black hat ASIA 2021

May 6-7, 2021

BRIEFINGS

Mem2Img : Memory-Resident Malware Detection via Convolution Neural Network

Aragorn Tseng

Charles Li

TEAMT5
杜浦數位安全

Aragorn Tseng

Malware Researcher

Charles Li

Chief Analyst

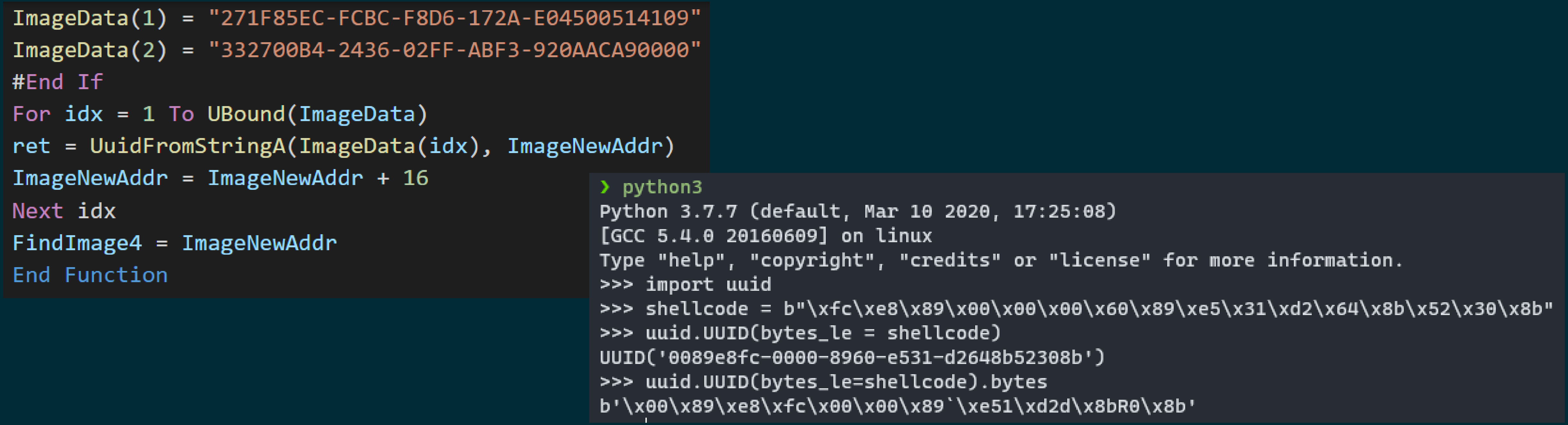
AGENDA

- Recent Injection Technique used by APT
- Dataset overview
- Mem2Img Framework
- Experiment result
- Saliency map
- Zero shot learning
- Adversarial Attack

Recent Injection Technique used by APT

UUID Shellcode

- UUidFromStrinA - it takes a string-based UUID and converts it to it's binary representation. It takes a pointer to a UUID, which will be used to return the converted binary data.



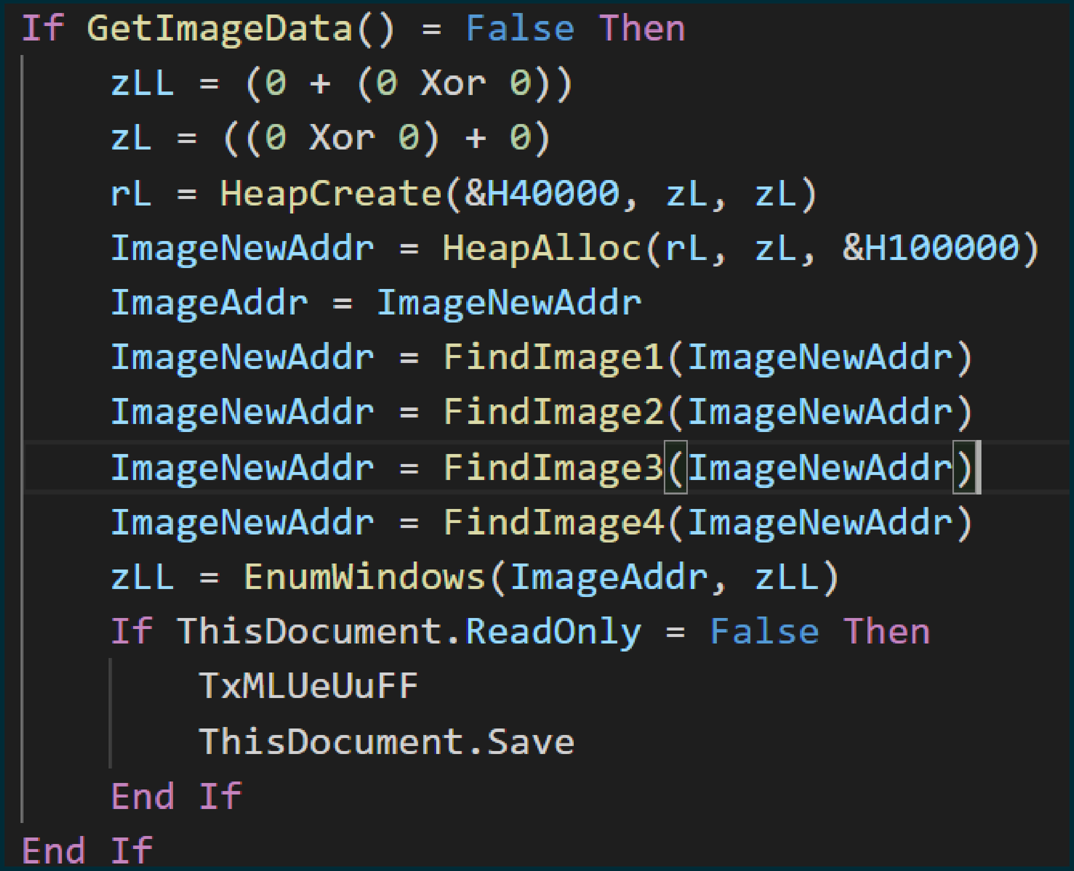
```
ImageData(1) = "271F85EC-FCBC-F8D6-172A-E04500514109"
ImageData(2) = "332700B4-2436-02FF-ABF3-920AACA90000"
#End If
For idx = 1 To UBound(ImageData)
ret = UuidFromStringA(ImageData(idx), ImageNewAddr)
ImageNewAddr = ImageNewAddr + 16
Next idx
FindImage4 = ImageNewAddr
End Function
```

```
❯ python3
Python 3.7.7 (default, Mar 10 2020, 17:25:08)
[GCC 5.4.0 20160609] on linux
Type "help", "copyright", "credits" or "license" for more information.
>>> import uuid
>>> shellcode = b"\xfc\xe8\x89\x00\x00\x00\x60\x89\xe5\x31\xd2\x64\x8b\x52\x30\x8b"
>>> uuid.UUID(bytes_le = shellcode)
UUID('0089e8fc-0000-8960-e531-d2648b52308b')
>>> uuid.UUID(bytes_le=shellcode).bytes
b'\x00\x89\xe8\xfc\x00\x00\x89`\xe51\xd2d\x8bR0\x8b'
```

Lazarus shellcode by nccgroup

UUID Shellcode

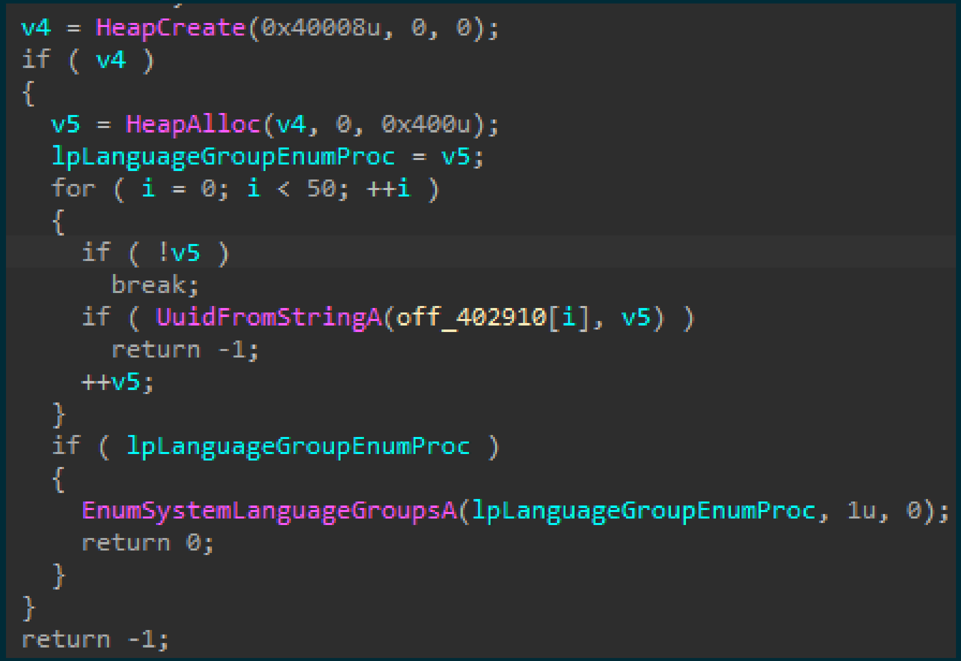
- By providing a pointer to an heap address, this function can be (ab)used to both decode data and write it to memory without using common functions such as memcpy or WriteProcessMemory.
- Then use callback function(EnumWindows) to execute shellcode
- This vba script was used by Lazarus



```
If GetImageData() = False Then
    zLL = (0 + (0 Xor 0))
    zL = ((0 Xor 0) + 0)
    rL = HeapCreate(&H40000, zL, zL)
    ImageNewAddr = HeapAlloc(rL, zL, &H100000)
    ImageAddr = ImageNewAddr
    ImageNewAddr = FindImage1(ImageNewAddr)
    ImageNewAddr = FindImage2(ImageNewAddr)
    ImageNewAddr = FindImage3(ImageNewAddr)
    ImageNewAddr = FindImage4(ImageNewAddr)
    zLL = EnumWindows(ImageAddr, zLL)
    If ThisDocument.ReadOnly = False Then
        TxMLUeUuFF
        ThisDocument.Save
    End If
End If
```

Callback function to execute shellcode

- the lpLocaleEnumProc parameter specifies a callback function! By providing the address returned by HeapAlloc, this function can be (ab)used to execute shellcode
- There are many callback functions can used to execute shellcode
- This case was used in a PE file

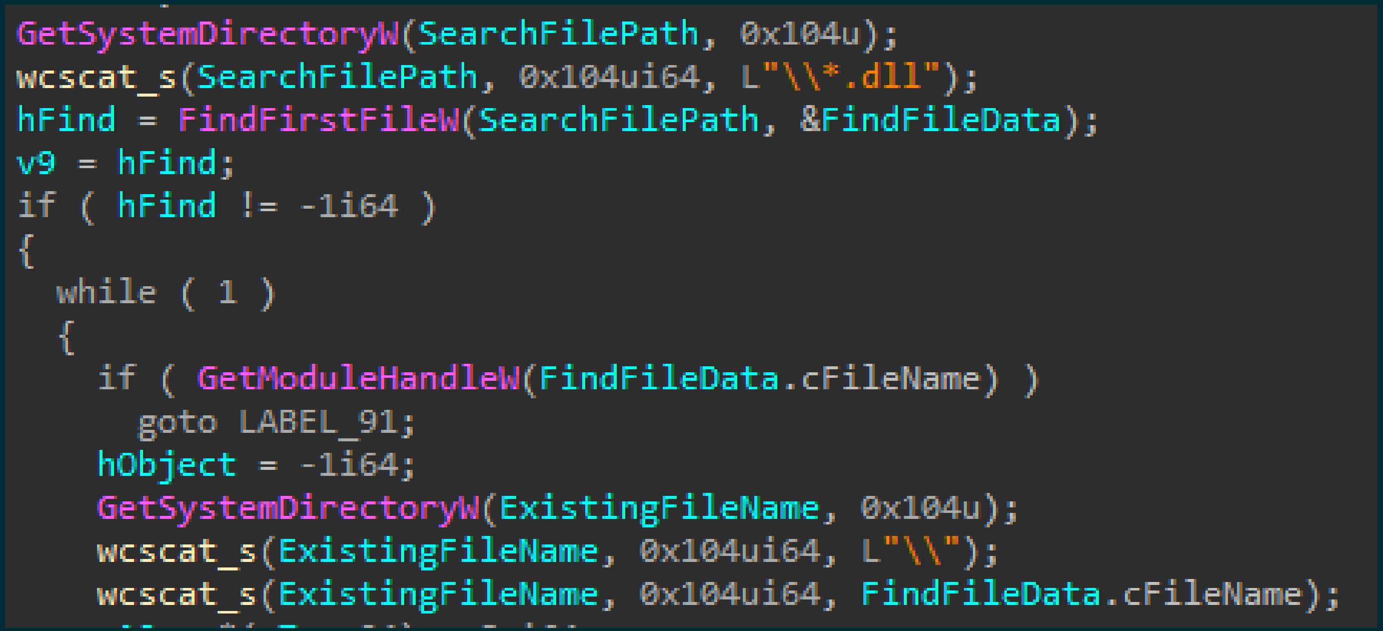


```
v4 = HeapCreate(0x40008u, 0, 0);
if ( v4 )
{
  v5 = HeapAlloc(v4, 0, 0x400u);
  lpLanguageGroupEnumProc = v5;
  for ( i = 0; i < 50; ++i )
  {
    if ( !v5 )
      break;
    if ( UuidFromStringA(off_402910[i], v5) )
      return -1;
    ++v5;
  }
  if ( lpLanguageGroupEnumProc )
  {
    EnumSystemLanguageGroupsA(lpLanguageGroupEnumProc, 1u, 0);
    return 0;
  }
}
return -1;
```

https://github.com/S4R1N/AlternativeShellcodeExec

Phantom DLL Hollowing

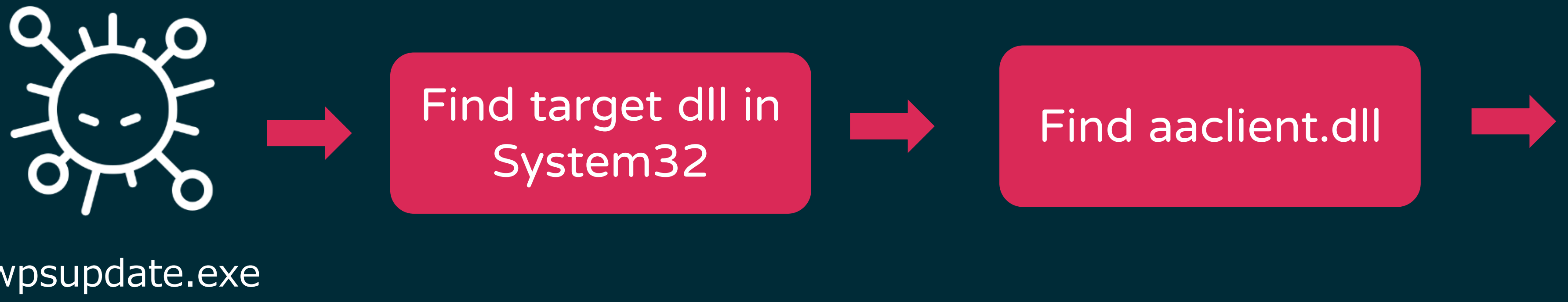
- The target dll is chosen based on the size of its .text section to house the reflective payload and then it could execute the binary within a + RX section in that dll
- We have found APT27 used this technique to spread CobaltStrike Beacon



```
GetSystemDirectoryW(SearchFilePath, 0x104u);
wcscat_s(SearchFilePath, 0x104ui64, L"\\*.dll");
hFind = FindFirstFileW(SearchFilePath, &FindFileData);
v9 = hFind;
if ( hFind != -1i64 )
{
  while ( 1 )
  {
    if ( GetModuleHandleW(FindFileData.cFileName) )
      goto LABEL_91;
    hObject = -1i64;
    GetSystemDirectoryW(ExistingFileName, 0x104u);
    wcscat_s(ExistingFileName, 0x104ui64, L"\\");
    wcscat_s(ExistingFileName, 0x104ui64, FindFileData.cFileName);
```

https://github.com/fancysauced/phantom-dll-hollower-poc

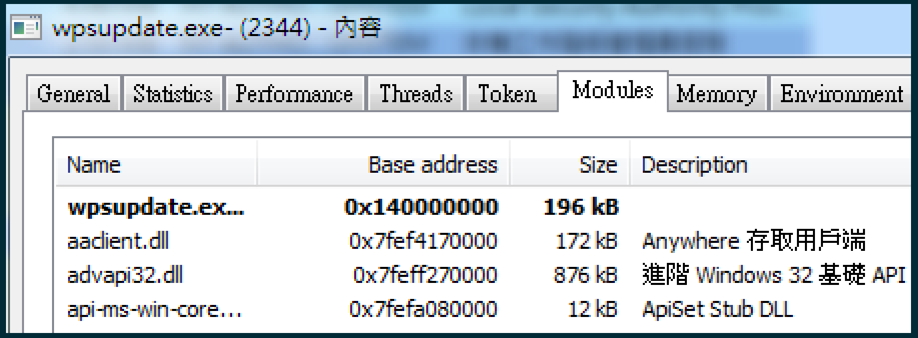
Phantom DLL Hollowing



wpsupdate.exe

Find target dll in System32

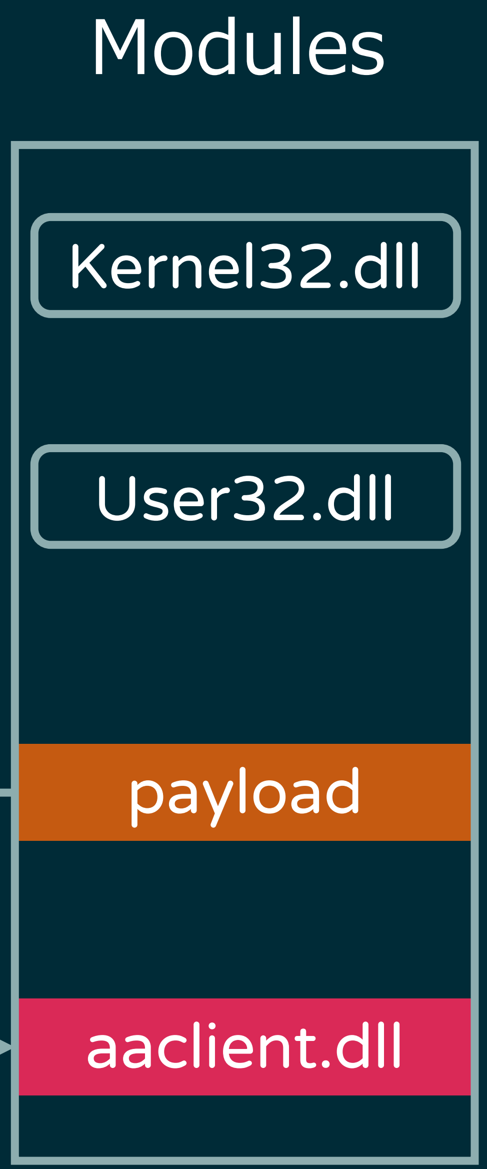
Find aaclient.dll



wpsupdate.exe- (2344) - 內容

Name	Base address	Size	Description
wpsupdate.ex...	**0x140000000**	**196 kB**	
aaclient.dll	0x7fef4170000	172 kB	Anywhere 存取用戶端
advapi32.dll	0x7feff270000	876 kB	進階 Windows 32 基礎 API
api-ms-win-core...	0x7fefa080000	12 kB	ApiSet Stub DLL

Phamtom
Dll hollowing



Modules

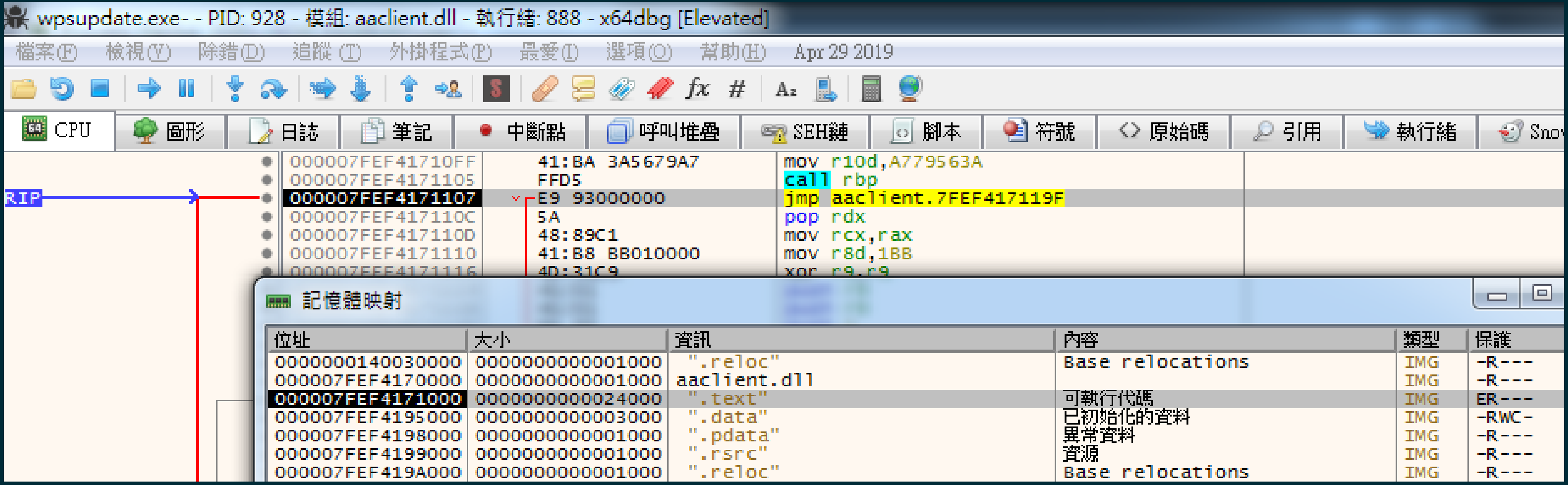
Kernel32.dll

User32.dll

payload

aaclient.dll

Phantom DLL Hollowing



wpsupdate.exe- - PID: 928 - 模組: aaclient.dll - 執行緒: 888 - x64dbg [Elevated]
檔案(F) 檢視(V) 除錯(D) 追蹤(T) 外掛程式(P) 最愛(I) 選項(O) 幫助(H) Apr 29 2019
CPU 圖形 日誌 筆記 中斷點 呼叫堆疊 SEH鏈 腳本 符號 原始碼 引用 執行緒 Sno
RIP
記憶體映射

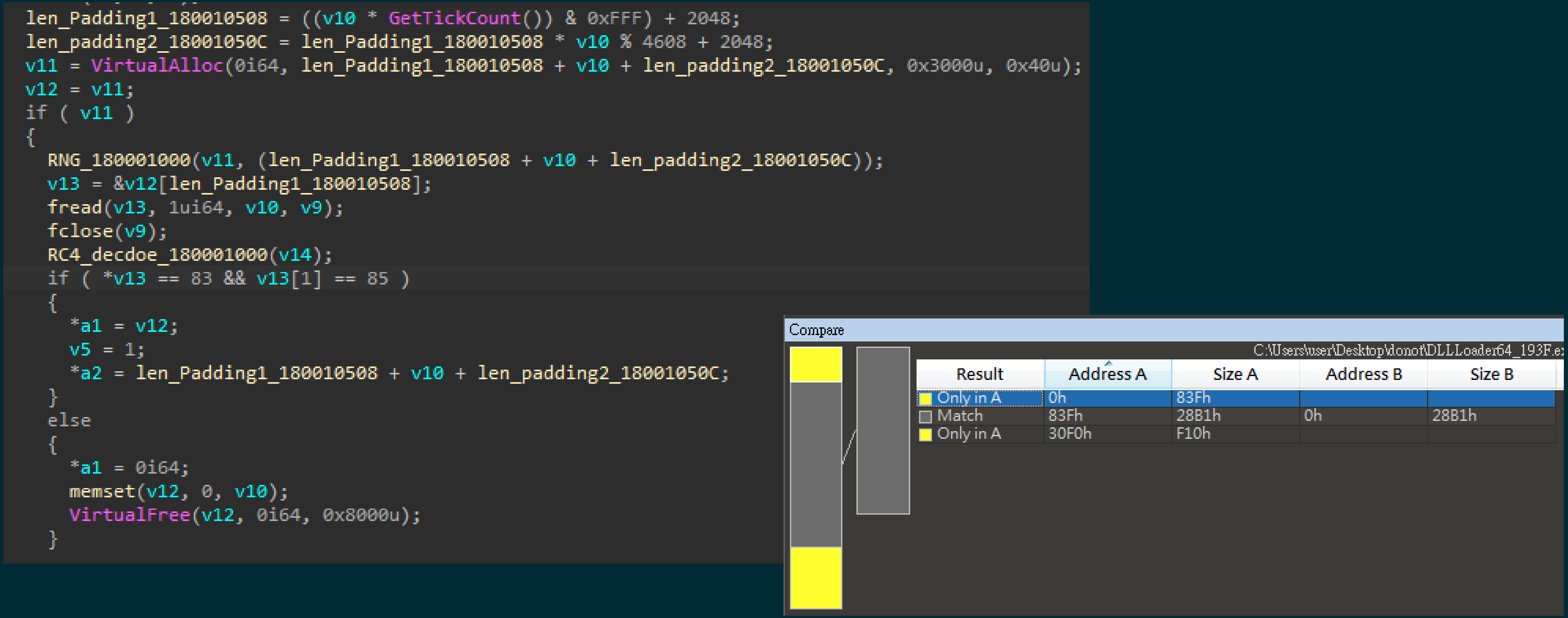
```
000007FEF41710FF    41:BA 3A5679A7    mov r10d,A779563A
000007FEF4171105    FFD5              call rbp
000007FEF4171107    E9 93000000       jmp aaclient.7FEF417119F
000007FEF417110C    5A                pop rdx
000007FEF417110D    48:89C1           mov rcx,rax
000007FEF4171110    41:B8 BB010000    mov r8d,1BB
000007FEF4171116    4D:31C9           xor r9,r9
```

位址	大小	資訊	內容	類型	保護
0000000140030000	0000000000001000	".reloc"	Base relocations	IMG	-R---
000007FEF4170000	0000000000001000	aaclient.dll		IMG	-R---
000007FEF4171000	0000000000024000	".text"	可執行代碼	IMG	ER---
000007FEF4195000	0000000000003000	".data"	已初始化的資料	IMG	-RWC-
000007FEF4198000	0000000000001000	".pdata"	異常資料	IMG	-R---
000007FEF4199000	0000000000001000	".rsrc"	資源	IMG	-R---
000007FEF419A000	0000000000001000	".reloc"	Base relocations	IMG	-R---

In this case, the DLL used to make the phantom dll hollowing is aaclient.dll, it execute the cobaltstrike stager shellcode within a + RX section in that dll

Shellcode injection - Waterbear

- Generate random junk bytes to envelop real shellcode when decoding



```
len_Padding1_180010508 = ((v10 * GetTickCount()) & 0xFFF) + 2048;
len_padding2_18001050C = len_Padding1_180010508 * v10 % 4608 + 2048;
v11 = VirtualAlloc(0i64, len_Padding1_180010508 + v10 + len_padding2_18001050C, 0x3000u, 0x40u);
v12 = v11;
if ( v11 )
{
  RNG_180001000(v11, (len_Padding1_180010508 + v10 + len_padding2_18001050C));
  v13 = &v12[len_Padding1_180010508];
  fread(v13, 1ui64, v10, v9);
  fclose(v9);
  RC4_decdoe_180001000(v14);
  if ( *v13 == 83 && v13[1] == 85 )
  {
    *a1 = v12;
    v5 = 1;
    *a2 = len_Padding1_180010508 + v10 + len_padding2_18001050C;
  }
  else
  {
    *a1 = 0i64;
    memset(v12, 0, v10);
    VirtualFree(v12, 0i64, 0x8000u);
  }
```

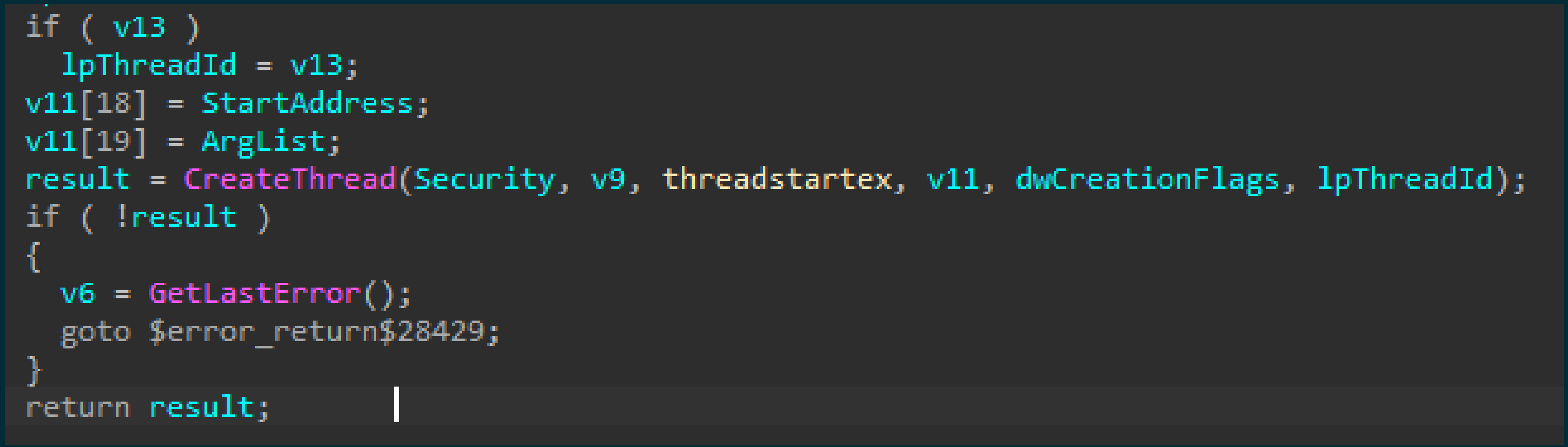
Compare

C:\Users\user\Desktop\donot\DLLLoader64_193F.e

Result	Address A	Size A	Address B	Size B
🟨 Only in A	0h	83Fh		
□ Match	83Fh	28B1h	0h	28B1h
🟨 Only in A	30F0h	F10h		

Shellcode injection - Waterbear

- Using beginthreadex() acts as a proxy and starts the new thread at threadstartex(), instead of using the address where the shellcode is located as if using CreateThread() directly



```
if ( v13 )
  lpThreadId = v13;
v11[18] = StartAddress;
v11[19] = ArgList;
result = CreateThread(Security, v9, threadstartex, v11, dwCreationFlags, lpThreadId);
if ( !result )
{
  v6 = GetLastError();
  goto $error_return$28429;
}
return result;
```

Dataset Overview

Memory Resident malware used by APT

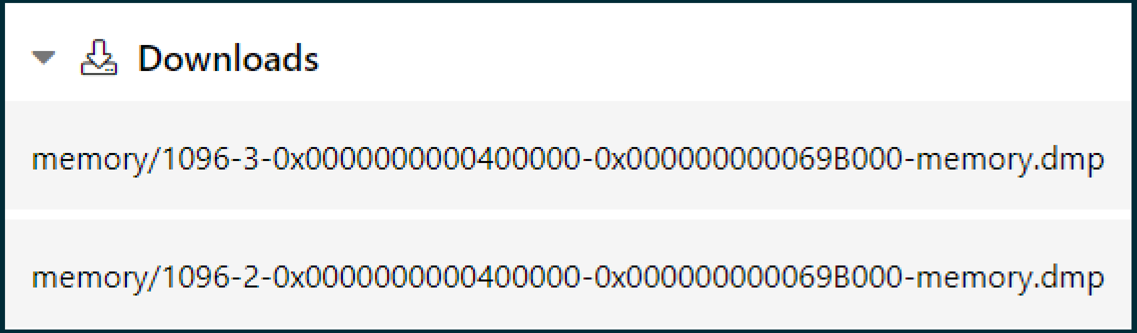
- APT32 (OceanLotus) - Denis backdoor
- APT37 – Rokrat RAT
- Tropic Trooper - TClient backdoor
- BlackTech (PLEAD) – TSCookie, Capgeld, waterbear, kivars
- APT10 – Sodamaster, Lodeinfo, P8RAT, CobaltStrike
- Mustang Panda – PlugX
- PhamtomIvy
- APT27 – Sysupdate, Hyperbro, CobaltStrike
- Winnti - CobaltStrike, ShadowPad
- Darkseoul – Dtrack
- Unknown group – Dropsocks, Dpass
- 21 malware family

Cyber Crime Memory-resident Malware

- Emotet
- Formbook
- Dridex
- AgentTesla
- Trickbot
- QuasarRAT(also used in APT)
- 6 malware family

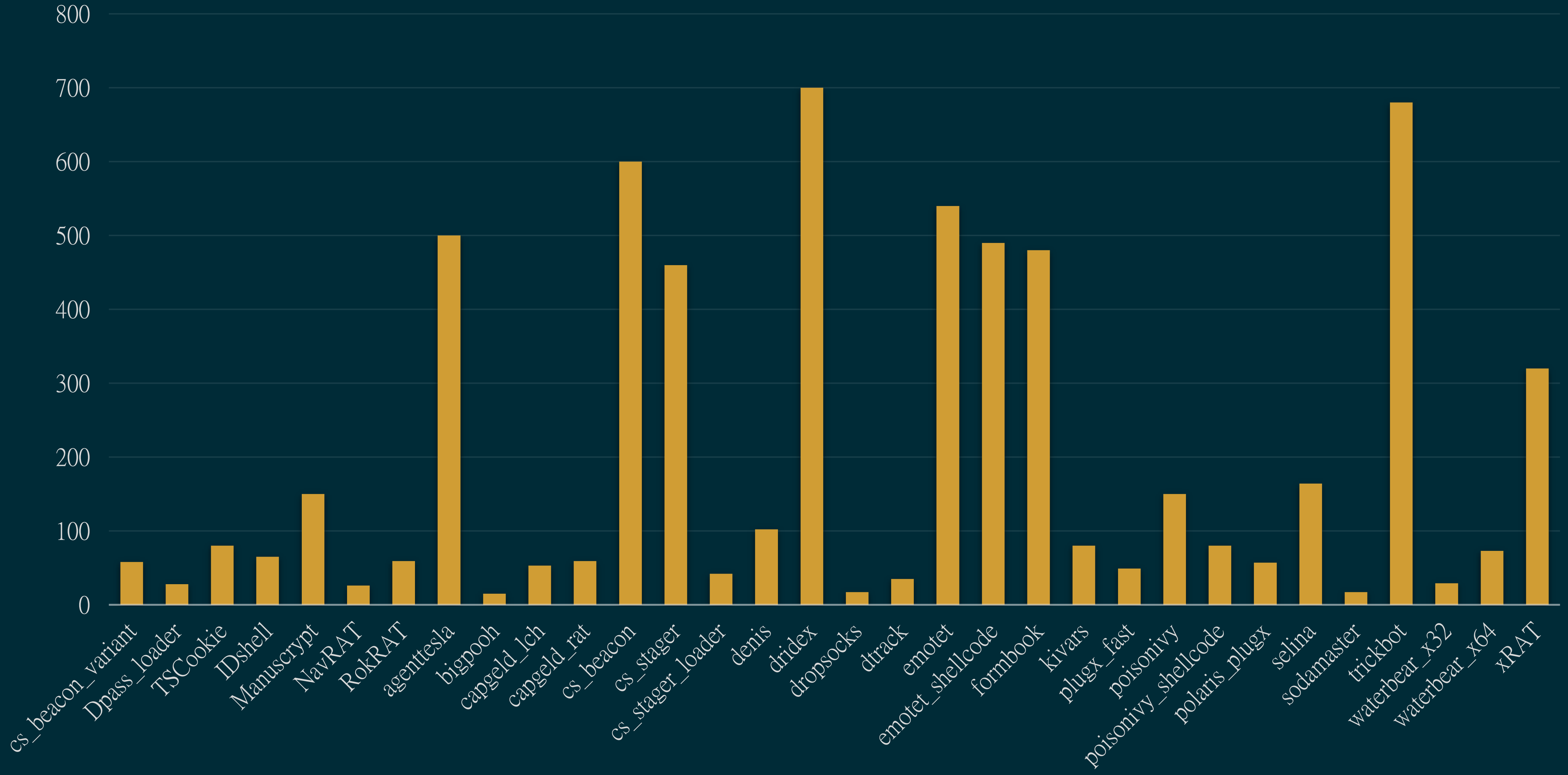
How to find memory-resident malware

- Tool
  - pe-sieve (hollows_hunter)
  - volatility(malfind)
  - Hollowfind
- Data source
  - Victim’s PC
  - Triage
  - VirusTotal



```
memory/1096-3-0x0000000000400000-0x000000000069B000-memory.dmp
memory/1096-2-0x0000000000400000-0x000000000069B000-memory.dmp
```

File distribution

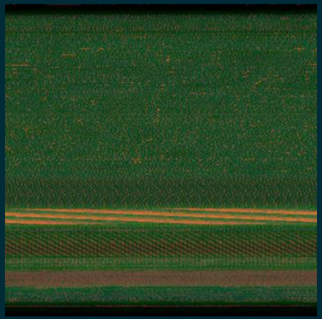


How to deal with Data Imbalance issue

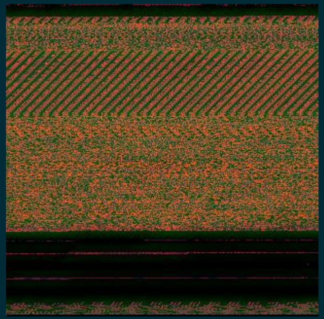
- Use class weights
  - class_1 has 1000 instances and class_2 has 100 instances
  - class_weights={"class_1": 1, "class_2": 10}
- SMOTE
- Data argumentation
  - Rotate, Flip, Scale
- Transfer learning
  - VGG16
  - InceptionV3

Why Transfer Learning

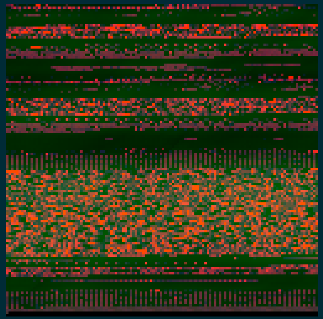
- Some APT Memory-resident malware is a small set of data
- Transfer learning uses knowledge from a learned task to improve the performance on a related task, typically reducing the amount of required training data.
- They allow models to make predictions for a new domain or task (target domain) using knowledge learned from another dataset or existing machine learning models (source domain).



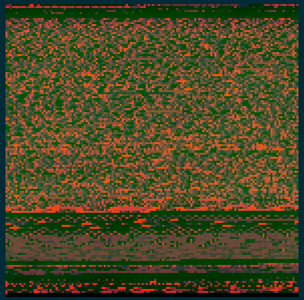
AgentTesla



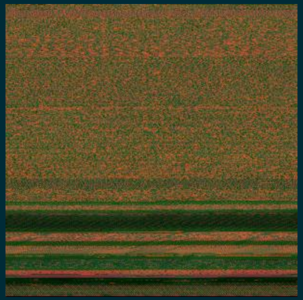
Bigpooh



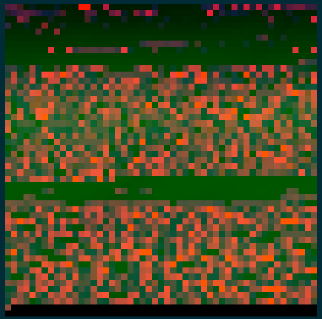
Capgeld_loader



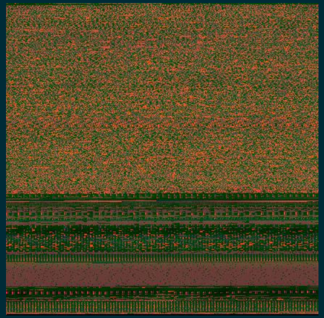
Capgeld_RAT



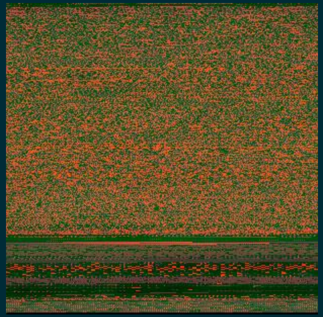
CobaltStrike beacon



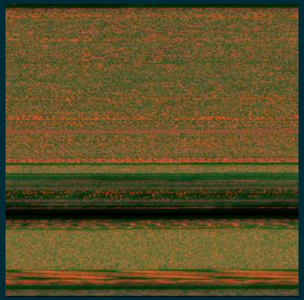
CobaltStrike stager



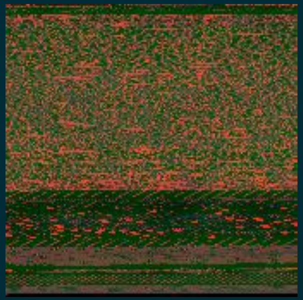
CobaltStrike stager loader



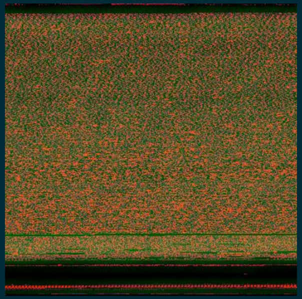
CobaltStrike variant



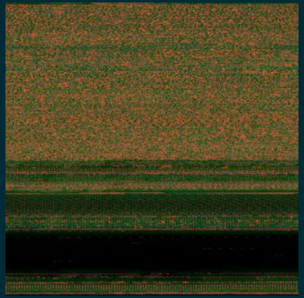
Denis RAT



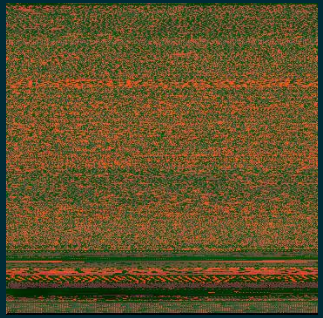
Dpass Loader



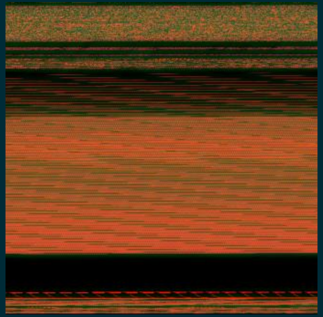
Dridex



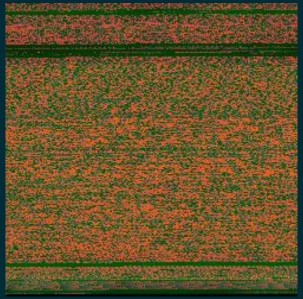
Dropsocks



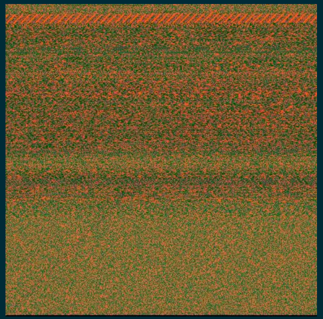
Dtrack



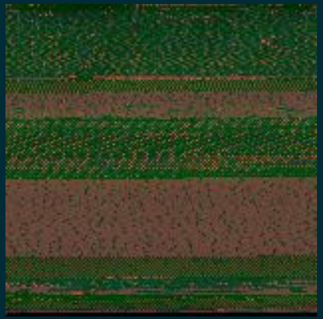
Emotet



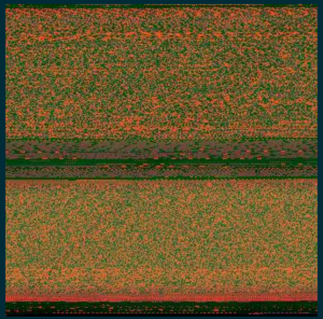
Emotet shellcode



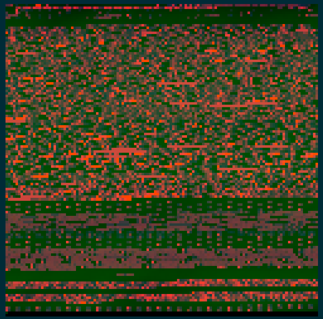
Formbook



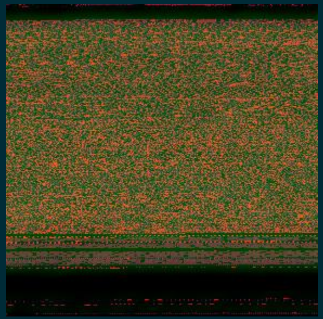
TSCookie



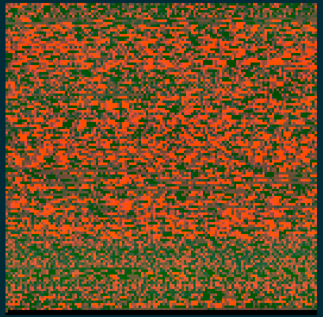
IDShell



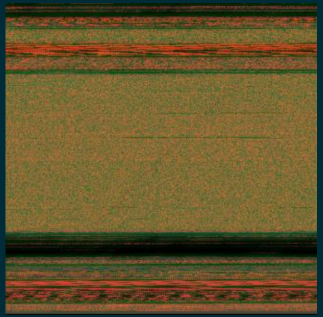
kivars



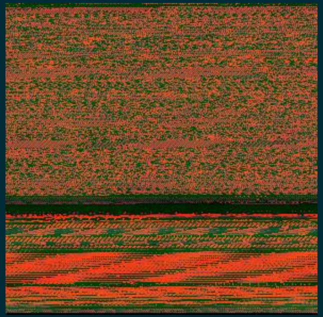
Manuscrypt



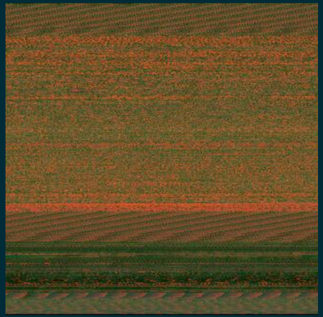
PoisonIvy



PhatomIvy

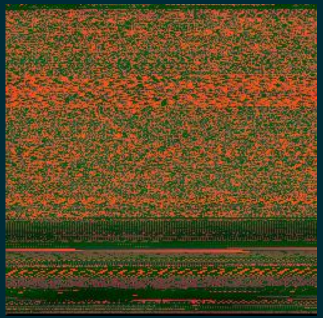


PlugX

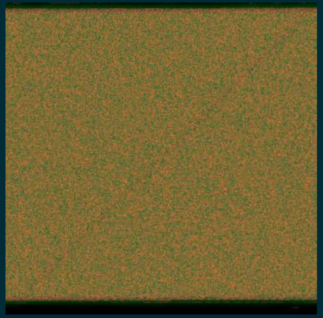


RokRAT

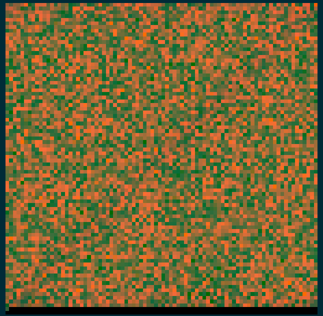
Selina



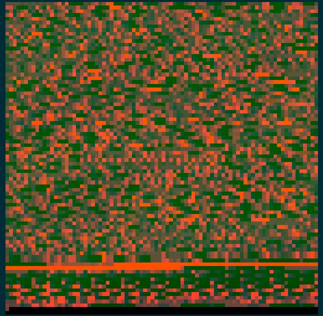
Sodamaster



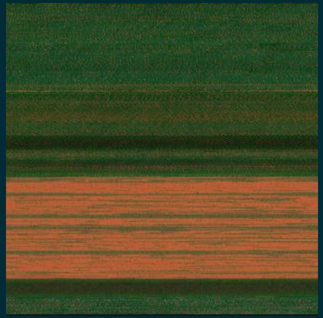
Trickbot



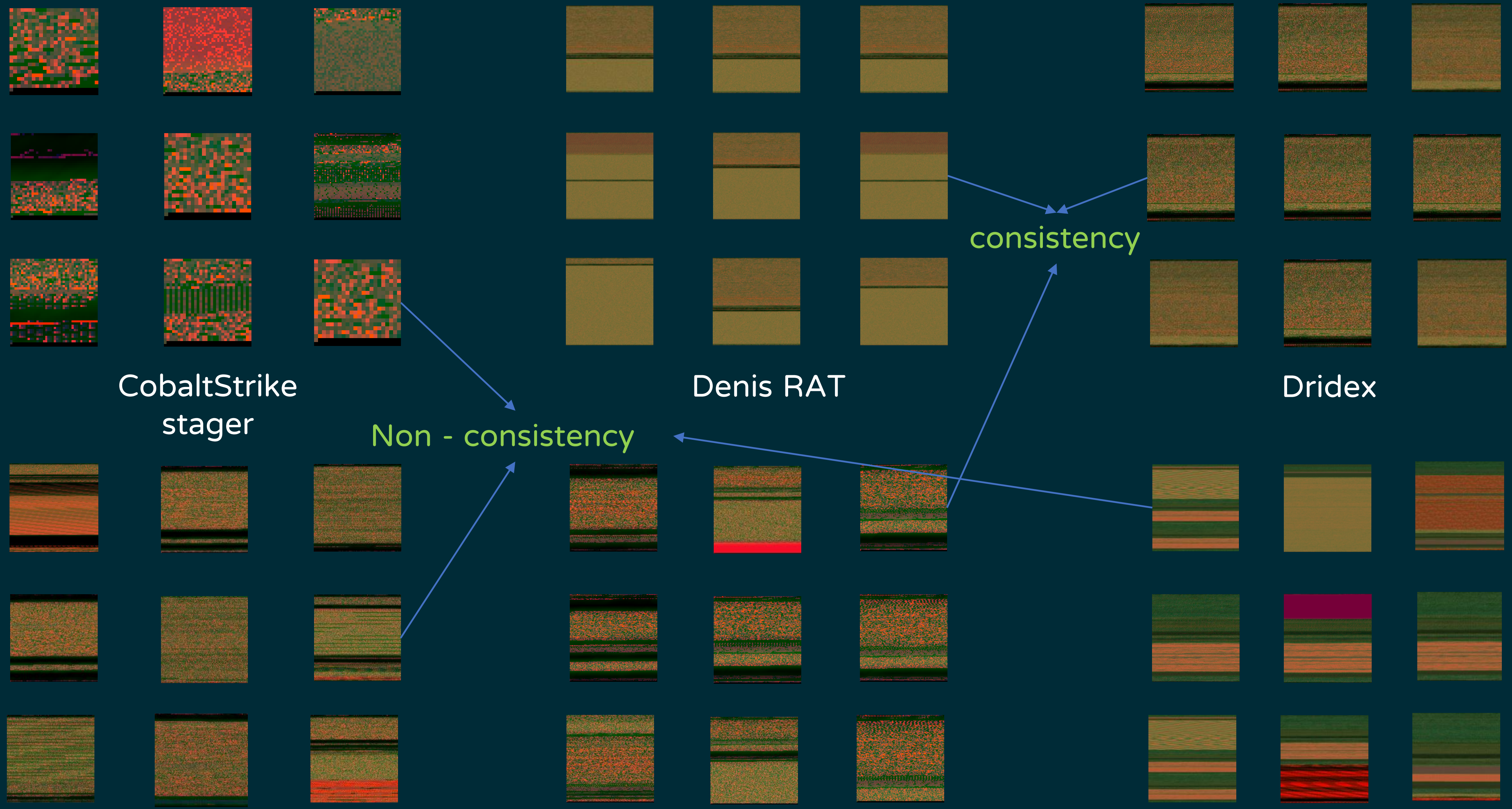
Waterbear_x32



Waterbear_x64



quarsarRAT



CobaltStrike stager

Denis RAT

Dridex

consistency

Non - consistency

Emotet

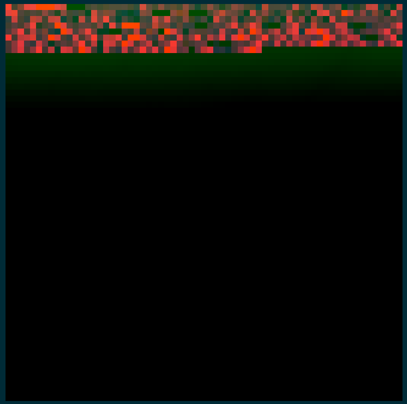
TSCookie

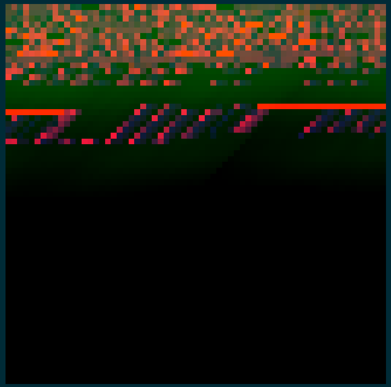
xRAT

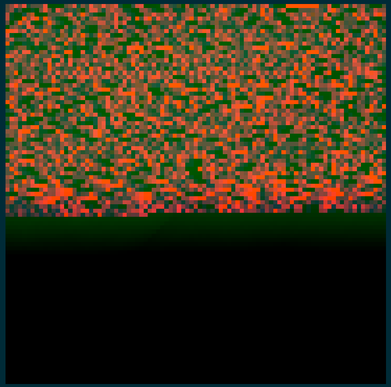
Mem2Img Framework

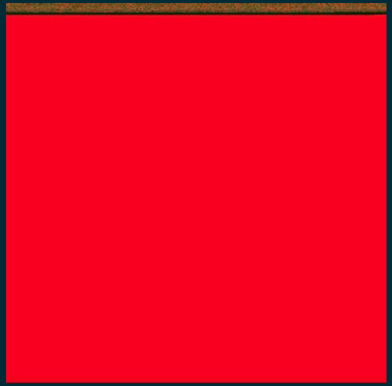
Preprocessing Data

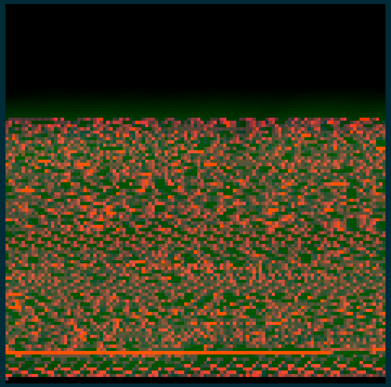
- Remove continuous bytes(junk bytes) in the binary, ex : NULL bytes, 0xFF



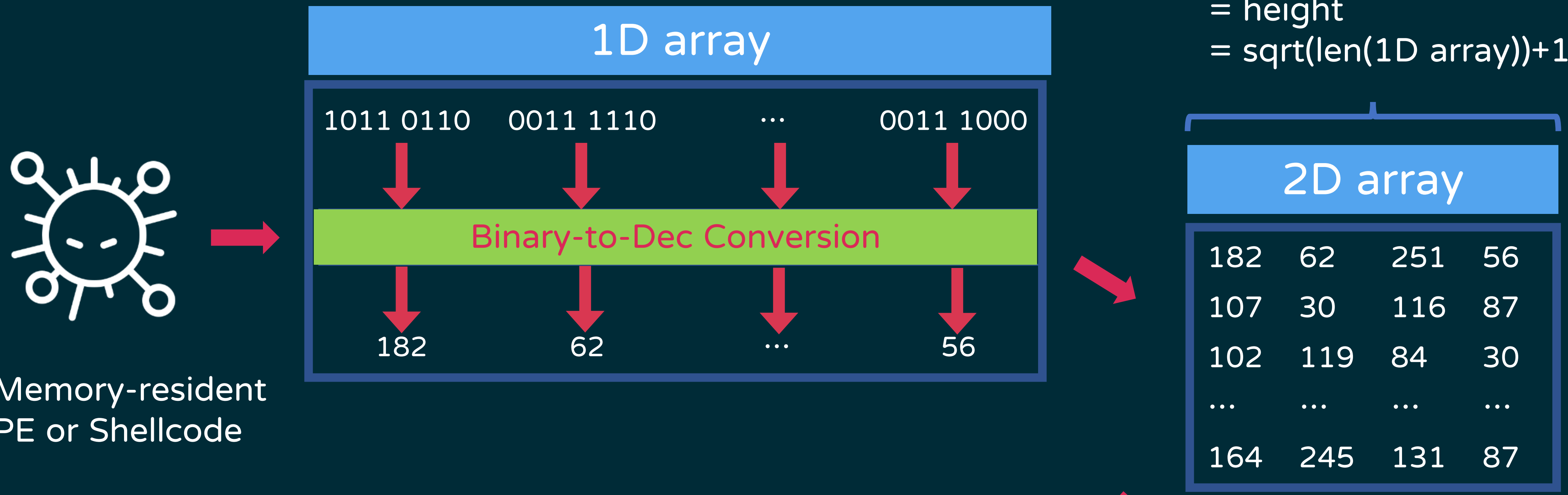








1D Array to 2D Array



Memory-resident PE or Shellcode

1D array

1011 0110 → 182

0011 1110 → 62

… → …

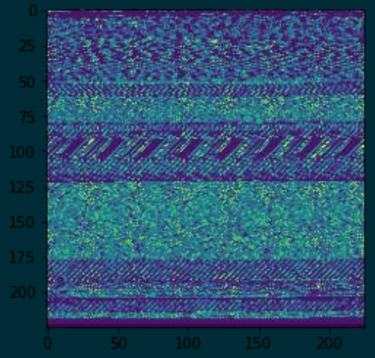
0011 1000 → 56

Binary-to-Dec Conversion

Image width
= height
= sqrt(len(1D array))+1

2D array

182	62	251	56
107	30	116	87
102	119	84	30
…	…	…	…
164	245	131	87



8-bit vectors to Images

Three channel of the image

- Red channel : decimal values of each bytes
- Green Channel : Shannon entropy values of each bytes
- Blue channel : Local entropy values of the image
    - Use entropy function of skimage library
    - Local entropy is computed using base 2 logarithm and related to the complexity contained in a given neighborhood
    - the filter returns the minimum number of bits needed to encode the local gray level distribution. The disk is set to 10 in Mem2Img framework

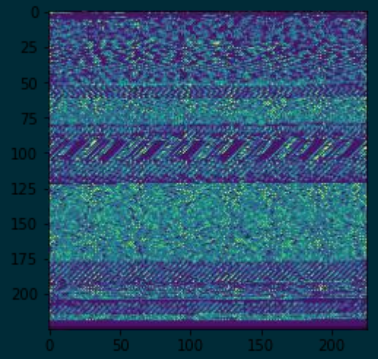
Memory Resident Malware

0011 1110	1011 0110	1111 1011	0011 1000
0101 0111	0111 0111	0111 0100	0110 1011
0110 0110	0001 1110	0101 0100	0001 1110
0010 0100	1001 1111	0101 0011	0101 0111
0000 1110	0000 1100	1100 1100	1111 0100

62	182	251	56
87	119	116	107
102	30	84	30
36	159	86	206
164	245	131	87

Decimal – Red Channel

with decimal values of each byte



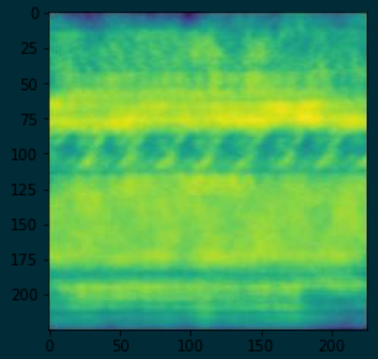
Convert to grayscale image

Count Shannon entropy bytes to bytes, ie:10110111 -> 0.9544

0.9544	0.9544	0.5436	0.9544
0.8544	0.8113	1	0.9544
1	1	0.9544	1
0.9544	0.8113	1	0.9544
0.9544	0.8113	1	0.9544

Shannon Entropy – Green Channel

with Shannon entropy values of each byte Value*15



Generate local entropy image

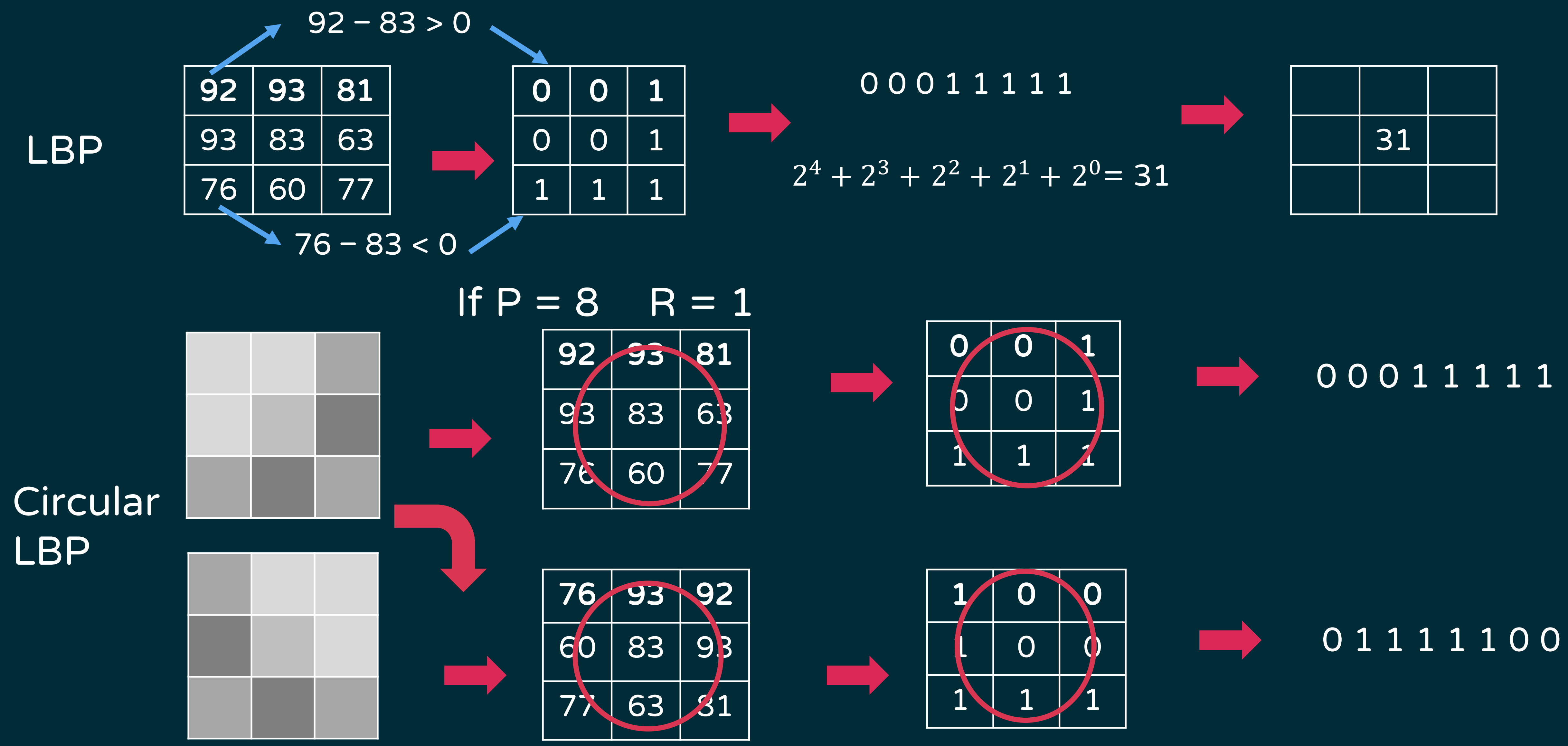
Put the value of entropy image to blue channel

3.1521	3.0935	3.0424	3.0606
3.0398	3.0642	3.0241	2.9824
2.8085	2.7159	2.7506	2.6820
2.5863	2.5259	2.4454	2.2180
2.4309	1.9847	1.8668	1.8170

Local Entropy – Blue Channel

with local entropy values of each byte Value*60

Local Binary Pattern(LBP)



LBP

92 − 83 > 0

92	**93**	**81**
93	83	63
76	60	77

76 − 83 < 0

0	**0**	**1**
0	0	1
1	1	1

0 0 0 1 1 1 1 1

$2^4 + 2^3 + 2^2 + 2^1 + 2^0 = 31$

	31	

If P = 8 R = 1

Circular LBP

92	**93**	**81**
93	83	63
76	60	77

0	0	1
0	0	1
1	1	1

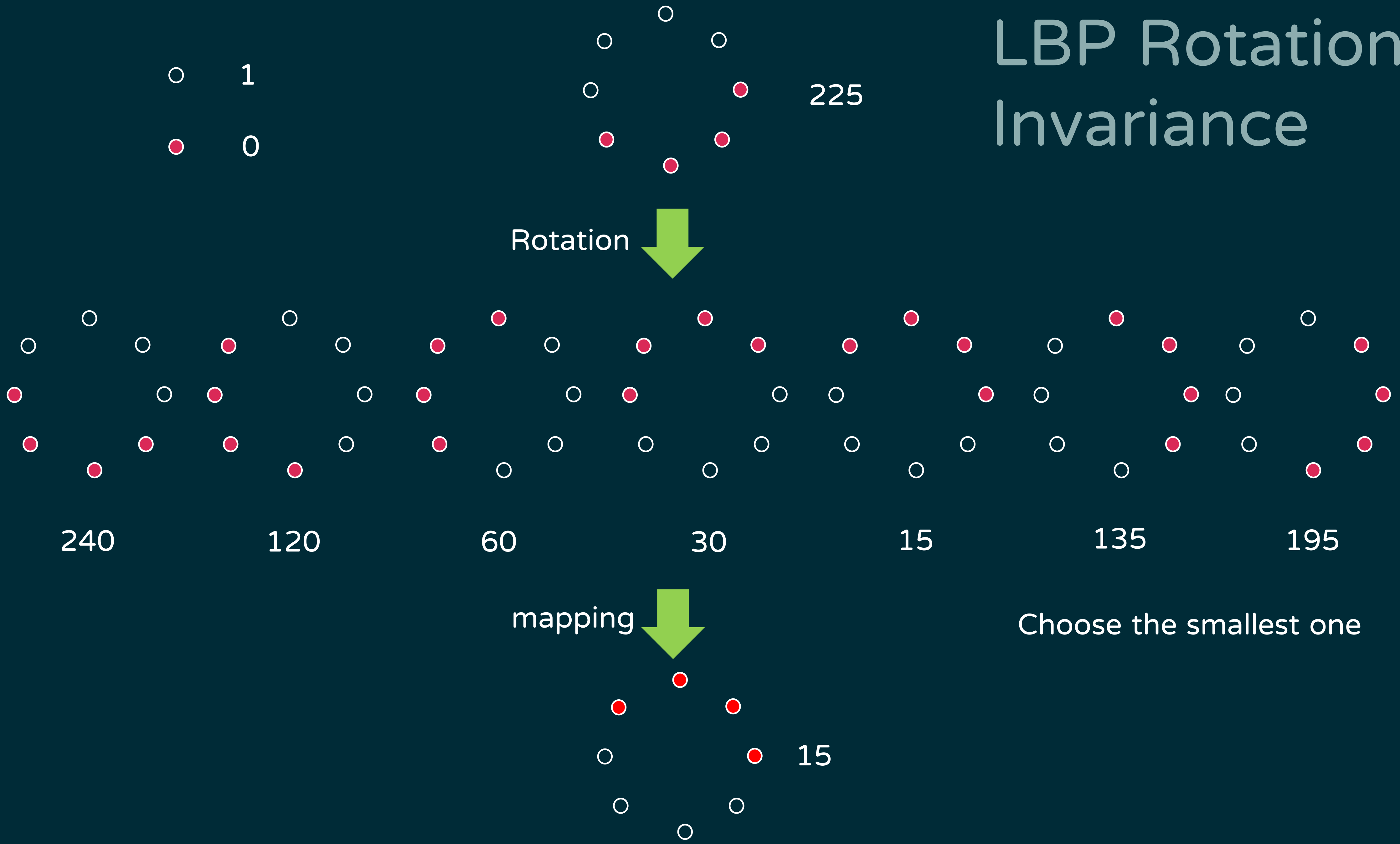
0 0 0 1 1 1 1 1

76	**93**	**92**
60	83	93
77	63	81

1	0	0
1	0	0
1	1	1

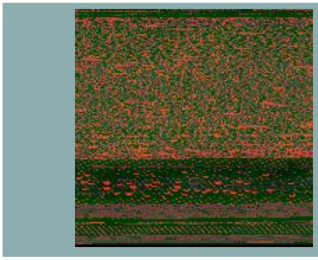
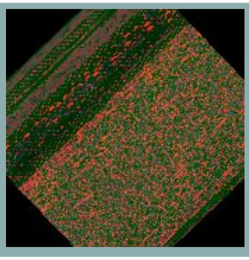
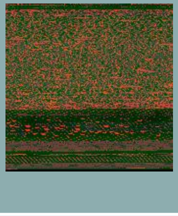
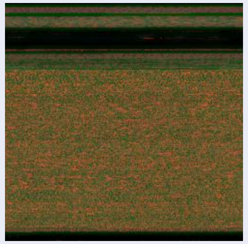
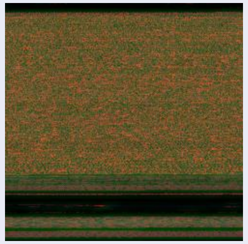
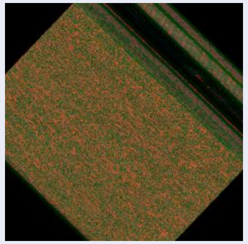
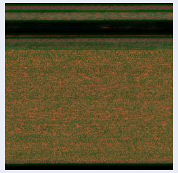
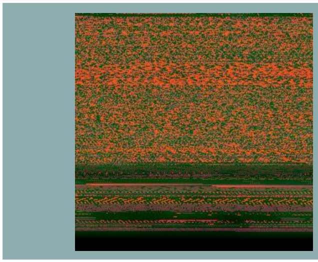
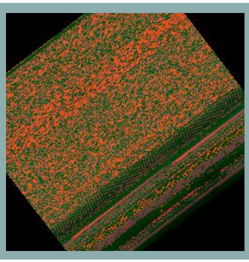
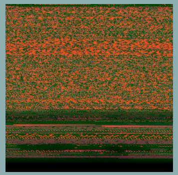
0 1 1 1 1 1 0 0

LBP Rotational Invariance



Choose the smallest one

Data Argumentation

Original	Flip	Rotate	Scale
			
			
			

Mem2Img

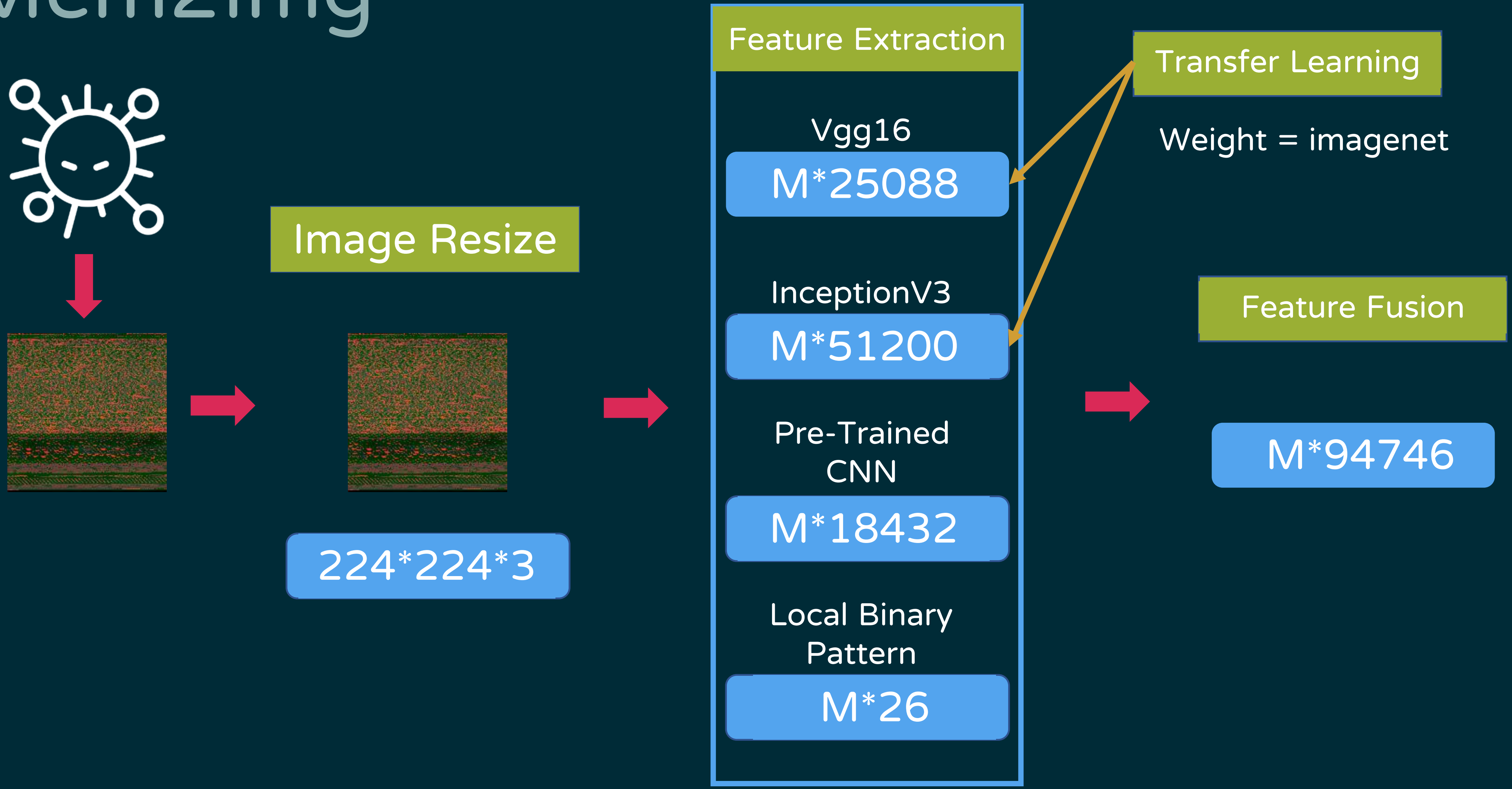


Image Resize

224*224*3

Feature Extraction

Vgg16

M*25088

InceptionV3

M*51200

Pre-Trained CNN

M*18432

Local Binary Pattern

M*26

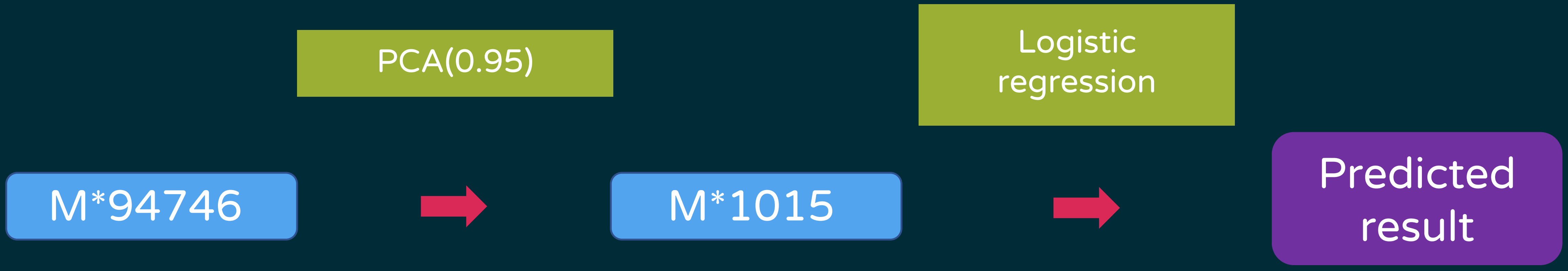
Transfer Learning

Weight = imagenet

Feature Fusion

M*94746

Mem2Ing(cont.)



M*94746

PCA(0.95)

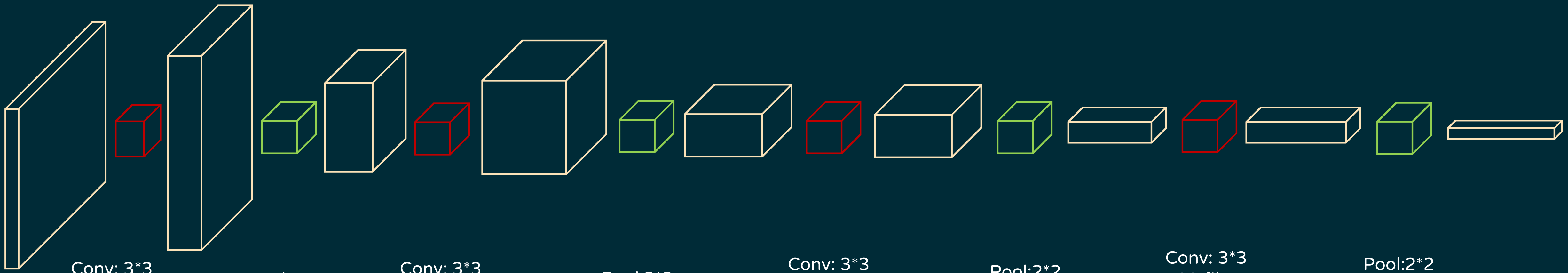
M*1015

Logistic regression

Predicted result

PlugX
Waterbear
Denis
CobaltStrike
…

CNN Architecture



Input:
224*224*3

Conv: 3*3
32 filters
Padding:2

222*222*32

Pool:2*2
Stride:2

111*111*32

Conv: 3*3
64 filters
Padding:2

109*109*64

Pool:2*2
Stride:2

54*54*64

Conv: 3*3
64 filters
Padding:2

52*52*64

Pool:2*2
Stride:2

26*26*64

Conv: 3*3
128 filters
Padding:2

24*24*128

Pool:2*2
Stride:2

12*12*128

Training parameter

- Training : Testing : 5:1
- 30 class classification
- 12569 memory blocks image(after data argumentation)
- CNN:
  - activation function : Relu
  - Batch normalization
  - Learning rate decay
  - Training ephocs:32
- Logistic regression
  - Class weight

Different Models’s Features

Model	Accuracy	Precision	Recall	F1 Score
Mem2Img	98.36%	98.51%	98.36%	98.38%
CNN	96.5%	97.09	96.5%	96.6%
Vgg16	96.73%	97.28%	96.7%	96.8%
Inception V3	95.8%	96.2%	95.8%	95.8%
LBP	84.8%	86.6%	84.8%	84.6%

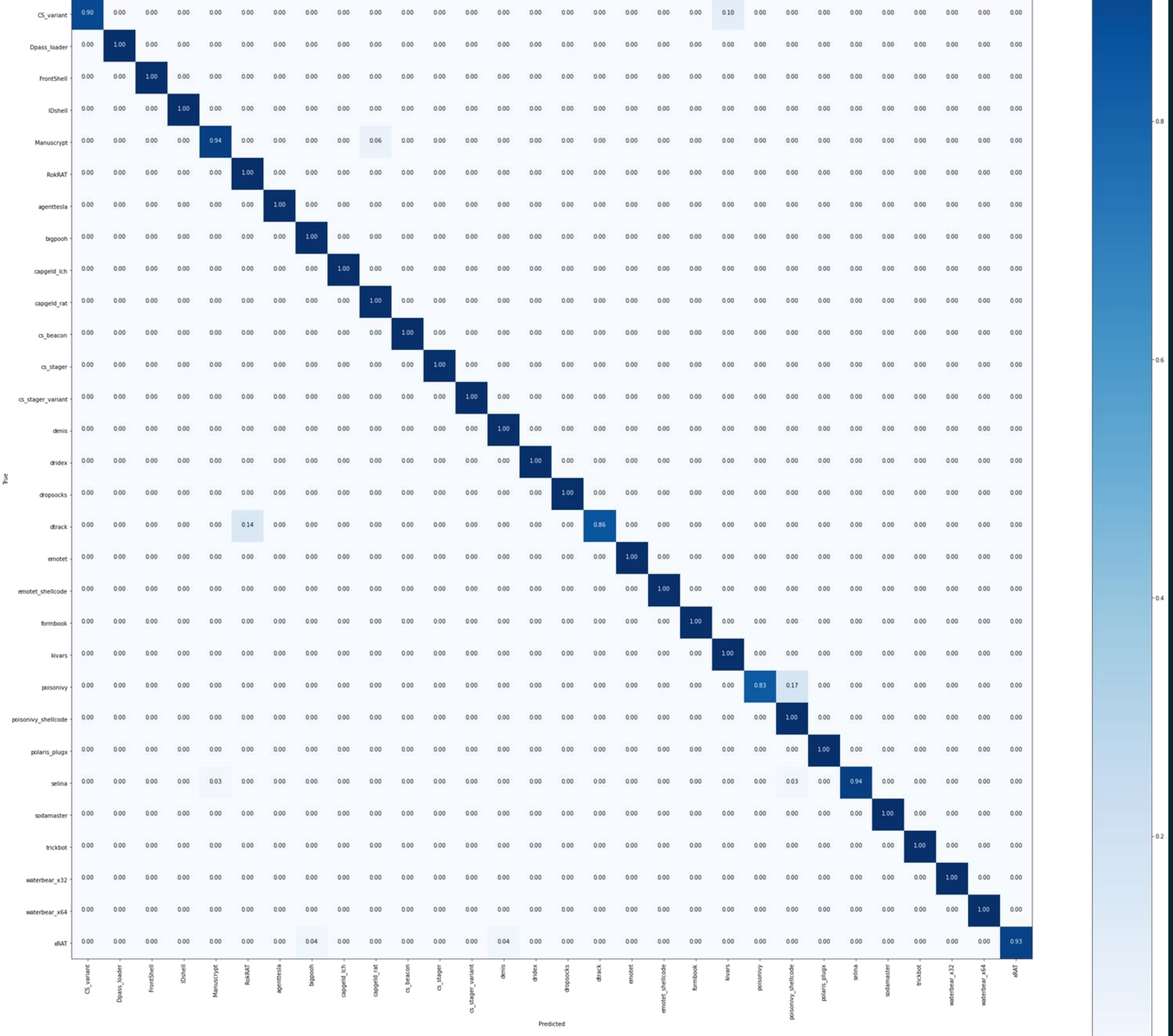
Different image

Model	Accuracy	Precision	Recall	F1 Score
RGB	98.13%	98.3%	98.13%	98.14%
RG (without Blue channel : Local Entropy)	92.23%	93.2%	92.23%	92.23%
Gray	88.8%	90.3%	88.8%	88.9%

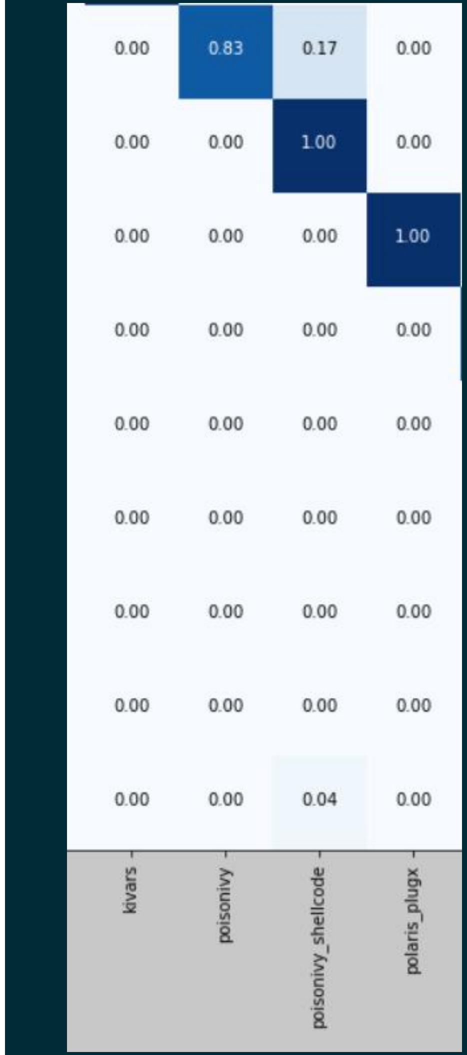
Different Algorithm

Model	Accuracy	Precision	Recall	F1 Score
Logistic Regression	98.36%	98.51%	98.36%	98.38%
SVM	98.36%	98.44%	98.36%	98.36%
Xgboost	94.17%	94.51%	94.17%	94.15%
Random Forest	93.7%	95%	93.7%	93.83%

Confusion matrix among 30 malware class

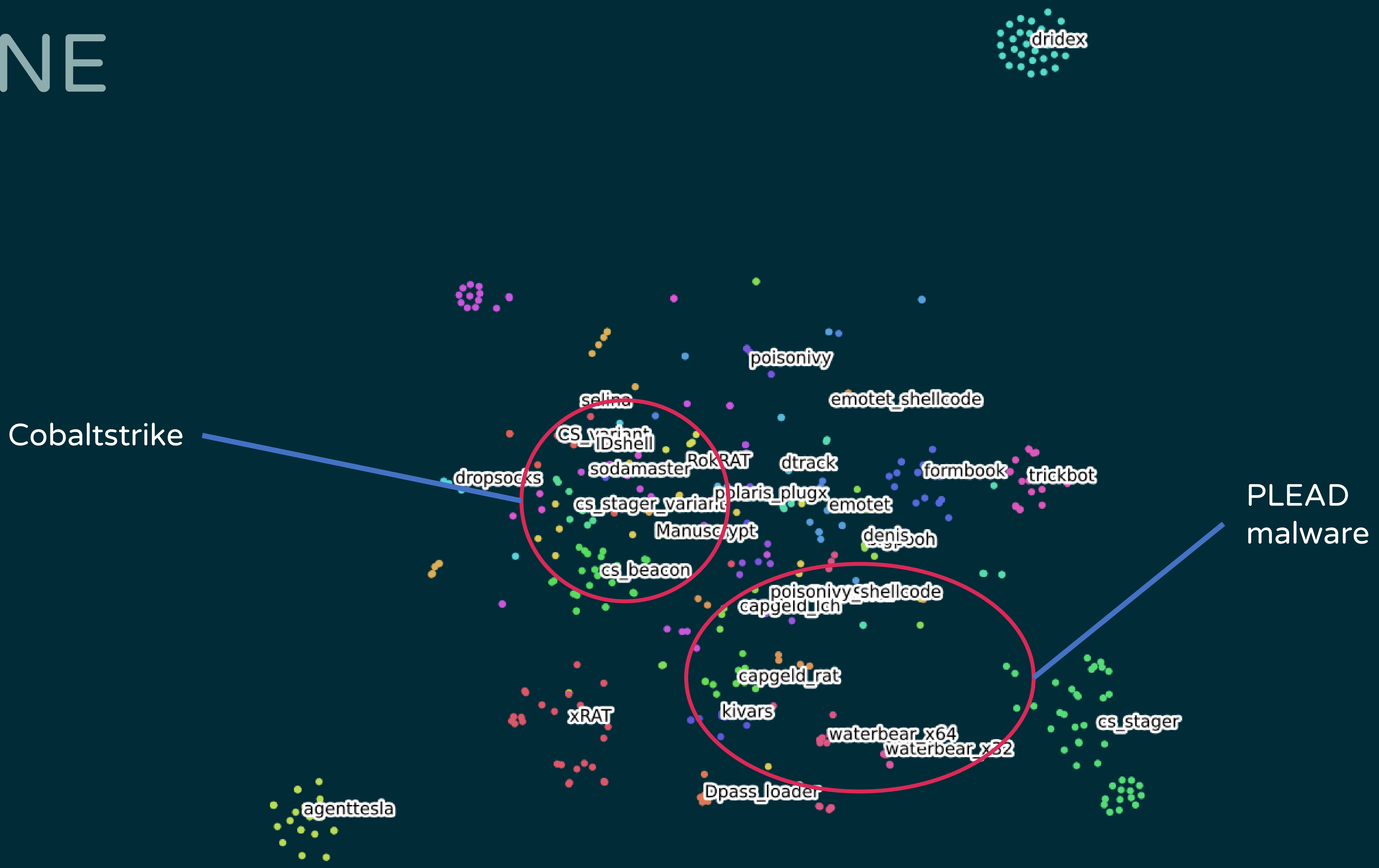


	CS_variant	Dpass_loader	FrontShell	IDshell	Manuscrypt	RokRAT	agenttesla	bigpooh	capgeld_lch	capgeld_rat	cs_beacon	cs_stager	cs_stager_variant	denis	dridex	dropsocks	dtrack	emotet	emotet_shellcode	formbook	kivars	poisonivy	poisonivy_shellcode	polaris_plugx	selina	sodamaster	trickbot	waterbear_x32	waterbear_x64	xRAT
CS_variant	0.90	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.10	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
Dpass_loader	0.00	1.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
FrontShell	0.00	0.00	1.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
IDshell	0.00	0.00	0.00	1.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
Manuscrypt	0.00	0.00	0.00	0.00	0.94	0.00	0.00	0.00	0.00	0.06	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
RokRAT	0.00	0.00	0.00	0.00	0.00	1.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
agenttesla	0.00	0.00	0.00	0.00	0.00	0.00	1.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
bigpooh	0.00	0.00	0.00	0.00	0.00	0.00	0.00	1.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
capgeld_lch	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	1.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
capgeld_rat	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	1.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
cs_beacon	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	1.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
cs_stager	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	1.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
cs_stager_variant	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	1.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
denis	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	1.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
dridex	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	1.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
dropsocks	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	1.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
dtrack	0.00	0.00	0.00	0.00	0.00	0.14	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.86	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
emotet	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	1.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
emotet_shellcode	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	1.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
formbook	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	1.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
kivars	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	1.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
poisonivy	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.83	0.17	0.00	0.00	0.00	0.00	0.00	0.00	0.00
poisonivy_shellcode	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	1.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
polaris_plugx	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	1.00	0.00	0.00	0.00	0.00	0.00	0.00
selina	0.00	0.00	0.00	0.00	0.03	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.03	0.00	0.94	0.00	0.00	0.00	0.00	0.00
sodamaster	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	1.00	0.00	0.00	0.00	0.00
trickbot	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	1.00	0.00	0.00	0.00
waterbear_x32	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	1.00	0.00	0.00
waterbear_x64	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	1.00	0.00
xRAT	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.04	0.00	0.00	0.00	0.00	0.00	0.04	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.93



kivars	poisonivy	poisonivy_shellcode	polaris_plugx
0.00	0.83	0.17	0.00
0.00	0.00	1.00	0.00
0.00	0.00	0.00	1.00
0.00	0.00	0.00	0.00
0.00	0.00	0.00	0.00
0.00	0.00	0.00	0.00
0.00	0.00	0.00	0.00
0.00	0.00	0.00	0.00
0.00	0.00	0.04	0.00

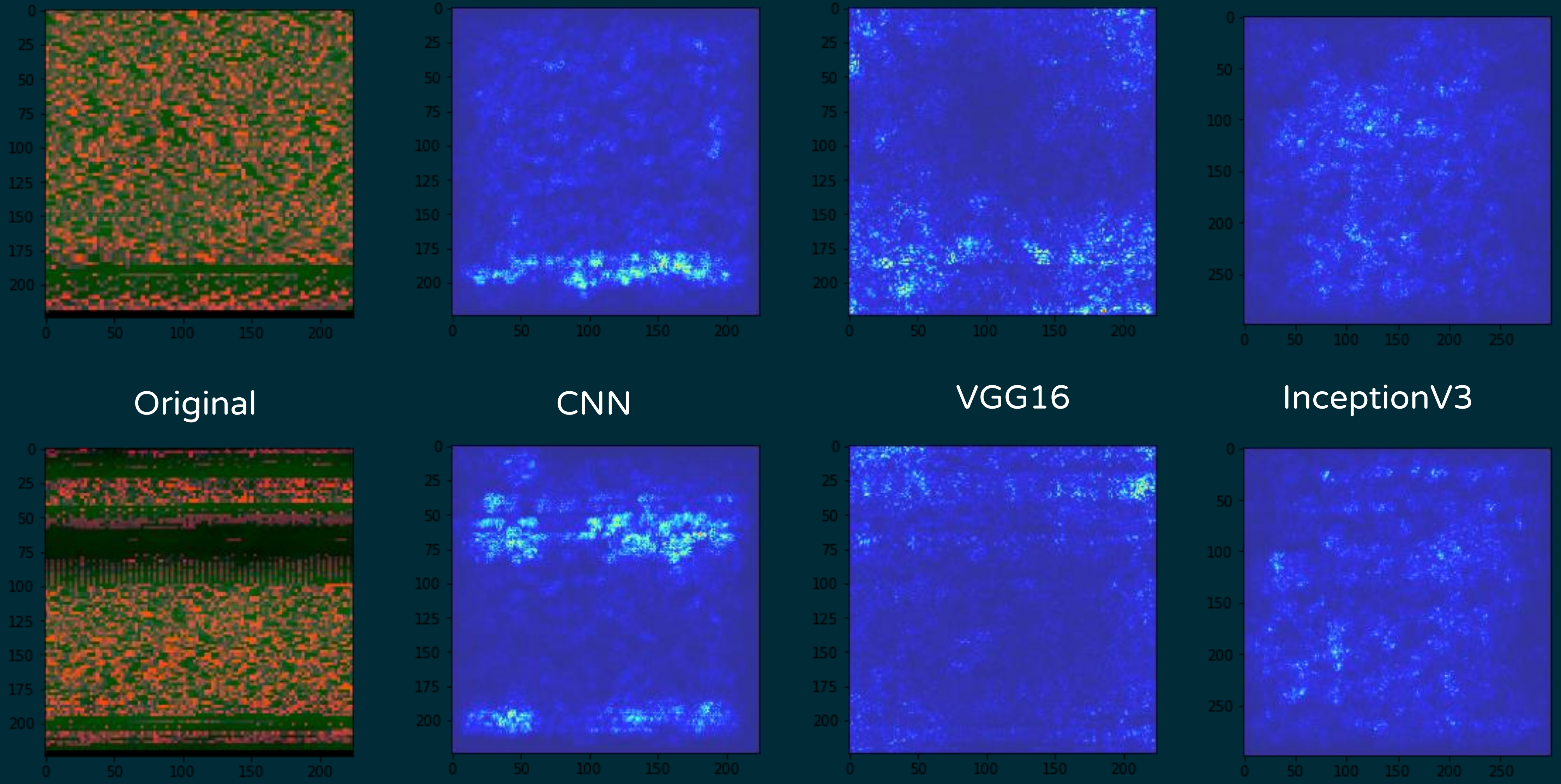
t-SNE



Cobaltstrike

PLEAD
malware

Saliency map



Original

CNN

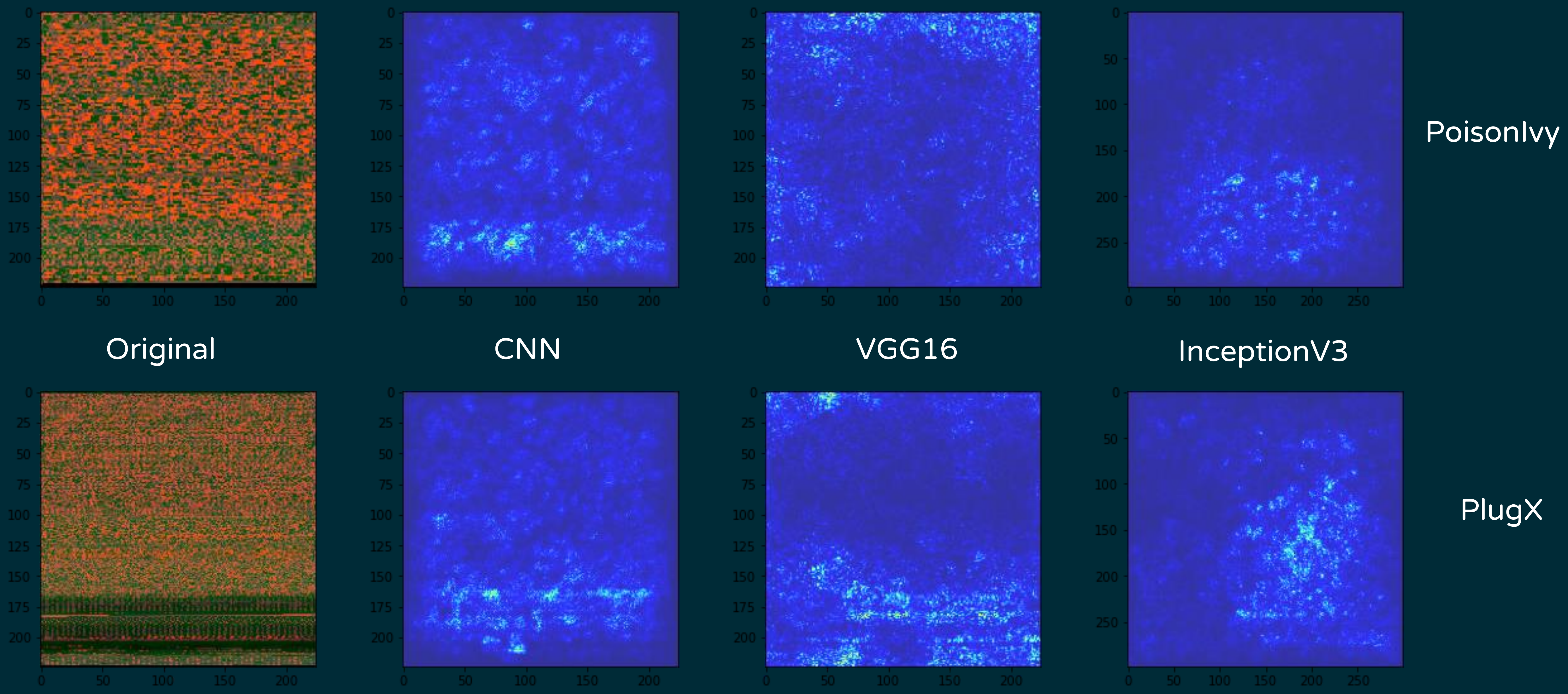
VGG16

InceptionV3

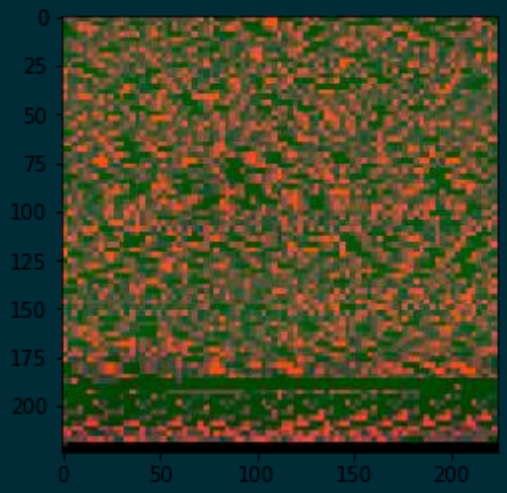
Waterbear
_x64

Capgeld
_loader

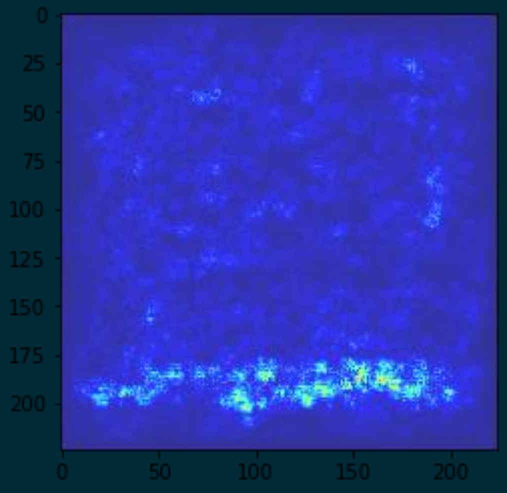
Saliency map



Saliency map - Waterbear

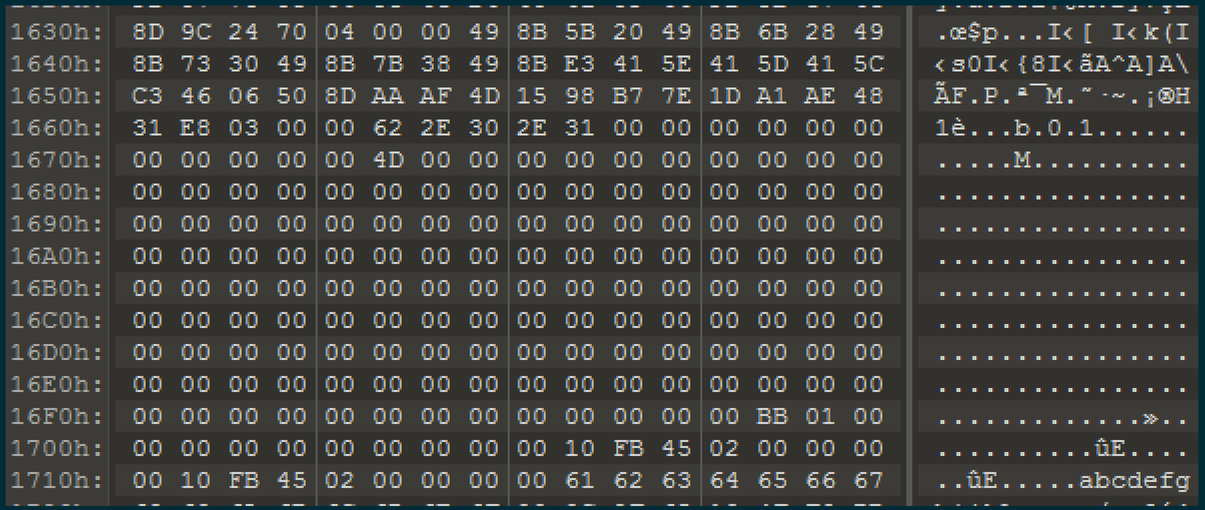


Original



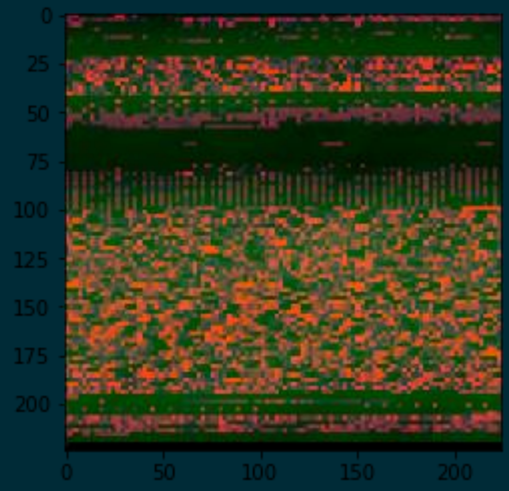
CNN

Config block of the waterbear stager

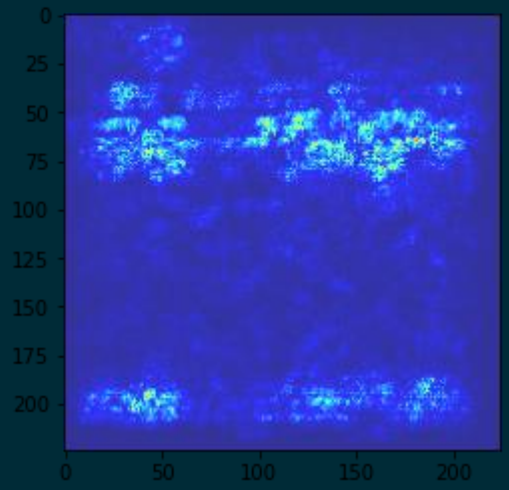


```
1630h:  8D 9C 24 70 04 00 00 49 8B 5B 20 49 8B 6B 28 49  .œ$p...I‹[ I‹k(I
1640h:  8B 73 30 49 8B 7B 38 49 8B E3 41 5E 41 5D 41 5C  ‹s0I‹{8I‹ãA^A]A\
1650h:  C3 46 06 50 8D AA AF 4D 15 98 B7 7E 1D A1 AE 48  ÃF.P.ª¯M.˜·~.¡®H
1660h:  31 E8 03 00 00 62 2E 30 2E 31 00 00 00 00 00 00  1è...b.0.1......
1670h:  00 00 00 00 00 4D 00 00 00 00 00 00 00 00 00 00  .....M..........
1680h:  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
1690h:  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
16A0h:  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
16B0h:  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
16C0h:  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
16D0h:  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
16E0h:  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
16F0h:  00 00 00 00 00 00 00 00 00 00 00 00 00 BB 01 00  .............»..
1700h:  00 00 00 00 00 00 00 00 00 10 FB 45 02 00 00 00  ..........ûE....
1710h:  00 10 FB 45 02 00 00 00 00 61 62 63 64 65 66 67  ..ûE.....abcdefg
```

Saliency map - Capgeld Loader

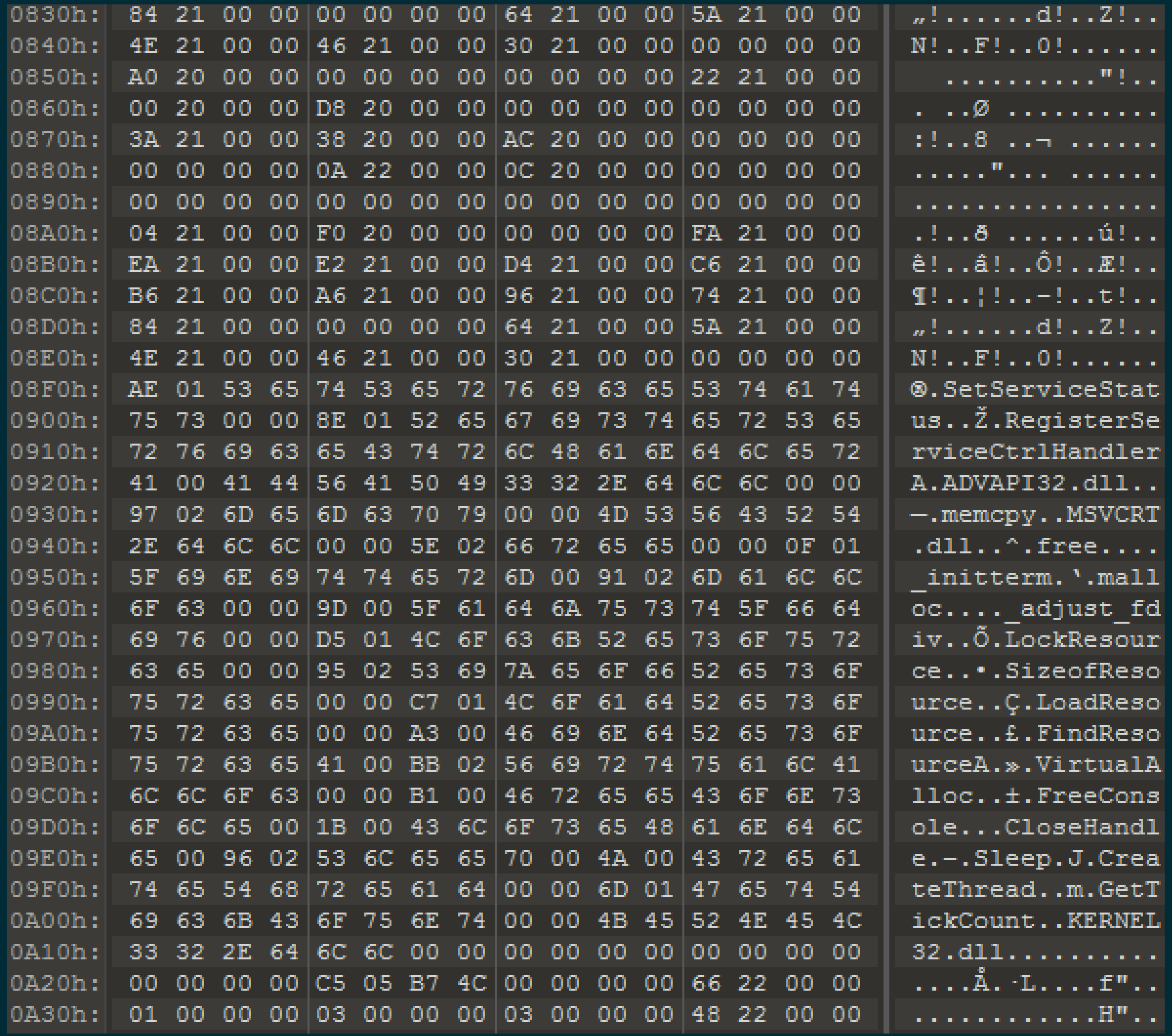


Original



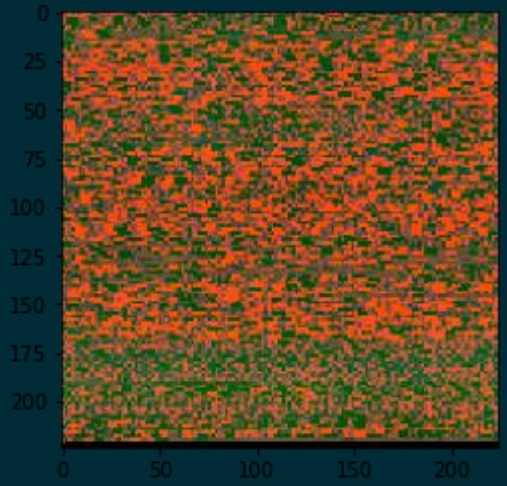
CNN

.rdata section of the Capgeld Loader

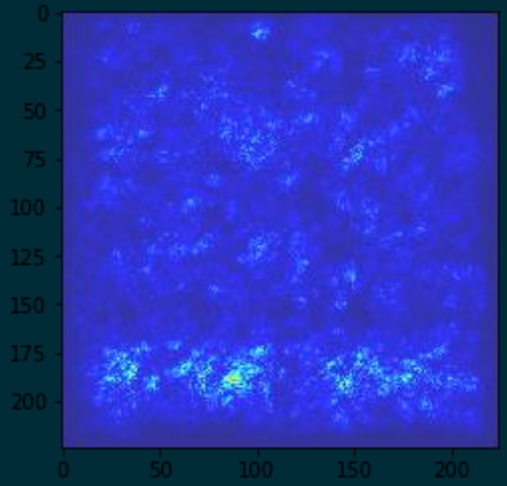


```
0830h:  84 21 00 00 00 00 00 00 64 21 00 00 5A 21 00 00  „!......d!..Z!..
0840h:  4E 21 00 00 46 21 00 00 30 21 00 00 00 00 00 00  N!..F!..0!......
0850h:  A0 20 00 00 00 00 00 00 00 00 00 00 22 21 00 00    .........."!..
0860h:  00 20 00 00 D8 20 00 00 00 00 00 00 00 00 00 00  . ..Ø ..........
0870h:  3A 21 00 00 38 20 00 00 AC 20 00 00 00 00 00 00  :!..8 ..¬ ......
0880h:  00 00 00 00 0A 22 00 00 0C 20 00 00 00 00 00 00  ....."... ......
0890h:  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
08A0h:  04 21 00 00 F0 20 00 00 00 00 00 00 FA 21 00 00  .!..ð ......ú!..
08B0h:  EA 21 00 00 E2 21 00 00 D4 21 00 00 C6 21 00 00  ê!..â!..Ô!..Æ!..
08C0h:  B6 21 00 00 A6 21 00 00 96 21 00 00 74 21 00 00  ¶!..¦!..–!..t!..
08D0h:  84 21 00 00 00 00 00 00 64 21 00 00 5A 21 00 00  „!......d!..Z!..
08E0h:  4E 21 00 00 46 21 00 00 30 21 00 00 00 00 00 00  N!..F!..0!......
08F0h:  AE 01 53 65 74 53 65 72 76 69 63 65 53 74 61 74  ®.SetServiceStat
0900h:  75 73 00 00 8E 01 52 65 67 69 73 74 65 72 53 65  us..Ž.RegisterSe
0910h:  72 76 69 63 65 43 74 72 6C 48 61 6E 64 6C 65 72  rviceCtrlHandler
0920h:  41 00 41 44 56 41 50 49 33 32 2E 64 6C 6C 00 00  A.ADVAPI32.dll..
0930h:  97 02 6D 65 6D 63 70 79 00 00 4D 53 56 43 52 54  —.memcpy..MSVCRT
0940h:  2E 64 6C 6C 00 00 5E 02 66 72 65 65 00 00 0F 01  .dll..^.free....
0950h:  5F 69 6E 69 74 74 65 72 6D 00 91 02 6D 61 6C 6C  _initterm.‘.mall
0960h:  6F 63 00 00 9D 00 5F 61 64 6A 75 73 74 5F 66 64  oc...._adjust_fd
0970h:  69 76 00 00 D5 01 4C 6F 63 6B 52 65 73 6F 75 72  iv..Õ.LockResour
0980h:  63 65 00 00 95 02 53 69 7A 65 6F 66 52 65 73 6F  ce..•.SizeofReso
0990h:  75 72 63 65 00 00 C7 01 4C 6F 61 64 52 65 73 6F  urce..Ç.LoadReso
09A0h:  75 72 63 65 00 00 A3 00 46 69 6E 64 52 65 73 6F  urce..£.FindReso
09B0h:  75 72 63 65 41 00 BB 02 56 69 72 74 75 61 6C 41  urceA.».VirtualA
09C0h:  6C 6C 6F 63 00 00 B1 00 46 72 65 65 43 6F 6E 73  lloc..±.FreeCons
09D0h:  6F 6C 65 00 1B 00 43 6C 6F 73 65 48 61 6E 64 6C  ole...CloseHandl
09E0h:  65 00 96 02 53 6C 65 65 70 00 4A 00 43 72 65 61  e.–.Sleep.J.Crea
09F0h:  74 65 54 68 72 65 61 64 00 00 6D 01 47 65 74 54  teThread..m.GetT
0A00h:  69 63 6B 43 6F 75 6E 74 00 00 4B 45 52 4E 45 4C  ickCount..KERNEL
0A10h:  33 32 2E 64 6C 6C 00 00 00 00 00 00 00 00 00 00  32.dll..........
0A20h:  00 00 00 00 C5 05 B7 4C 00 00 00 00 66 22 00 00  ....Å.·L....f"..
0A30h:  01 00 00 00 03 00 00 00 03 00 00 00 48 22 00 00  ............H"..
```

Saliency map - Phamtom Ivy

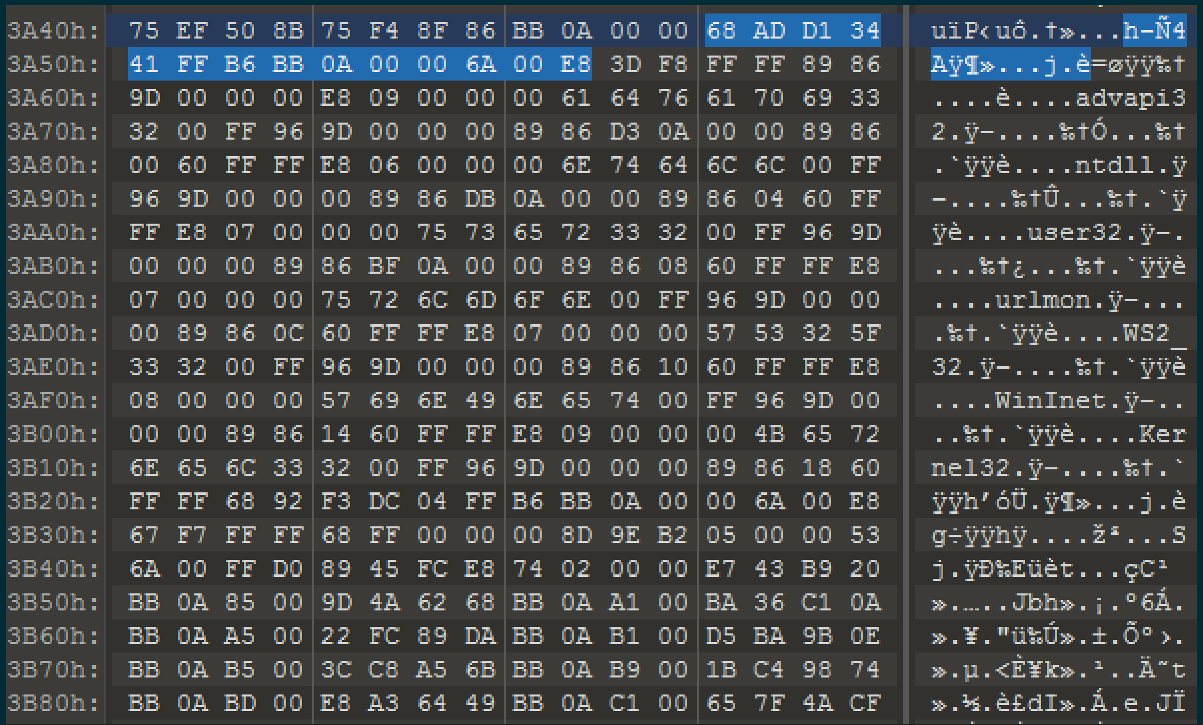


Original



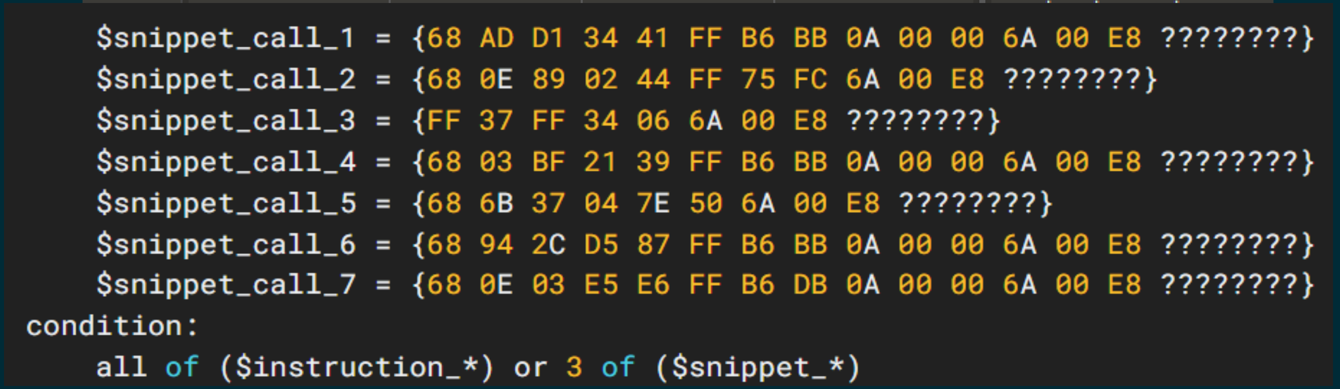
CNN

Some shellcode snippets of Phamtom Ivy



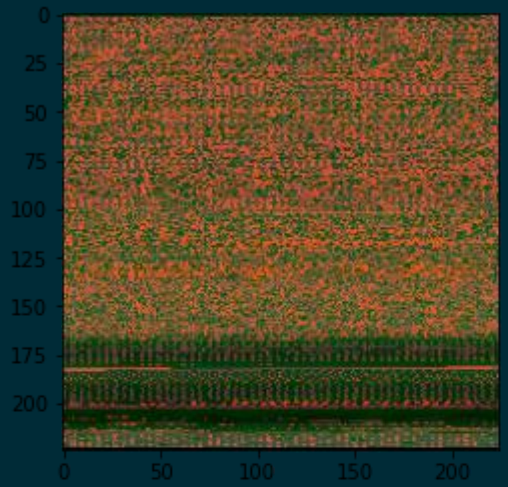
```
3A40h:  75 EF 50 8B 75 F4 8F 86 BB 0A 00 00 68 AD D1 34   uïP‹uô.†»...h-Ñ4
3A50h:  41 FF B6 BB 0A 00 00 6A 00 E8 3D F8 FF FF 89 86   Aÿ¶»...j.è=øÿÿ‰†
3A60h:  9D 00 00 00 E8 09 00 00 00 61 64 76 61 70 69 33   ....è....advapi3
3A70h:  32 00 FF 96 9D 00 00 00 89 86 D3 0A 00 00 89 86   2.ÿ–....‰†Ó...‰†
3A80h:  00 60 FF FF E8 06 00 00 00 6E 74 64 6C 6C 00 FF   .`ÿÿè....ntdll.ÿ
3A90h:  96 9D 00 00 00 89 86 DB 0A 00 00 89 86 04 60 FF   –....‰†Û...‰†.`ÿ
3AA0h:  FF E8 07 00 00 00 75 73 65 72 33 32 00 FF 96 9D   ÿè....user32.ÿ–.
3AB0h:  00 00 00 89 86 BF 0A 00 00 89 86 08 60 FF FF E8   ...‰†¿...‰†.`ÿÿè
3AC0h:  07 00 00 00 75 72 6C 6D 6F 6E 00 FF 96 9D 00 00   ....urlmon.ÿ–...
3AD0h:  00 89 86 0C 60 FF FF E8 07 00 00 00 57 53 32 5F   .‰†.`ÿÿè....WS2_
3AE0h:  33 32 00 FF 96 9D 00 00 00 89 86 10 60 FF FF E8   32.ÿ–....‰†.`ÿÿè
3AF0h:  08 00 00 00 57 69 6E 49 6E 65 74 00 FF 96 9D 00   ....WinInet.ÿ–..
3B00h:  00 00 89 86 14 60 FF FF E8 09 00 00 00 4B 65 72   ..‰†.`ÿÿè....Ker
3B10h:  6E 65 6C 33 32 00 FF 96 9D 00 00 00 89 86 18 60   nel32.ÿ–....‰†.`
3B20h:  FF FF 68 92 F3 DC 04 FF B6 BB 0A 00 00 6A 00 E8   ÿÿh’óÜ.ÿ¶»...j.è
3B30h:  67 F7 FF FF 68 FF 00 00 00 8D 9E B2 05 00 00 53   g÷ÿÿhÿ....ž²...S
3B40h:  6A 00 FF D0 89 45 FC E8 74 02 00 00 E7 43 B9 20   j.ÿÐ‰Eüèt...çC¹ 
3B50h:  BB 0A 85 00 9D 4A 62 68 BB 0A A1 00 BA 36 C1 0A   ».….Jbh».¡.º6Á.
3B60h:  BB 0A A5 00 22 FC 89 DA BB 0A B1 00 D5 BA 9B 0E   ».¥."ü‰Ú».±.Õº›.
3B70h:  BB 0A B5 00 3C C8 A5 6B BB 0A B9 00 1B C4 98 74   ».µ.<È¥k».¹..Ä˜t
3B80h:  BB 0A BD 00 E8 A3 64 49 BB 0A C1 00 65 7F 4A CF   ».½.è£dI».Á.e.JÏ
```

Yara rules of Phhamtom Ivy

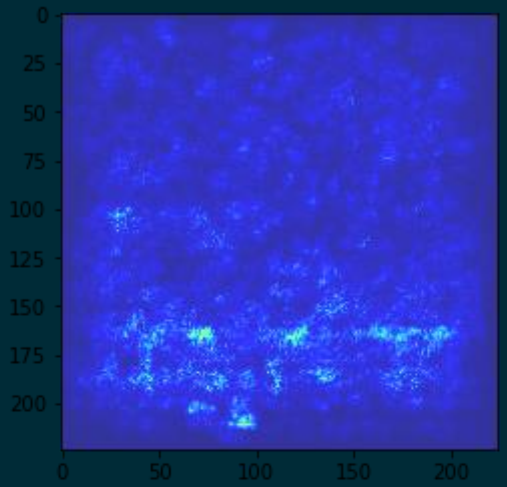


```
    $snippet_call_1 = {68 AD D1 34 41 FF B6 BB 0A 00 00 6A 00 E8 ????????}
    $snippet_call_2 = {68 0E 89 02 44 FF 75 FC 6A 00 E8 ????????}
    $snippet_call_3 = {FF 37 FF 34 06 6A 00 E8 ????????}
    $snippet_call_4 = {68 03 BF 21 39 FF B6 BB 0A 00 00 6A 00 E8 ????????}
    $snippet_call_5 = {68 6B 37 04 7E 50 6A 00 E8 ????????}
    $snippet_call_6 = {68 94 2C D5 87 FF B6 BB 0A 00 00 6A 00 E8 ????????}
    $snippet_call_7 = {68 0E 03 E5 E6 FF B6 DB 0A 00 00 6A 00 E8 ????????}
condition:
    all of ($instruction_*) or 3 of ($snippet_*)
```

Saliency map - Mustang Panda PlugX

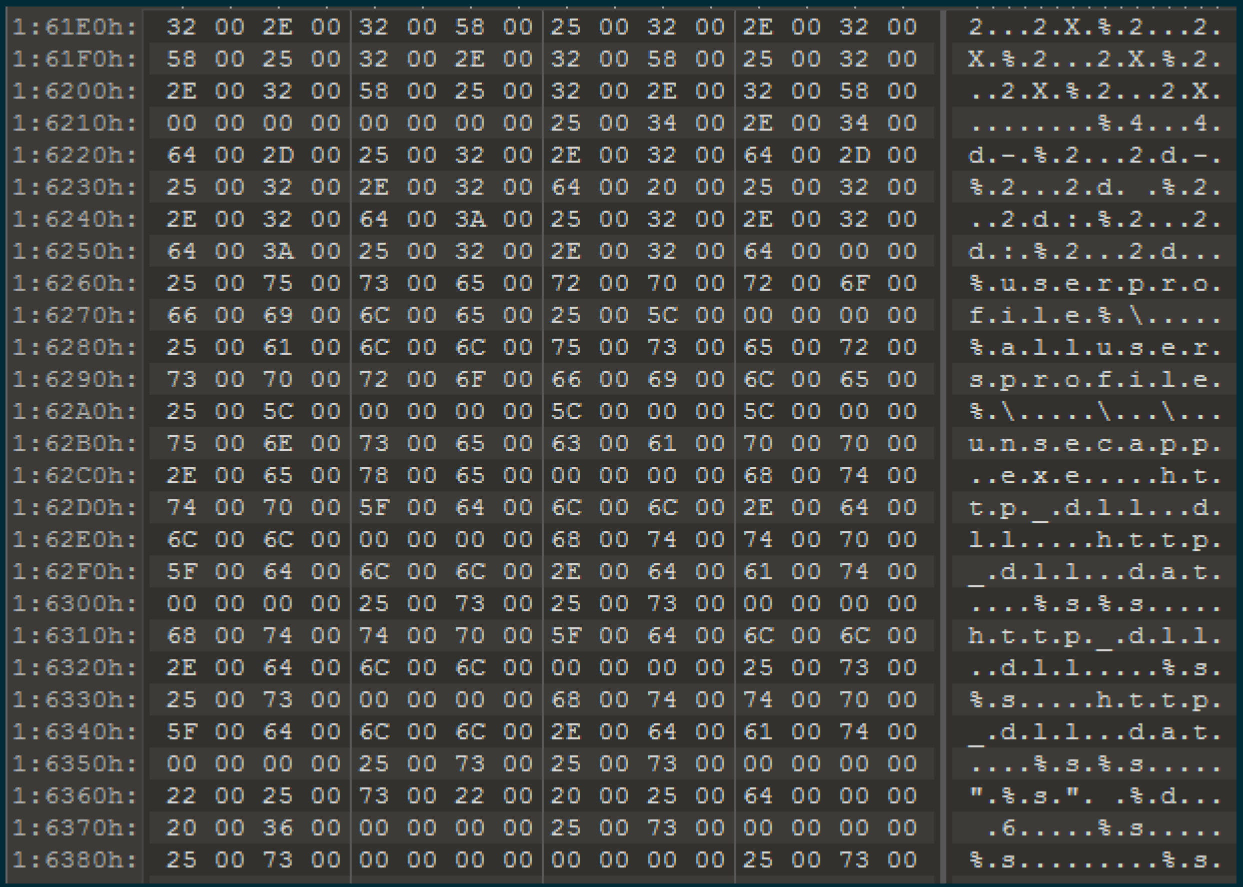


Original



CNN

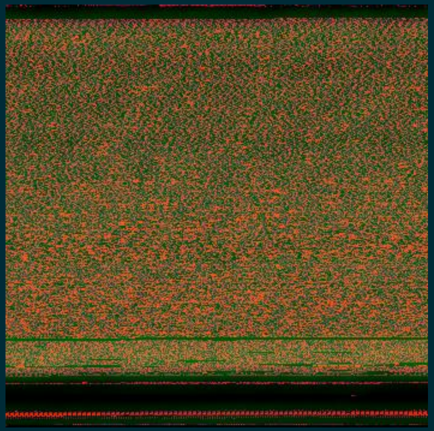
Stack strings of PlugX

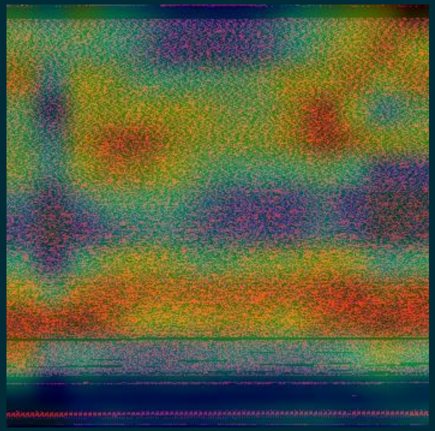


```
1:61E0h: 32 00 2E 00 32 00 58 00 25 00 32 00 2E 00 32 00  2...2.X.%.2...2.
1:61F0h: 58 00 25 00 32 00 2E 00 32 00 58 00 25 00 32 00  X.%.2...2.X.%.2.
1:6200h: 2E 00 32 00 58 00 25 00 32 00 2E 00 32 00 58 00  ..2.X.%.2...2.X.
1:6210h: 00 00 00 00 00 00 00 00 25 00 34 00 2E 00 34 00  ........%.4...4.
1:6220h: 64 00 2D 00 25 00 32 00 2E 00 32 00 64 00 2D 00  d.-.%.2...2.d.-.
1:6230h: 25 00 32 00 2E 00 32 00 64 00 20 00 25 00 32 00  %.2...2.d. .%.2.
1:6240h: 2E 00 32 00 64 00 3A 00 25 00 32 00 2E 00 32 00  ..2.d.:.%.2...2.
1:6250h: 64 00 3A 00 25 00 32 00 2E 00 32 00 64 00 00 00  d.:.%.2...2.d...
1:6260h: 25 00 75 00 73 00 65 00 72 00 70 00 72 00 6F 00  %.u.s.e.r.p.r.o.
1:6270h: 66 00 69 00 6C 00 65 00 25 00 5C 00 00 00 00 00  f.i.l.e.%.\.....
1:6280h: 25 00 61 00 6C 00 6C 00 75 00 73 00 65 00 72 00  %.a.l.l.u.s.e.r.
1:6290h: 73 00 70 00 72 00 6F 00 66 00 69 00 6C 00 65 00  s.p.r.o.f.i.l.e.
1:62A0h: 25 00 5C 00 00 00 00 00 5C 00 00 00 5C 00 00 00  %.\.....\...\...
1:62B0h: 75 00 6E 00 73 00 65 00 63 00 61 00 70 00 70 00  u.n.s.e.c.a.p.p.
1:62C0h: 2E 00 65 00 78 00 65 00 00 00 00 00 68 00 74 00  ..e.x.e.....h.t.
1:62D0h: 74 00 70 00 5F 00 64 00 6C 00 6C 00 2E 00 64 00  t.p._.d.l.l...d.
1:62E0h: 6C 00 6C 00 00 00 00 00 68 00 74 00 74 00 70 00  l.l.....h.t.t.p.
1:62F0h: 5F 00 64 00 6C 00 6C 00 2E 00 64 00 61 00 74 00  _.d.l.l...d.a.t.
1:6300h: 00 00 00 00 25 00 73 00 25 00 73 00 00 00 00 00  ....%.s.%.s.....
1:6310h: 68 00 74 00 74 00 70 00 5F 00 64 00 6C 00 6C 00  h.t.t.p._.d.l.l.
1:6320h: 2E 00 64 00 6C 00 6C 00 00 00 00 00 25 00 73 00  ..d.l.l.....%.s.
1:6330h: 25 00 73 00 00 00 00 00 68 00 74 00 74 00 70 00  %.s.....h.t.t.p.
1:6340h: 5F 00 64 00 6C 00 6C 00 2E 00 64 00 61 00 74 00  _.d.l.l...d.a.t.
1:6350h: 00 00 00 00 25 00 73 00 25 00 73 00 00 00 00 00  ....%.s.%.s.....
1:6360h: 22 00 25 00 73 00 22 00 20 00 25 00 64 00 00 00  ".%.s.". .%.d...
1:6370h: 20 00 36 00 00 00 00 00 25 00 73 00 00 00 00 00   .6.....%.s.....
1:6380h: 25 00 73 00 00 00 00 00 00 00 00 00 25 00 73 00  %.s.........%.s.
```

Grad-cam Analysis

Dridex





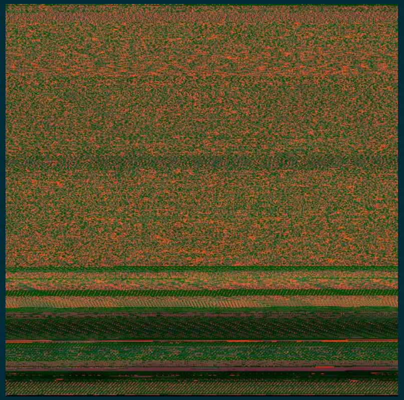
C2 parsing function
And API Spam Bypass

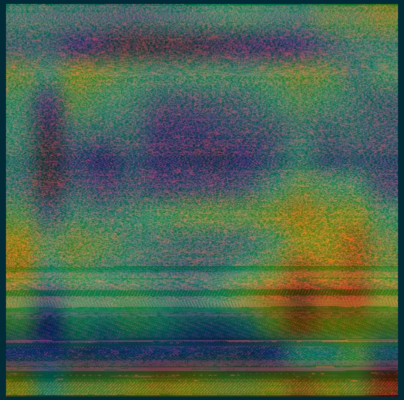
Some decode function
before .rdata section

Raw image

Heatmap over raw image

Cobalstrike
Beacon

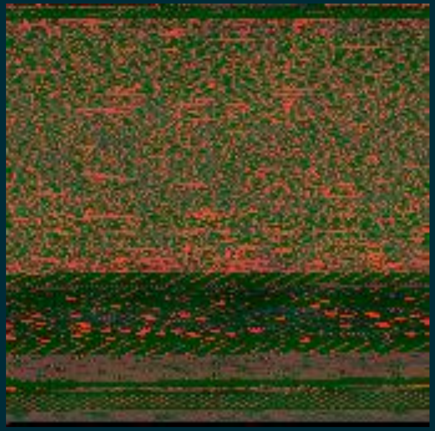


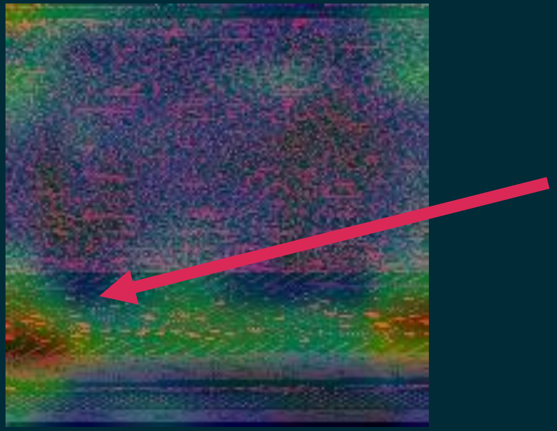


Part .rdata section and
part .data section

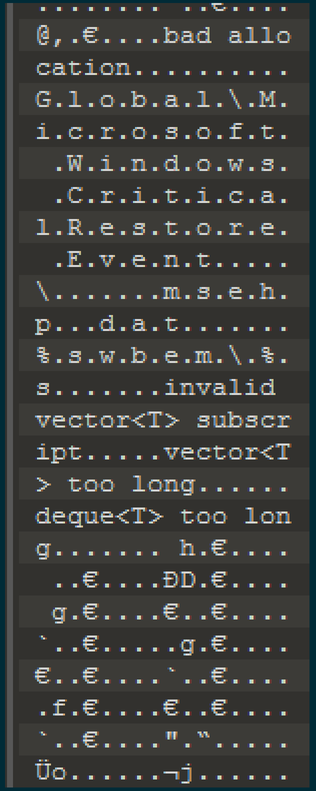
Grad-cam Analysis

Dpass loader





Unique strings block

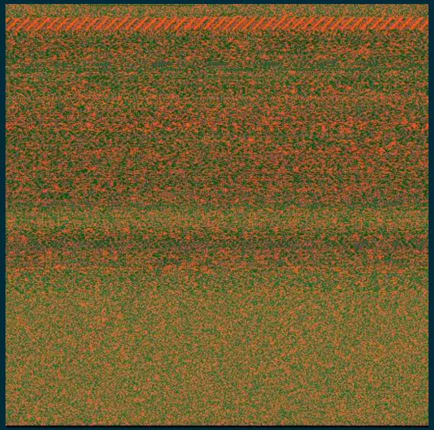


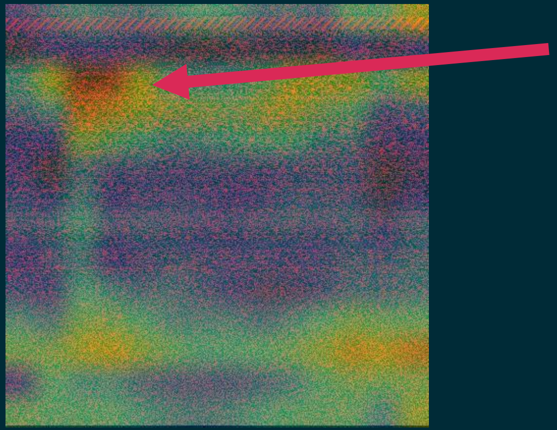
```
........ ..€....
@,.€....bad allo
cation..........
G.l.o.b.a.l.\.M.
i.c.r.o.s.o.f.t.
 .W.i.n.d.o.w.s.
 .C.r.i.t.i.c.a.
l.R.e.s.t.o.r.e.
 .E.v.e.n.t.....
\.......m.s.e.h.
p...d.a.t.......
%.s.w.b.e.m.\.%.
s.......invalid 
vector<T> subscr
ipt.....vector<T
> too long......
deque<T> too lon
g....... h.€....
 ..€....ÐD.€....
 g.€....€..€....
`..€.....g.€....
€..€....`..€....
.f.€....€..€....
`..€....".".....
Üo......¬j......
```

Raw image

Heatmap over raw image

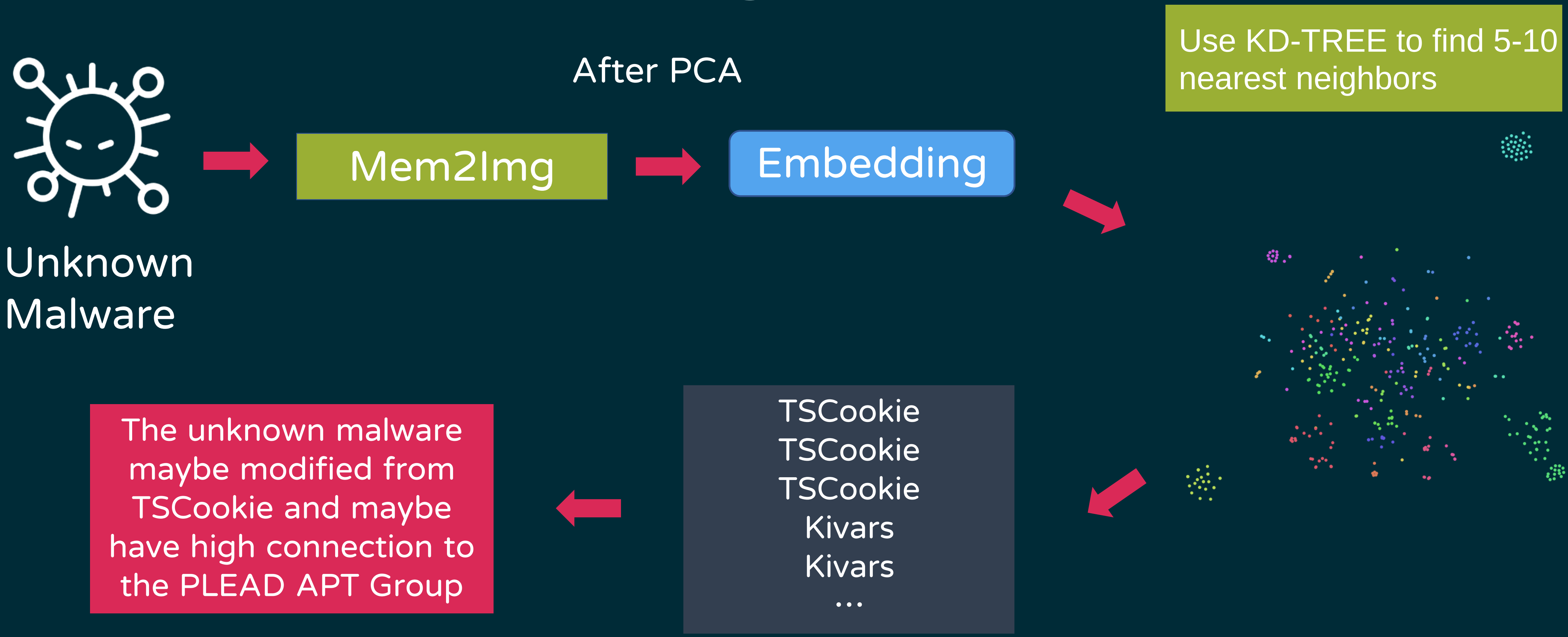
formbook





Obfuscated stack strings

Zero-shot Learning



Unknown Malware

Mem2Img

After PCA

Embedding

Use KD-TREE to find 5-10 nearest neighbors

TSCookie
TSCookie
TSCookie
Kivars
Kivars
…

The unknown malware maybe modified from TSCookie and maybe have high connection to the PLEAD APT Group

when we input the same unknown malware in to Mem2Img next time, the nearest neighbors may be the unknown malware input last time, and they can be new class when they have reached a certain amount. No need to retrain a new model!

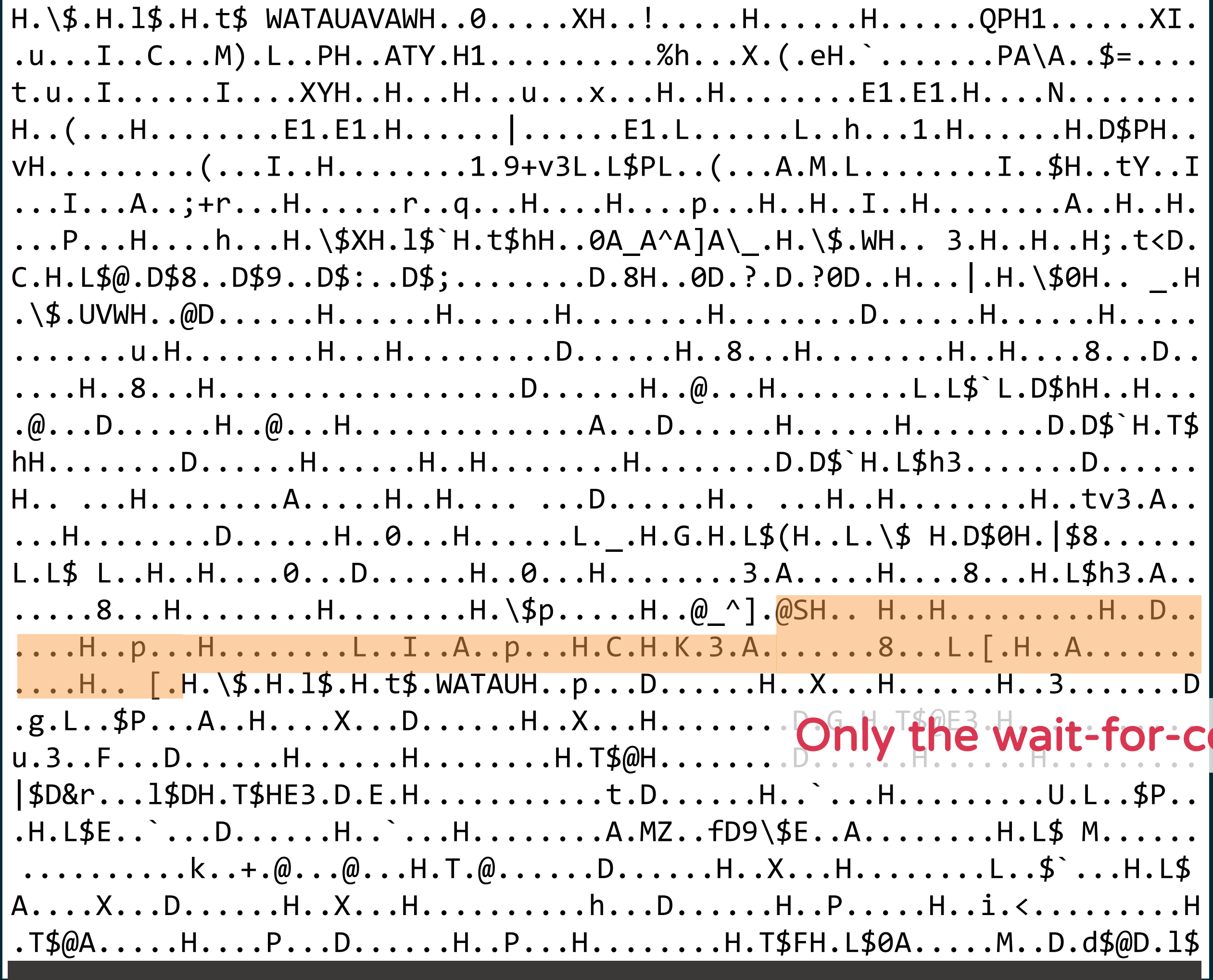
Zero-shot Learning

- Jinhospy used by APT37
    - [RokRAT RokRAT Manuscrypt Selina RokRAT]
- plugX_fast
    - [polaris_plugx polaris_plugx poisonivy poisonivy poisonivy]
- Plugx_variant
    - [polaris_plugx polaris_plugx polaris_plugx polaris_plugx poisonivy]
- TEBShell
    - [APT10’Cs loader APT10’Cs loader …]
- P8RAT
    - [xRAT xRAT xRAT …]
- Framecacher used by Chinese APT
    - [Selina Selina Selina Selina Selina]

Adversarial Attack

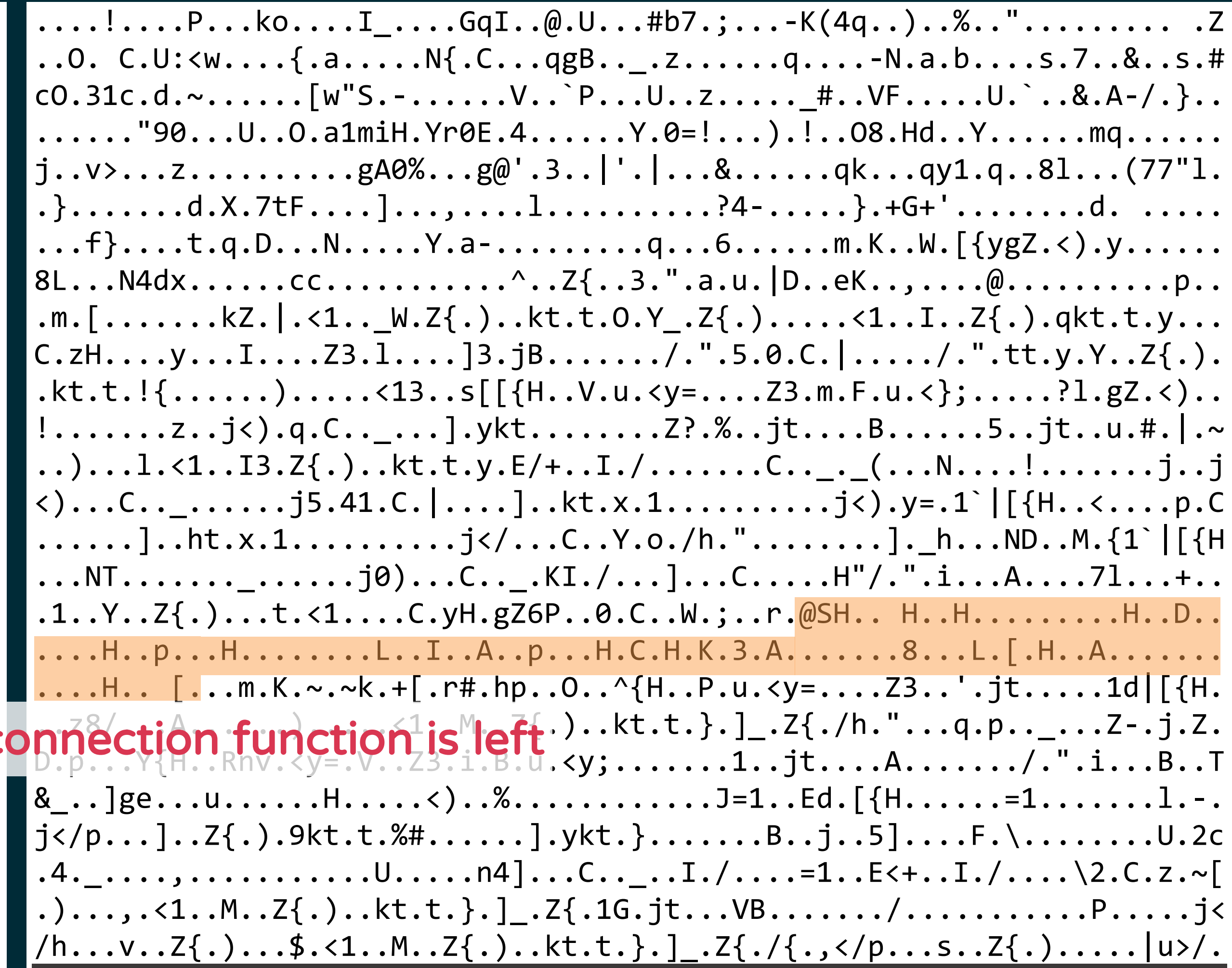
- Padding junk bytes to make the file size large
- Deliberately put the code of other malware families into the original malware for obfuscation
- Pack the malware files
- Self Modifying Code
    - self-modifying code is code that alters its own instructions while it is executing

Self-Modifying Code - Waterbear



```
H.\$.H.l$.H.t$ WATAUAVAWH..0.....XH..!.....H......H......QPH1......XI.
.u...I..C...M).L..PH..ATY.H1..........%h...X.(.eH.`.......PA\A..$=....
t.u..I......I....XYH..H...H...u...x...H..H.........E1.E1.H....N........
H..(...H.........E1.E1.H......|......E1.L......L..h...1.H......H.D$PH..
vH.........(...I..H........1.9+v3L.L$PL..(...A.M.L........I..$H..tY..I
...I...A..;+r...H......r..q...H....H....p...H..H..I..H........A..H..H.
...P...H....h...H.\$XH.l$`H.t$hH..0A_A^A]A\_.H.\$.WH.. 3.H..H..H;.t<D.
C.H.L$@.D$8..D$9..D$:..D$;........D.8H..0D.?.D.?0D..H...|.H.\$0H.. _.H
.\$.UVWH..@D......H......H......H........H........D......H......H.....
.......u.H.........H...H.........D......H..8...H.........H..H....8...D..
....H..8...H..................D......H..@...H.........L.L$`L.D$hH..H...
.@...D......H..@...H...............A...D......H......H........D.D$`H.T$
hH........D......H......H..H........H.........D.D$`H.L$h3.......D......
H.. ...H.........A....H..H.... ...D......H.. ...H..H........H..tv3.A..
...H.........D......H..0...H......L._.H.G.H.L$(H..L.\$ H.D$0H.|$8......
L.L$ L..H..H....0...D......H..0...H.........3.A.....H....8...H.L$h3.A..
.....8...H.........H.........H.\$p.....H..@_^].@SH.. H..H..........H..D..
....H..p...H.........L..I..A..p...H.C.H.K.3.A.......8...L.[.H..A........
....H.. [.H.\$.H.l$.H.t$.WATAUH..p...D......H..X...H......H..3.......D
.g.L..$P...A..H....X...D......H..X...H.......D.G.H.T$@E3.H..........
u.3..F...D......H......H........H.T$@H........D......H......H.........
|$D&r...l$DH.T$HE3.D.E.H...........t.D......H..`...H.........U.L..$P..
.H.L$E..`...D......H..`...H.........A.MZ..fD9\$E..A.........H.L$ M......
..........k..+.@...@...H.T.@.......D......H..X...H.........L..$`...H.L$
A....X...D......H..X...H............h...D......H..P.....H..i.<.........H
.T$@A.....H....P...D......H..P...H.........H.T$FH.L$0A.....M..D.d$@D.l$
```

Before self-modifying



```
....!....P...ko....I_....GqI..@.U...#b7.;...-K(4q..)..%.."......... .Z
..O. C.U:<w....{.a.....N{.C...qgB.._.z.......q....-N.a.b....s.7..&..s.#
cO.31c.d.~......[w"S.-.......V..`P...U..z....._#..VF.....U.`..&.A-/.}..
......"90...U..O.a1miH.Yr0E.4......Y.0=!...).!..O8.Hd..Y......mq......
j..v>...z...........gA0%...g@'.3..|'.|...&......qk...qy1.q..8l...(77"l.
.}.......d.X.7tF....]...,....l...........?4-.....}.+G+'.........d. .....
...f}....t.q.D...N.....Y.a-..........q...6......m.K..W.[{ygZ.<).y......
8L...N4dx......cc............^..Z{..3.".a.u.|D..eK..,....@..........p..
.m.[........kZ.|.<1.._W.Z{.)..kt.t.O.Y_.Z{.).....<1..I..Z{.).qkt.t.y...
C.zH....y...I....Z3.l....]3.jB......./.".5.0.C.|...../.".tt.y.Y..Z{.).
.kt.t.!{......).....<13..s[[{H..V.u.<y=....Z3.m.F.u.<};.....?l.gZ.<)..
!.......z..j<).q.C.._...].ykt........Z?.%..jt....B......5..jt..u.#.|.~
..)...l.<1..I3.Z{.)..kt.t.y.E/+..I./.......C.._._(...N....!.......j..j
<)...C.._......j5.41.C.|....]..kt.x.1..........j<).y=.1`|[{H..<....p.C
......]..ht.x.1..........j</...C..Y.o./h."........]._h...ND..M.{1`|[{H
...NT........_......j0)...C.._.KI./...]...C.....H"/.".i...A....7l...+..
.1..Y..Z{.)...t.<1....C.yH.gZ6P..0.C..W.;..r.@SH.. H..H..........H..D..
....H..p...H.........L..I..A..p...H.C.H.K.3.A.......8...L.[.H..A........
....H.. [...m.K.~.~k.+[.r#.hp..O..^{H..P.u.<y=....Z3..'.jt.....1d|[{H.
..z8/....A......)......<1..M..Z{.)..kt.t.}.]_.Z{./h."...q.p.._...Z-.j.Z.
D.p...Y{H..Rnv.<y=.V..Z3.i.B.u.<y;.......1..jt....A......./.".i...B..T
&_..]ge...u......H.....<)..%............J=1..Ed.[{H......=1........l.-.
j</p...]..Z{.).9kt.t.%#......].ykt.}........B..j..5]....F.\.........U.2c
.4._....,...........U.....n4]...C.._..I./....=1..E<+..I./....\2.C.z.~[
.)...,.<1..M..Z{.)..kt.t.}.]_.Z{.1G.jt...VB......./...........P.....j<
/h...v..Z{.)...$.<1..M..Z{.)..kt.t.}.]_.Z{./{.,</p...s..Z{.).....|u>/.
```

After self-modifying

Only the wait-for-connection function is left

Conclusion

- More and more advanced methods of process injection have been used
- Transfer Learning have great performance on memory-resident malware classification, especially on small set of data
- The features extract via Convolutional Network can find out the special area of malware
- We have also proposed some attackable methods for Adversarial attack

- https://github.com/AragornTseng/Mem2Img black hat ASIA 2021

THANK YOU!

aragorn@teamt5.org
charles@teamt5.org

TEAMT5
杜 浦 數 位 安 全
Persistent Cyber Threat Hunters