Malware analysis using visualized images and entropy graphs

Kyoung Soo Han · Jae Hyun Lim · Boojoong Kang · Eul Gyu Im

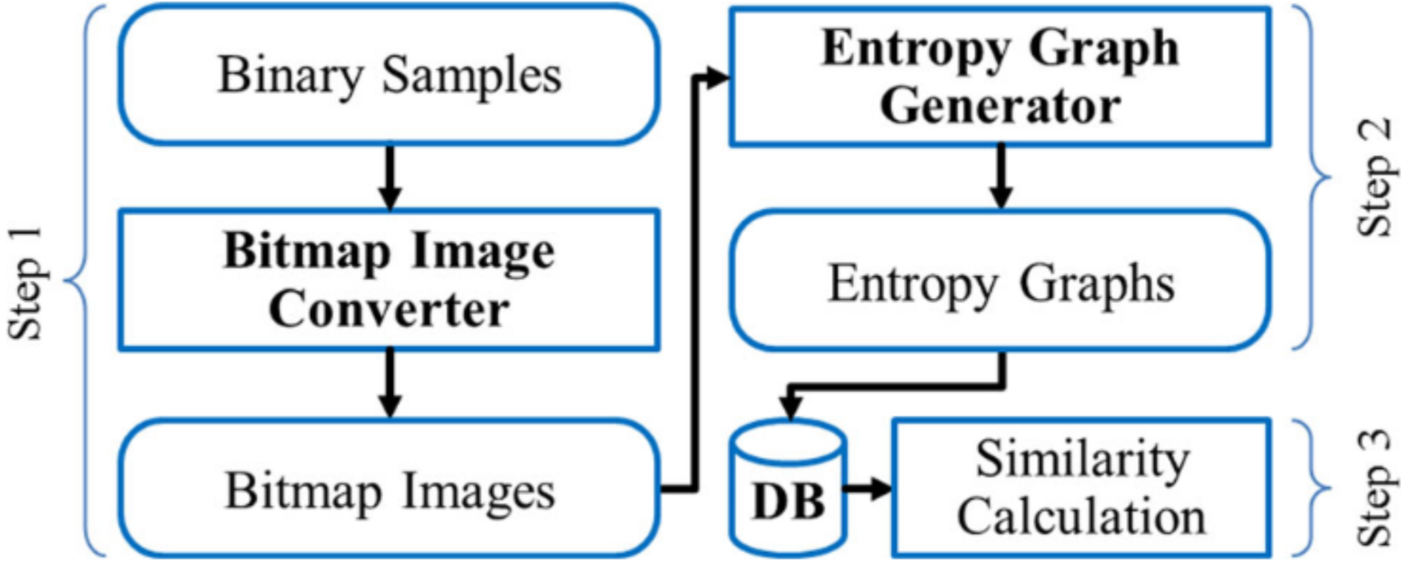
Presented by Ruikai Zheng

CISC850
Cyber Analytics

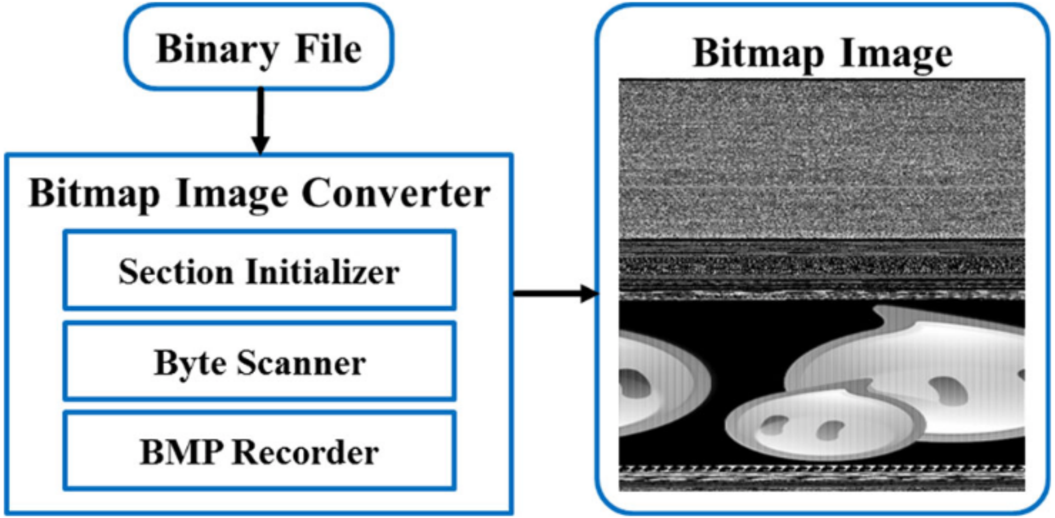
1.Introduction

- Malware variants developed using automated tools
- Automated tools reuse modules
- Similarities may exist among malware variants

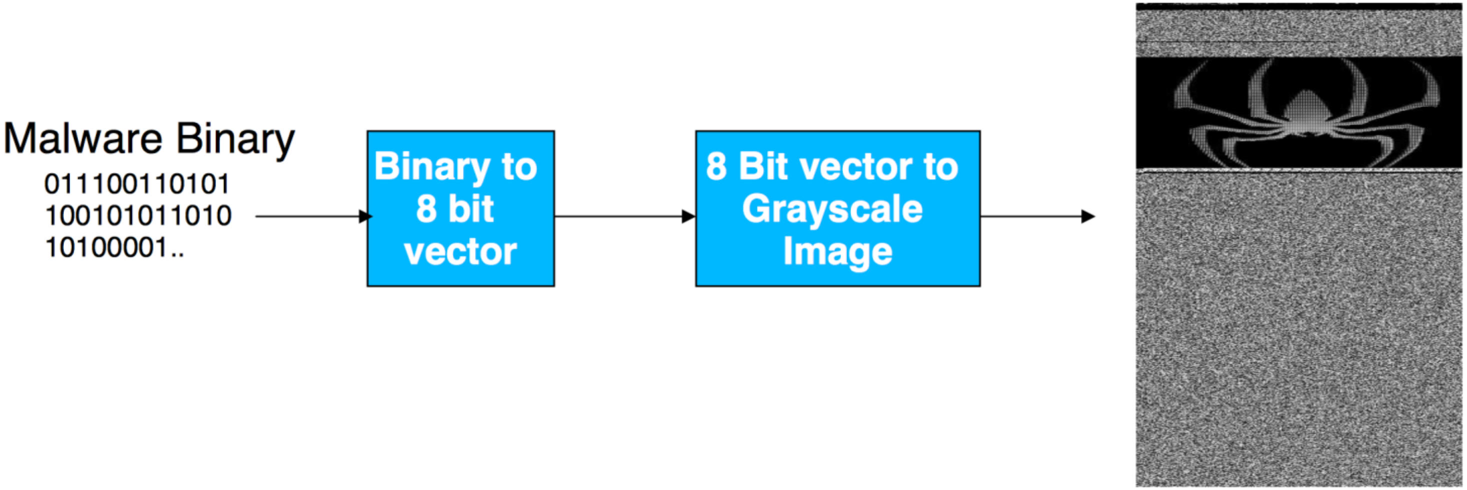
2.General Idea



3. Bitmap Image



Bitmap Image converter



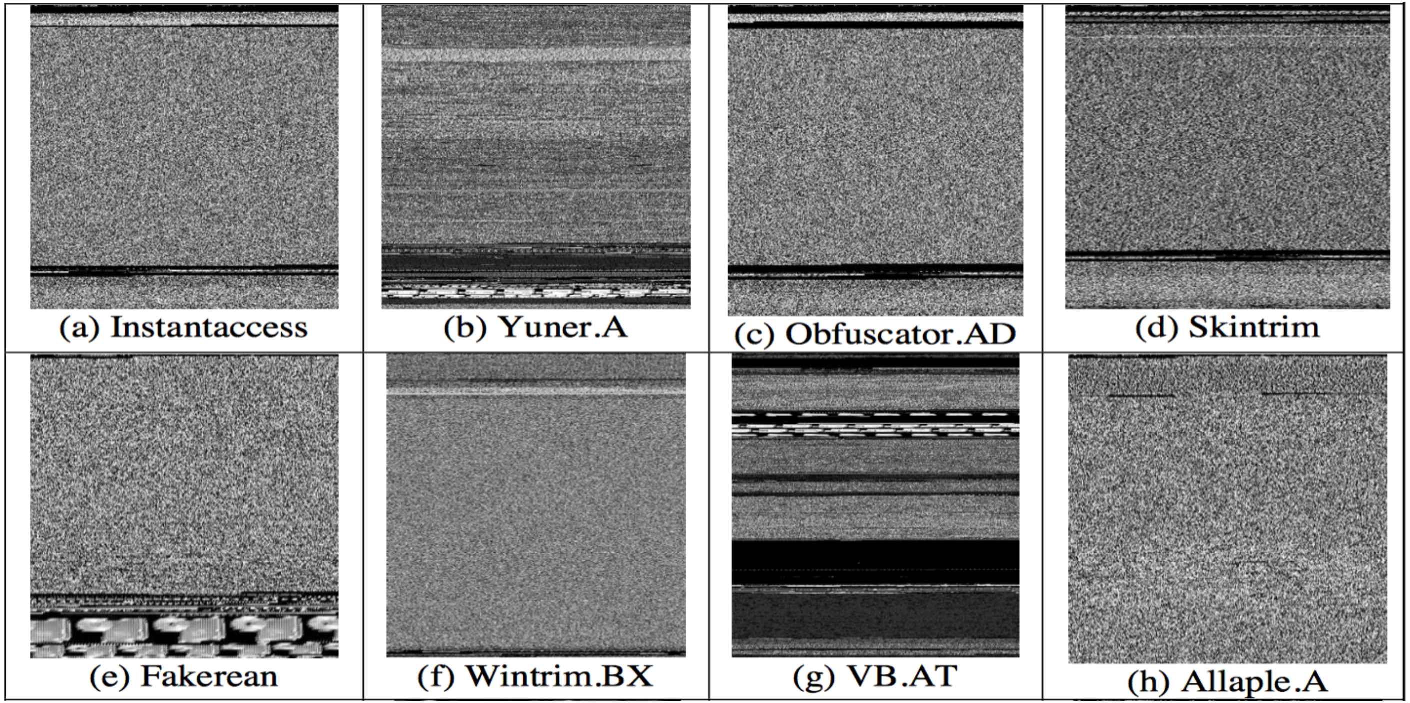
Malware Binary

011100110101
100101011010
10100001..

Binary to 8 bit vector

8 Bit vector to Grayscale Image

Some examples



(a) Instantaccess

(b) Yuner.A

(c) Obfuscator.AD

(d) Skintrim

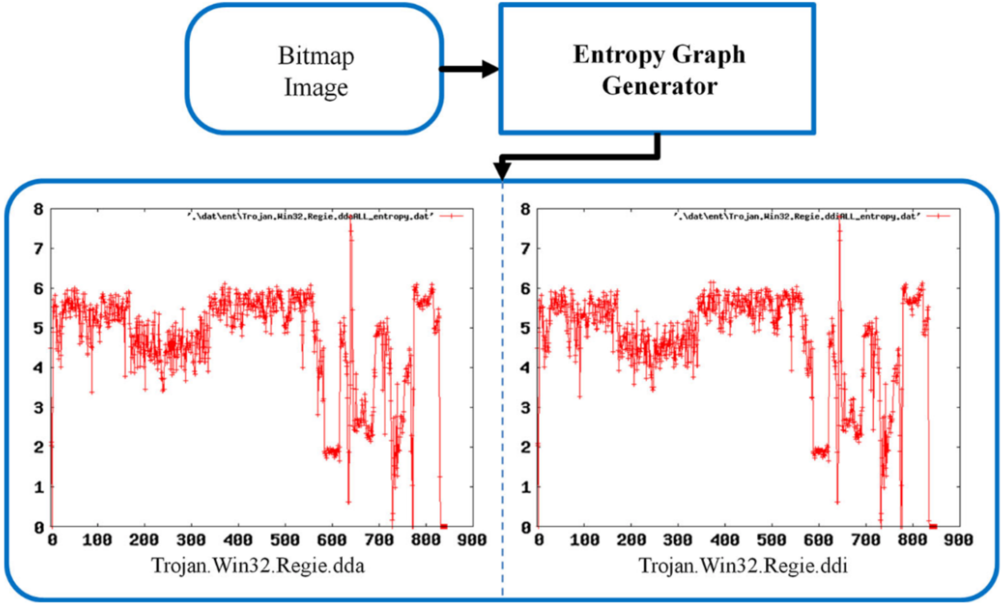
(e) Fakerean

(f) Wintrim.BX

(g) VB.AT

(h) Allaple.A

4. Entropy graph



Bitmap Image

Entropy Graph Generator

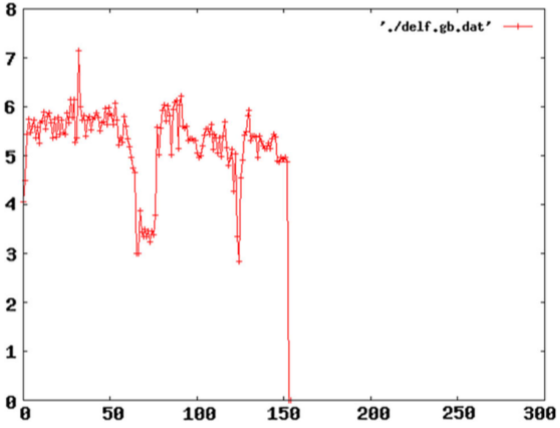
Entropy graph generator

For each line of bitmap image:
(suppose the image is 256 * 256)

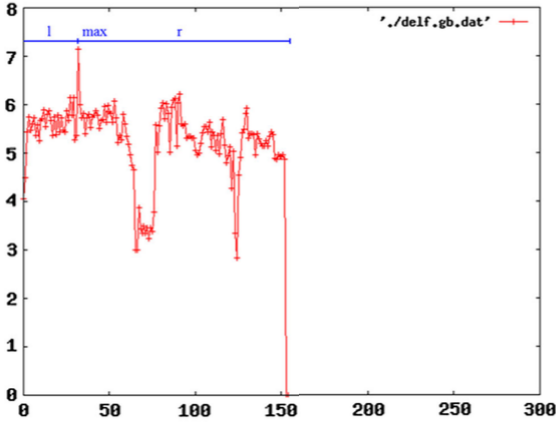
$$Entropy = -\sum_{i=0}^{255} p_i \times \log_2 p_i$$

5. Compute similarities

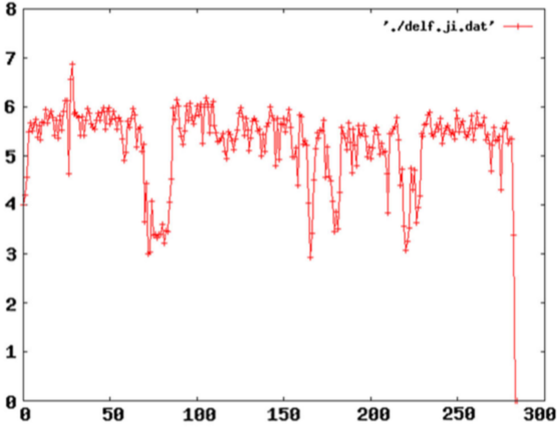
- Align the x-axes(the heights of bitmap images) of the two entropy graphs



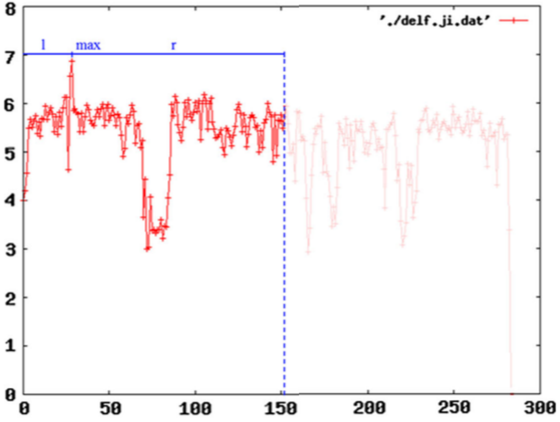
(a) Shorter histogram *H*



(a) Shorter histogram *H*



(b) Longer histogram *L*



(b) Longer histogram *L*

Compute similarities

- Compute K_1 and K_2
  - K_1

$$k_1[H, L] = \exp\left(-\frac{s[H, L]}{\bar{s}}\right)$$

where

$$s[H, L] = \int_x |H(x) - L(x)|\, dx$$

$$\bar{s} = Average(H) \times Length \text{ of } y \text{ axis}$$

Compute similarities

- Compute K_1 and K_2
  - K_2

$$k_2[H, L] = \sum_i u_i[H] c_i[H, L]$$

where

$$u_i = \frac{\left|H^{(2)}(x_i)\right| \times l_i}{\sum_{i=1}^{n(H)} \left|H^{(2)}(x_i)\right| \times l_i}$$

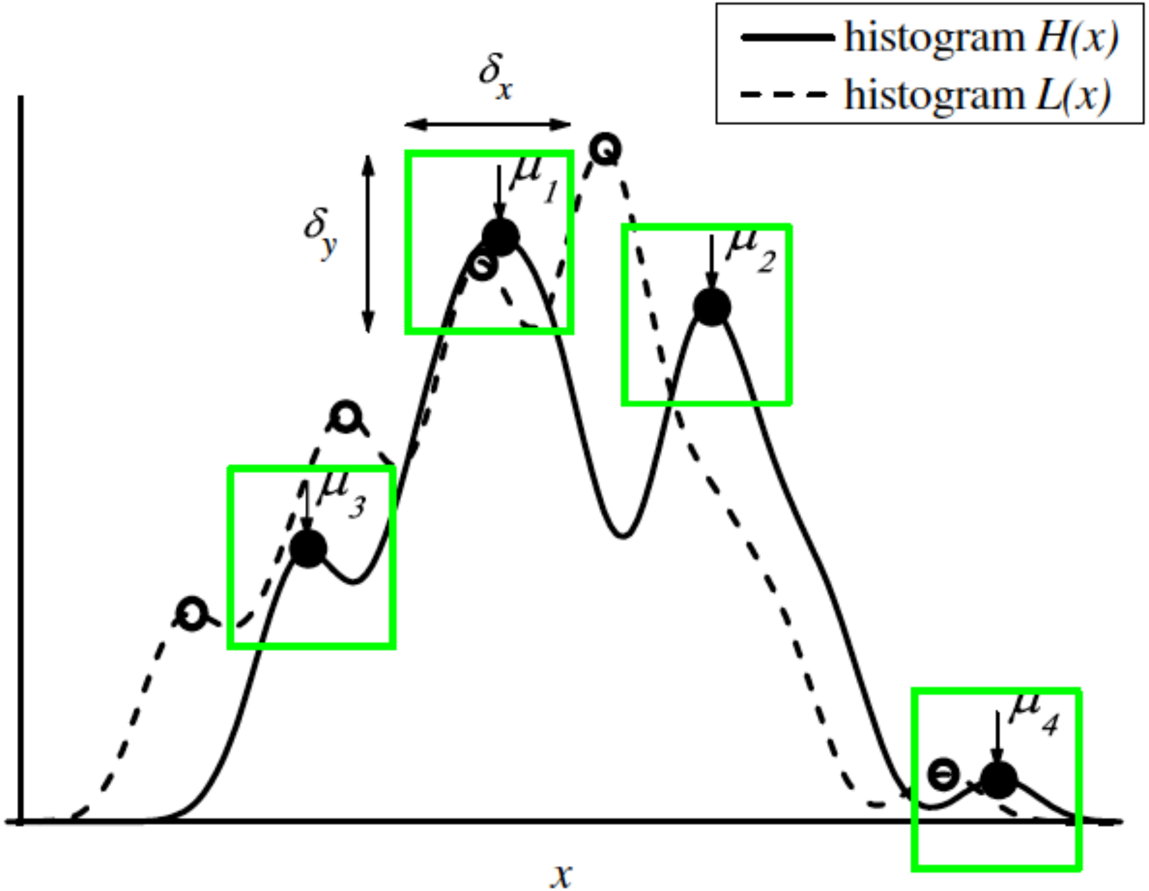
$$c_{i,j} = c^x_{i,j} \times c^y_{i,j}$$

$$c^x_{i,j} = -\exp\left(\frac{\Delta x_{i,j}}{\delta x}\right)^2$$

$$\Delta x_{i,j} = x_i - \tilde{x}_j$$

$$c^y_{i,j} = -\exp\left(\frac{\Delta y_{i,j}}{\delta y}\right)^2$$

$$\Delta y_{i,j} = H(x_i) - L(\tilde{x}_j)$$

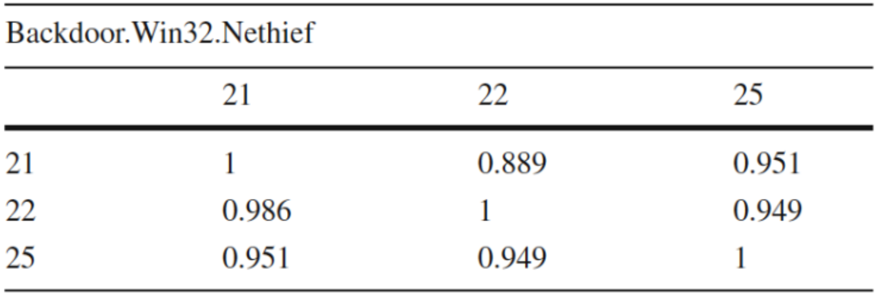

histogram $H(x)$
histogram $L(x)$
δ_x
δ_y
μ_1
μ_2
μ_3
μ_4
x

Compute similarities

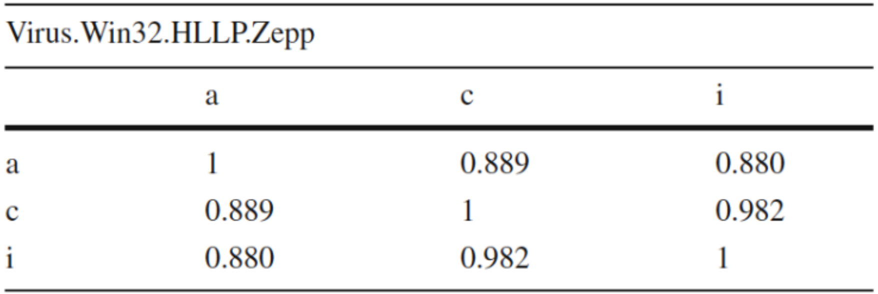
- Similarity value

$$S = t_1 \times k_1 + t_2 \times k_2$$

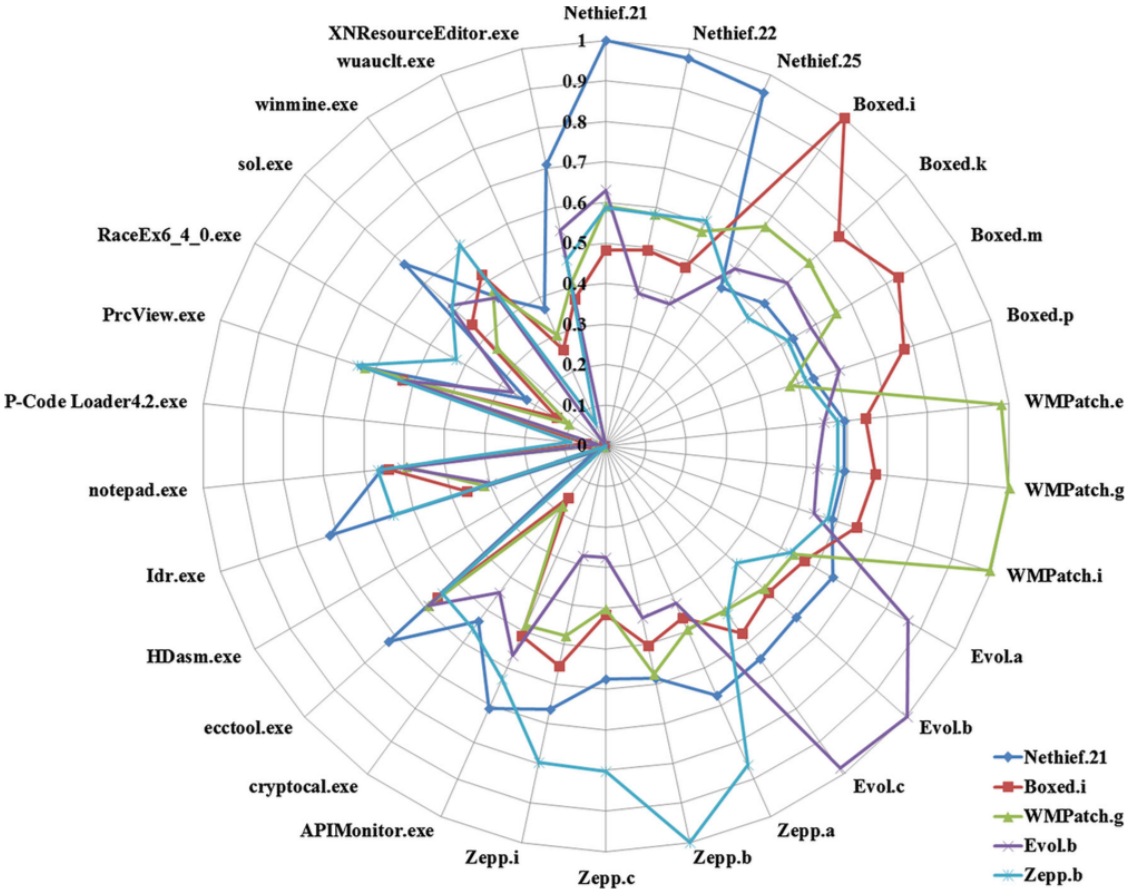
Experiment result



Backdoor.Win32.Nethief			
	21	22	25
21	1	0.889	0.951
22	0.986	1	0.949
25	0.951	0.949	1

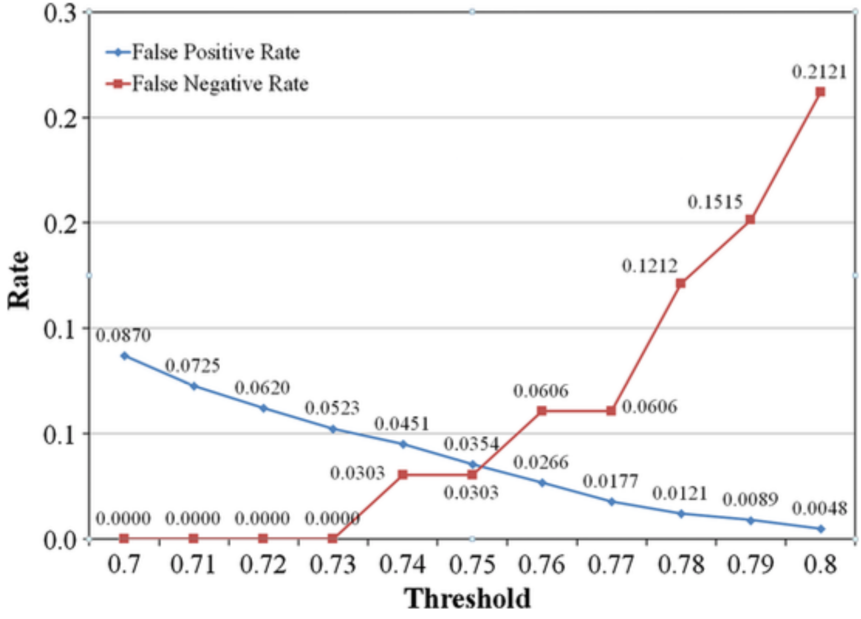


Virus.Win32.HLLP.Zepp			
	a	c	i
a	1	0.889	0.880
c	0.889	1	0.982
i	0.880	0.982	1


Nethief.21
Nethief.22
Nethief.25
Boxed.i
Boxed.k
Boxed.m
Boxed.p
WMPatch.e
WMPatch.g
WMPatch.i
Evol.a
Evol.b
Evol.c
Zepp.a
Zepp.b
Zepp.c
Zepp.i
APIMonitor.exe
cryptocal.exe
ecctool.exe
HDasm.exe
Idr.exe
notepad.exe
P-Code Loader4.2.exe
PrcView.exe
RaceEx6_4_0.exe
sol.exe
winmine.exe
wuauclt.exe
XNResourceEditor.exe
1
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0
Nethief.21
Boxed.i
WMPatch.g
Evol.b
Zepp.b

Experiment result

- Threshold
  - False positive rate
  - False negative rate



Limitation

- Malware applied with packing technique
  - The entropy values of binaries can be very high
  - Packed malware binaries are difficult to classify

Conclusion

- The paper proposed a malware visualization method that using binary grayscale bitmap images and entropy graphs.
- The paper proposed a method to calculate similarities of malware to classify malware families.
- Experimental results showed that proposed method can classify malware families with a small false-positive/false -negative rate.

Thank you