            - number: 2 = CREATE_ALWAYS
            - number: 0x40000000 = GENERIC_WRITE
```



```
write file
namespace  host-interaction/file-system/write
author     BruCON'23
scope      function
mbc        File System::Writes File [C0052]
function @ 0x4011FC
  and:
    api: WriteFile @ 0x401329
    optional:
      basic block:
        and:
          api: CreateFile @ 0x401305
          number: 0x2 = CREATE_ALWAYS @ 0x4012F3
          number: 0x40000000 = GENERIC_WRITE @ 0x4012F9
```

lab three answers (b)

Write a capa rule that matches **writing to a file** against sample **3f8e2b945deba235fa4888682bd0d640**

1. What features did you reference? Are there any alternatives?

   WriteFile

   optional: CreateFile with arguments

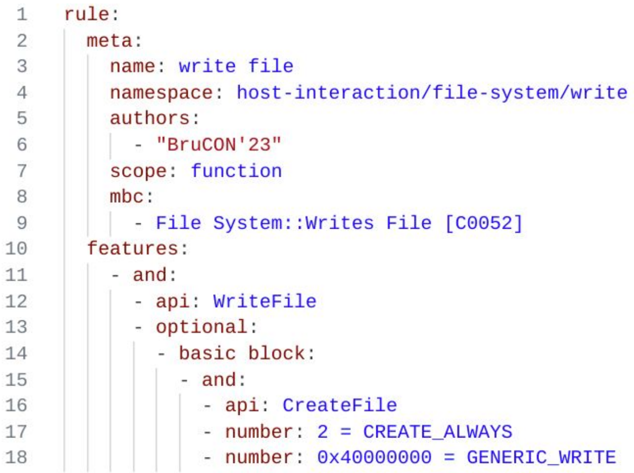
2. Which scope did you use? Why?

   function, so that the basic block subscope can work.

   Otherwise, instruction scope.

3. Can you write a yara rule for this?

   yes? maybe?



```yaml
rule:
  meta:
    name: write file
    namespace: host-interaction/file-system/write
    authors:
      - "BruCON'23"
    scope: function
    mbc:
      - File System::Writes File [C0052]
  features:
    - and:
      - api: WriteFile
      - optional:
        - basic block:
          - and:
            - api: CreateFile
            - number: 2 = CREATE_ALWAYS
            - number: 0x40000000 = GENERIC_WRITE
```

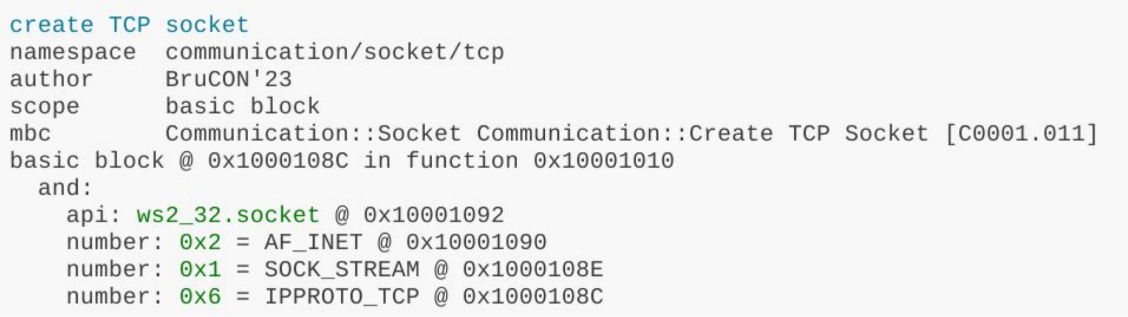
lab three answers (c)

Write a capa rule that matches **creating a TCP socket** against sample **`290934c61de9176ad682ffdd65f0a669`**

1. What features did you reference? Are there any alternatives?
2. Which scope did you use? Why?
3. Can you write a yara rule for this?

lab three answers (c)

```
rule:
  meta:
    name: create TCP socket
    namespace: communication/socket/tcp
    authors:
      - "BruCON'23"
    scope: basic block
    mbc:
      - Communication::Socket Communication::Create TCP Socket [C0001.011]
  features:
    - and:
      - api: ws2_32.socket
      - number: 2 = AF_INET
      - number: 1 = SOCK_STREAM
      - number: 6 = IPPROTO_TCP
```



```
create TCP socket
namespace  communication/socket/tcp
author     BruCON'23
scope      basic block
mbc        Communication::Socket Communication::Create TCP Socket [C0001.011]
basic block @ 0x1000108C in function 0x10001010
  and:
    api: ws2_32.socket @ 0x10001092
    number: 0x2 = AF_INET @ 0x10001090
    number: 0x1 = SOCK_STREAM @ 0x1000108E
    number: 0x6 = IPPROTO_TCP @ 0x1000108C
```

lab three answers (c)

Write a capa rule that matches **creating a TCP socket** against sample **`290934c61de9176ad682ffdd65f0a669`**

1. What features did you reference? Are there any alternatives?

   ws2_32.socket, but also:

   - WSASocket
   - socket

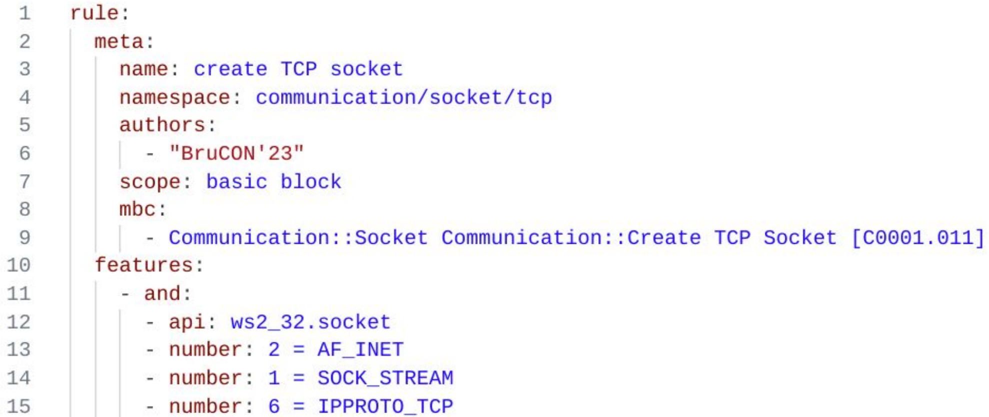
2. Which scope did you use? Why?

   basic block, to capture the arguments to the API call.

3. Can you write a yara rule for this?

   No, due to operand decoding.

   (Also, did you notice: socket is imported by ordinal?)



```
rule:
  meta:
    name: create TCP socket
    namespace: communication/socket/tcp
    authors:
      - "BruCON'23"
    scope: basic block
    mbc:
      - Communication::Socket Communication::Create TCP Socket [C0001.011]
  features:
    - and:
      - api: ws2_32.socket
      - number: 2 = AF_INET
      - number: 1 = SOCK_STREAM
      - number: 6 = IPPROTO_TCP
```

shortcomings

capa limitations

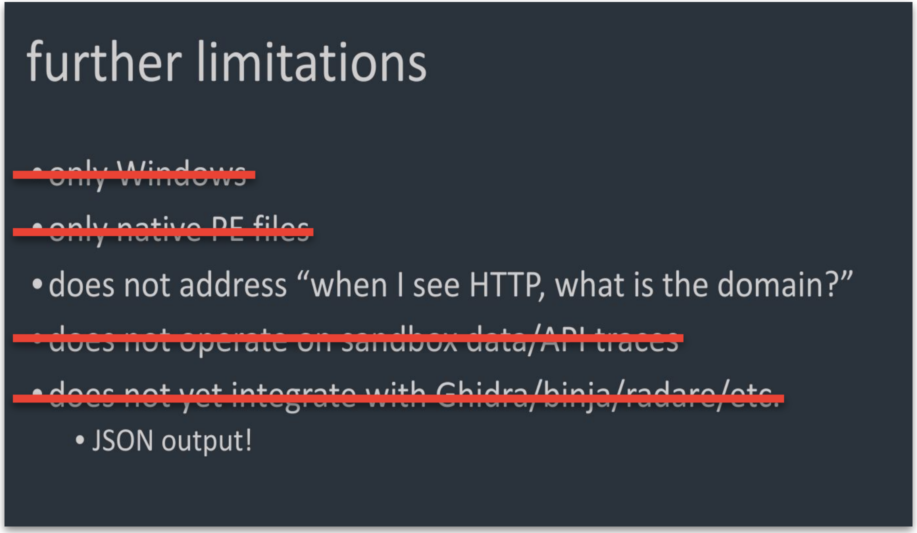
Obfuscation

- Hides logic preventing capa from working well

No call scope

- Workaround: group features using basic block scope

Expertise to author rules



further limitations

- ~~only Windows~~
- ~~only native PE files~~
- does not address "when I see HTTP, what is the domain?"
- ~~does not operate on sandbox data/API traces~~
- ~~does not yet integrate with Ghidra/binja/radare/etc.~~
  - JSON output!

06

Conclusion

ongoing and future work

Ghidra UI

In progress, but help wanted

Dynamic Analysis

In progress

Call Scope

Help wanted

ARM Architecture

ARM enthusiasts wanted

Website

In progress, help wanted

Thank you.