IoCannon

Marion Marschalek
@pinkflawd

Eireann Leverett
@blackswanburst

„There is hardly anything in the world that someone cannot make a little worse and sell a little cheaper“

– J. Ruskin

Network IoC and Attacker Costs

IoC	Buy	Rent
IPv4	~$ 14	~$2
IPv6	~$250	~$2
Domain	$1	$1
Bitcoin account	$0	N/A
SSL/TLS Certificates	$0-5	N/A

Code & Binaries
take time
not money...

Logistical Burden

Measure what you can:

- How many binaries in Sofacy campaign over time?

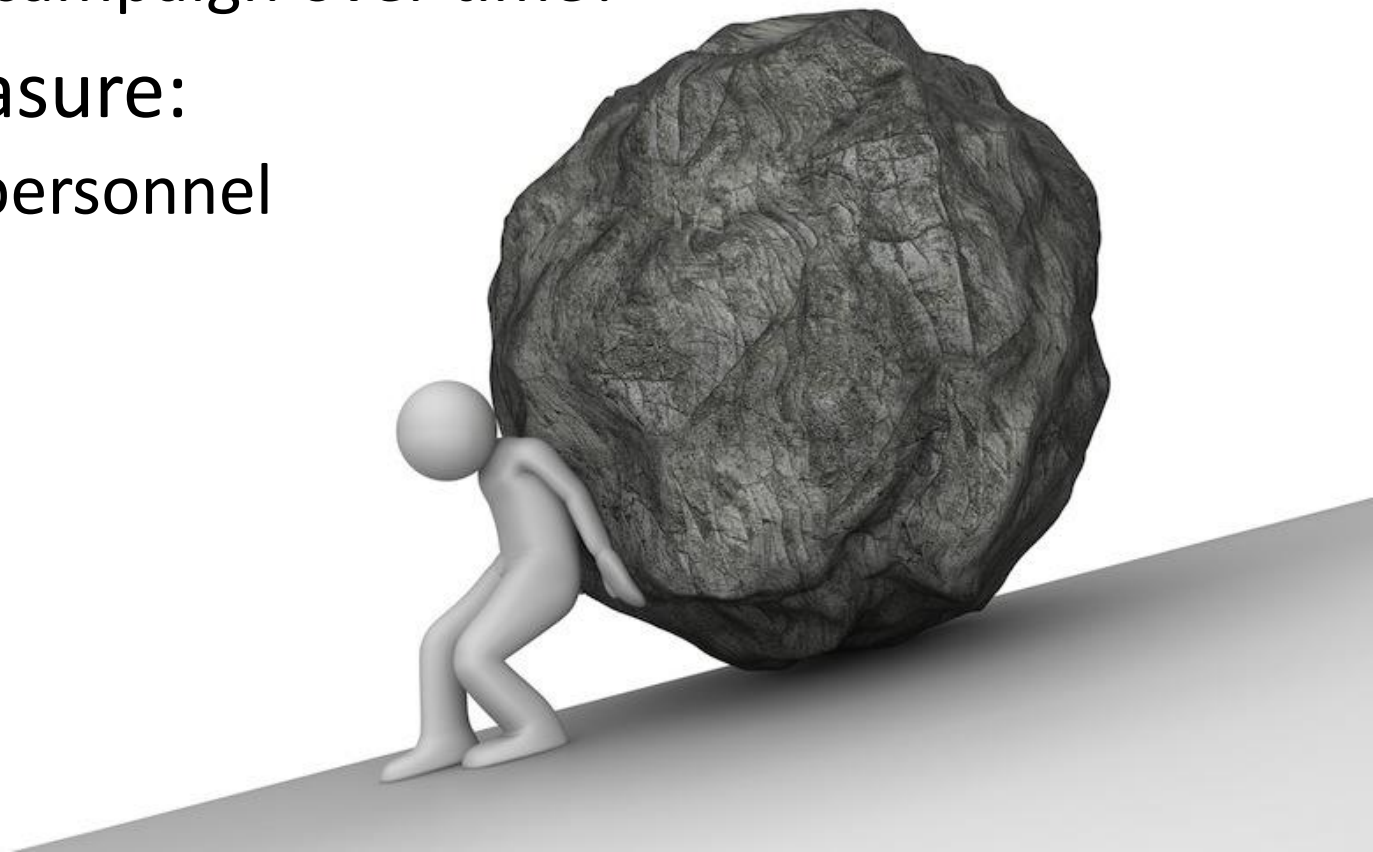
Estimate what you can´t measure:

- Software development cost, personnel

Stealth cost

- Hiding is tedious

Operations < Infrastructure



Cost of Attack

Time

Skills

Tools

World class exploit developer $250k/year

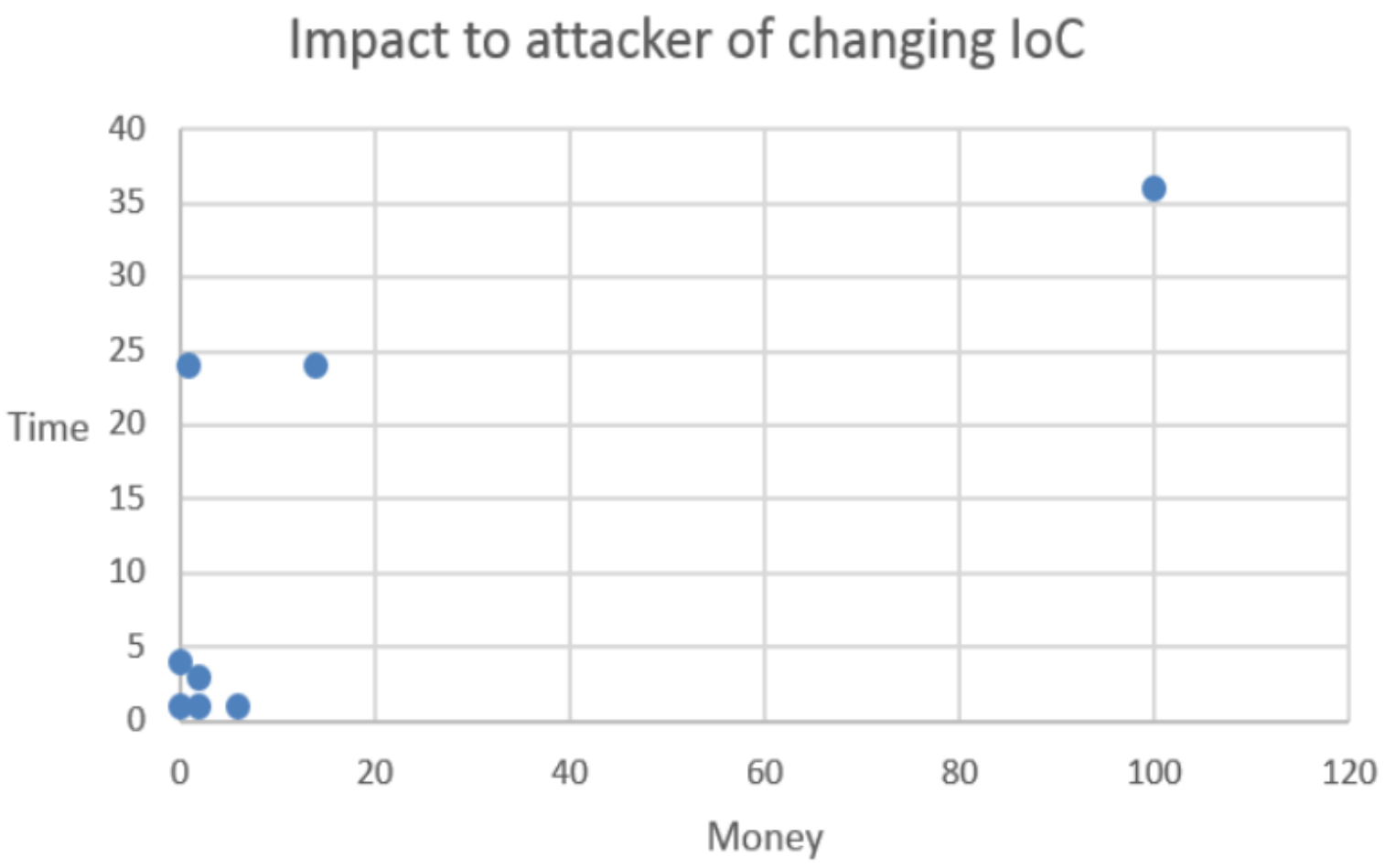
Estimating cyber war costs: $45.9 Mio/year

Estimated costs for development of Stuxnet: $500k

APT1 group thoroughly exposed in Feb. 2013, back in business 160 days after exposure

Attacker Infrastructure Change Cost

IoC	Money (USD)	Time (Hours)
IPv4	2	1
Ipv6	6	1
v4Netblock	14	24
v6NetBlock	1	24
Domain	2	3
c2	100	36
Hash	0	1
ImpHash	0	4



Moarrrr IoC

We can‘t find the needle in the haystack,
give us more hay!

Threat Detection: The hay that we got

File hashes	System behavior	Known-bad
File fragments	Network patterns	Non known-good
File behavior	Abnormal system behavior	Known-bad origin
File properties	Abnormal network patterns	Non known-good origin

Threat detection metrics heavily **build on known fragments**, while aiming to **find the largely unknown**.

The problem we got with the hay

The time blind spot

The geographical blind spot

The data quality blind spot

Historical data is gold

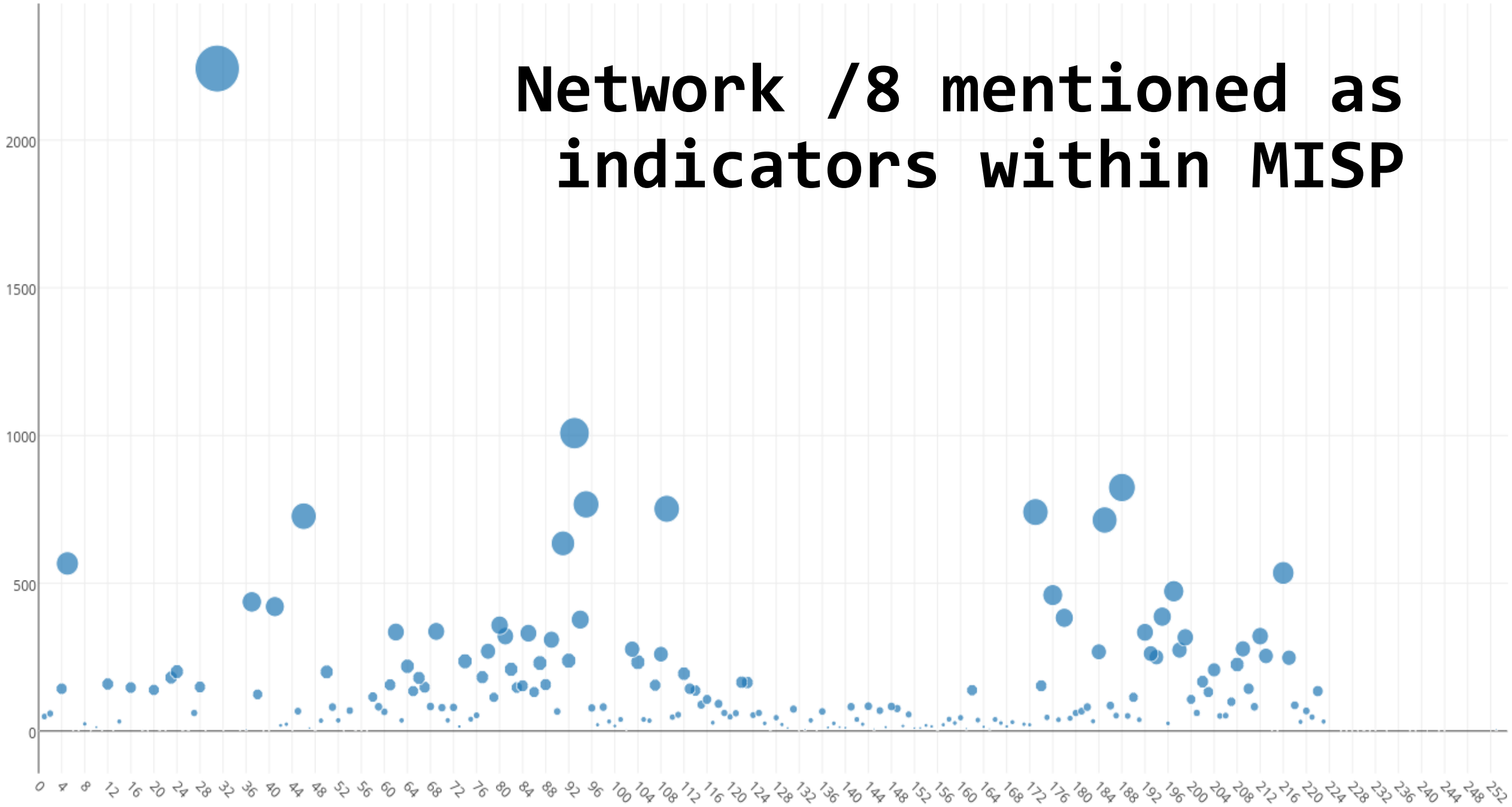
Sensors in China deliver TI for China

Can’t compare a file hash to a domain name, can you?

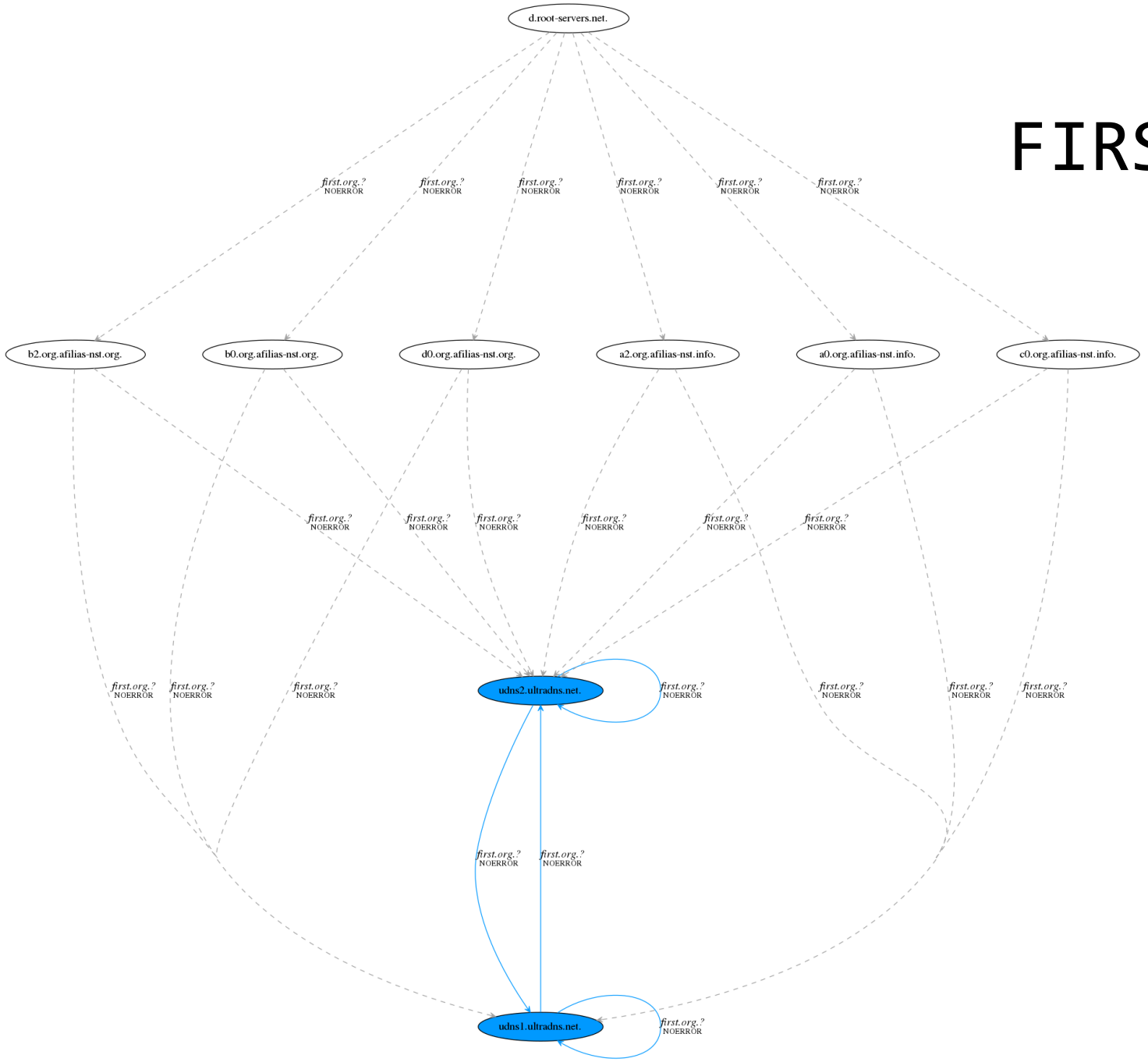
shared ground truth
for engineering
and extracting
comparable features

FEATURE FACTORIES

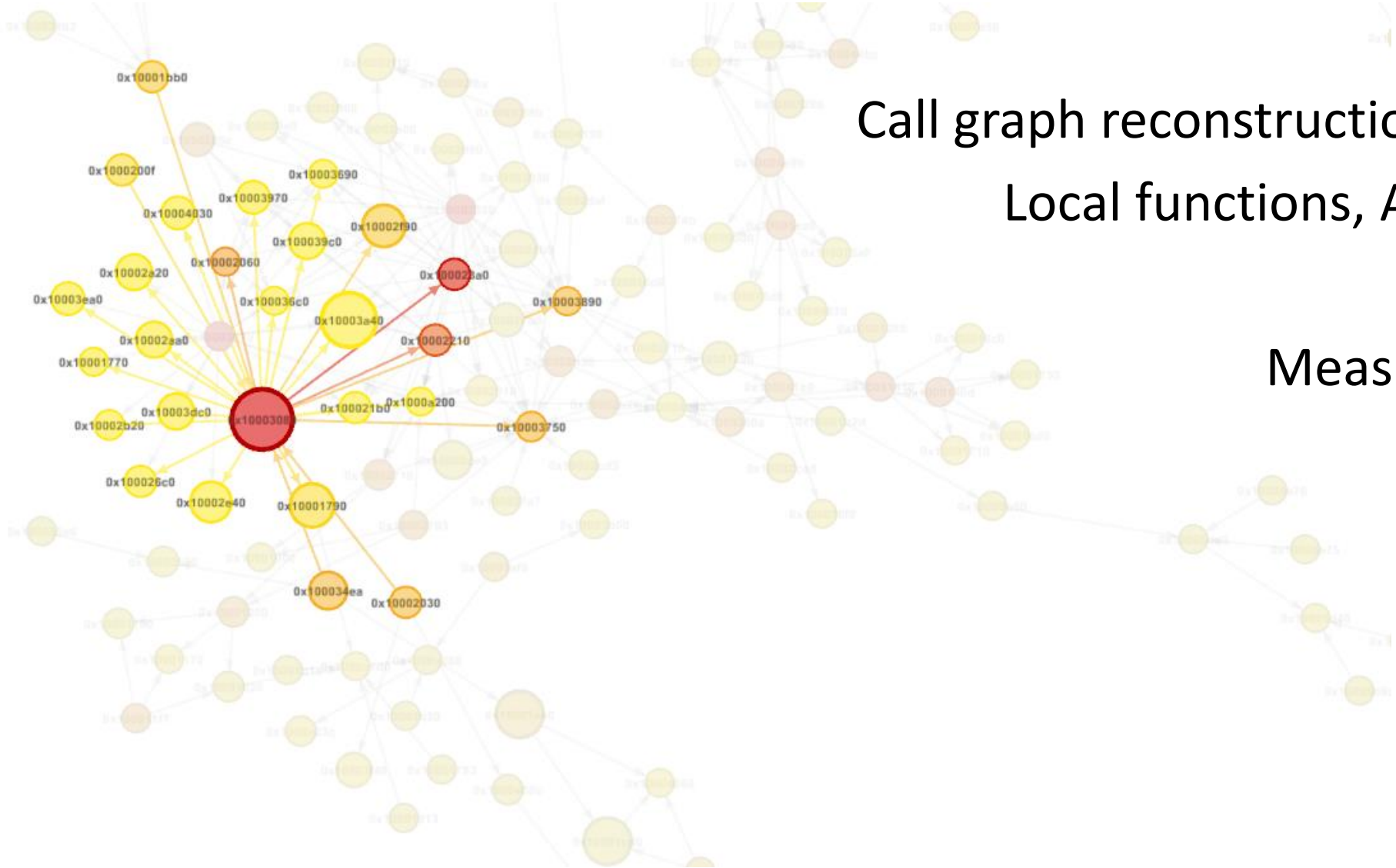
Network /8 mentioned as indicators within MISP



FIRST.org Trust Tree



Call Graph Layout



Call graph reconstruction with radare2

Local functions, API calls, strings

Measuring the graph

functions

edges

calls

ratios

sizes

paths

API Calls

Interface between software and operating system

Windows executables w/o API calls highly unlikely

Documented - mostly..

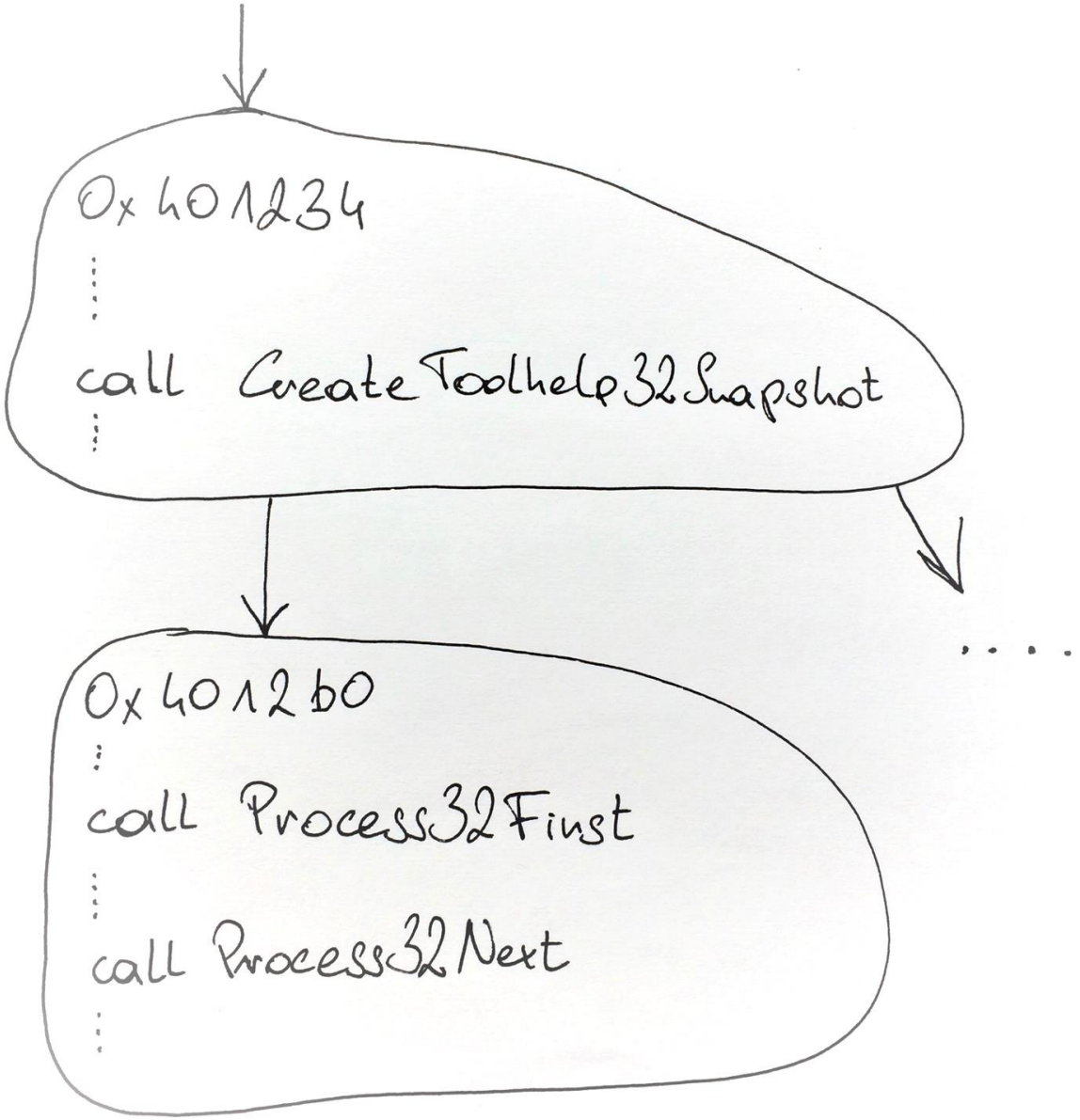
Frequently „hidden“ within malware

```
[0x004344b6]> axt @@ sym.*
data 0x40e552 mov ebp, dword [sym.imp.KERNEL32.dll_LoadLibraryA] in fcn.00402db0
data 0x40e558 mov ebx, dword [sym.imp.KERNEL32.dll_GetProcAddress] in fcn.00402db0
call 0x4345de call dword [sym.imp.KERNEL32.dll_GetModuleHandleA] in entry0
data 0x4345de call dword [sym.imp.KERNEL32.dll_GetModuleHandleA] in entry0
call 0x4345ba call dword [sym.imp.KERNEL32.dll_GetStartupInfoA] in entry0
data 0x4345ba call dword [sym.imp.KERNEL32.dll_GetStartupInfoA] in entry0
call 0x401c3f call dword [sym.imp.GDI32.dll_RealizePalette] in fcn.00401040
data 0x401c3f call dword [sym.imp.GDI32.dll_RealizePalette] in fcn.00401040
call 0x401b5b call dword [sym.imp.GDI32.dll_CreateDIBSection] in fcn.00401040
call 0x401bd6 call dword [sym.imp.GDI32.dll_CreateDIBSection] in fcn.00401040
data 0x401b5b call dword [sym.imp.GDI32.dll_CreateDIBSection] in fcn.00401040
data 0x401bd6 call dword [sym.imp.GDI32.dll_CreateDIBSection] in fcn.00401040
call 0x401b6b call dword [sym.imp.GDI32.dll_IntersectClipRect] in fcn.00401040
data 0x401b6b call dword [sym.imp.GDI32.dll_IntersectClipRect] in fcn.00401040
call 0x401c5d call dword [sym.imp.GDI32.dll_CreateRectRgn] in fcn.00401040
data 0x401c5d call dword [sym.imp.GDI32.dll_CreateRectRgn] in fcn.00401040
call 0x401c4f call dword [sym.imp.GDI32.dll_GetBkMode] in fcn.00401040
data 0x401c4f call dword [sym.imp.GDI32.dll_GetBkMode] in fcn.00401040
call 0x401c47 call dword [sym.imp.GDI32.dll_CreateCompatibleDC] in fcn.00401040
data 0x401c47 call dword [sym.imp.GDI32.dll_CreateCompatibleDC] in fcn.00401040
data 0x401c2d mov esi, dword [sym.imp.GDI32.dll_SetPaletteEntries] in fcn.00401040
call 0x401c27 call dword [sym.imp.GDI32.dll_GetClipBox] in fcn.00401040
```

Scanning for Gadgets

Singular gadgets to count
e.g. API resolution, memory allocations

Pre-defined API patterns
Searching the graph for anchor
Scanning nodes in close vicinity

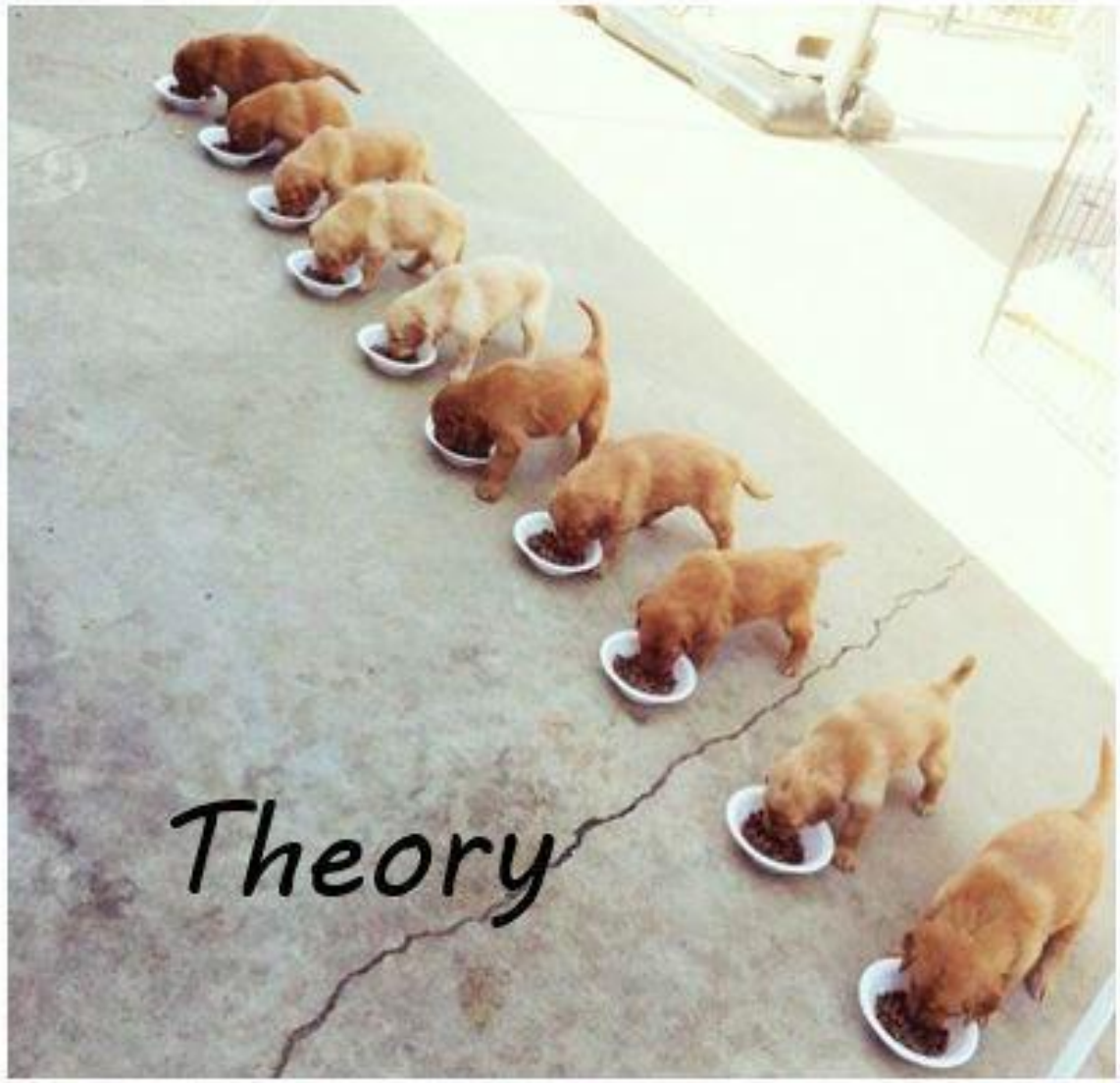


Behaviorgadgets: An IRC-Bot

```
For SEND found {'send': '0x40128a'}
For SEND found {'send': '0x401ad5'}
For SEND found {'send': '0x402315'}
For SEND found {'send': '0x4016d2'}
For CREATEPROC found {'CreateProcess': '0x402aa1'}
For EXITSYSTEM found {'ExitWindows': '0x402aa1'}
For CREATETHREAD found {'CreateThread': '0x402aa1'}
For APILOADING found {'GetProcAddress': '0x407313'}
For RECV found {'recv': '0x402230'}
For RECV found {'recv': '0x40198f'}
For REGSETVAL found {'RegOpenKey': '0x402670', 'RegSetValue': '0x402670'}
```

Which gadgets, how often => matrix

Multithreaded programming

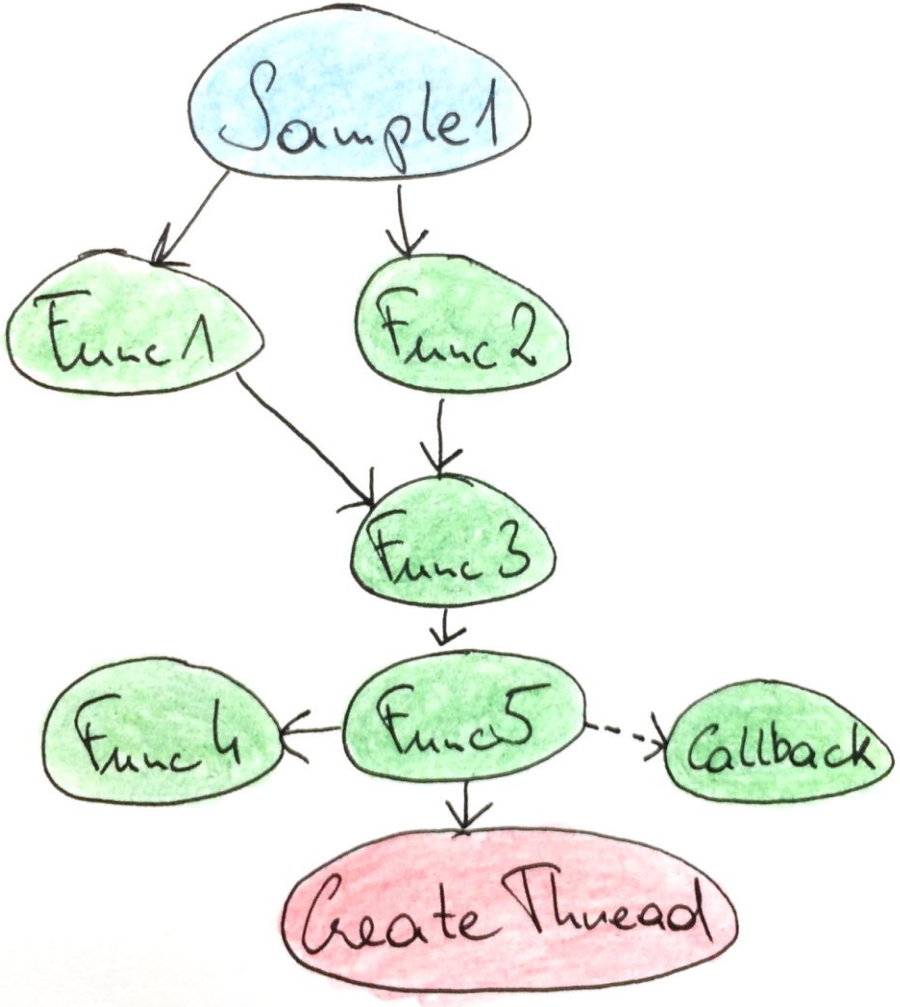


```
55              push ebp
8bec            mov ebp, esp
6a00            push 0
6a00            push 0
8b4508          mov eax, dword [ebp + arg_8h] ; [0x8:4]=4
50              push eax
6a00            push 0
ff150c914000    call dword [sym.imp.USER32.dll_MessageBoxA]
33c0            xor eax, eax
5d              pop ebp
c3              ret
```

```
0x00403046      83c209          add edx, 9
0x00403049      52              push edx
0x0040304a      68e225e2        push fcn.004025e2 ; fcn.004025e2 ; "U...." @ 0x
0x0040304f      6a00            push 0
0x00403051      6a00            push 0
0x00403053      ff1528904000    call dword [sym.imp.KERNEL32.DLL_CreateThread]
0x00403059      e9db000000      jmp 0x403139
    ; JMP XREF from 0x0040303d (fcn.00402aa1)
0x0040305e      6a07            push 7
0x00403060      68d0a54000      push str.clswnd ; str.clswnd ; "clswnd " @ 0x40
```

Thread Model Modelling

Number of calls to CreateThread
Shortest path to CreateThread
Number of handler functions
Average size of handler functions
Size of biggest handler function



String Constants

Human readable strings give information away

Presence or absence of readable strings is relevant information

Graph structure, character frequency and character repetition allow string constant evaluation

```
freqs = {
    'a': 0.0651738,
    'b': 0.0124248,
    'c': 0.0217339,
    'd': 0.0349835,
    'e': 0.1041442,
    'f': 0.0197881,
    'g': 0.0158610,
    'h': 0.0492888,
    'i': 0.0558094,
    'j': 0.0109033,
    'k': 0.0150529,
    'l': 0.0331490,
    'm': 0.0202124,
    'n': 0.0564513,
    'o': 0.0596302,
    'p': 0.0137645,
    'q': 0.0058606,
    'r': 0.0497563,
    's': 0.0515760,
    't': 0.0729357,
    'u': 0.0225134,
    'v': 0.0182903,
    'w': 0.0271272,
    'x': 0.0013692,
    'y': 0.0145984,
    'z': 0.0017836,
    ' ': 0.0500000,
    '0': 0.0500000,
    '1': 0.0500000,
    '2': 0.0500000,
    '3': 0.0500000,
    '4': 0.0500000,
    '5': 0.0500000,
    '6': 0.0500000,
    '7': 0.0500000,
    '8': 0.0500000,
    '9': 0.0500000,
    '.': 0.0400000,
    '_': 0.0400000
}
```

lnkfile \\ shellex \\ IconHandler	0.08975369696969697
OptionFlags	0.0457972
Progman	0.04012135714285714 6
_^Ã‹L$	0.01462979999999998
<A| \\ b<Z	0.01793821999999998
<A| \\ b<Z	0.01793821999999998
0123456789abcdefghijklmnopqrstuvwxyzABC	0.0702613625
_^][	0.01
_^][	0.01
SUVW	0.029876725
\\ *.*	0.02
X_^[	0.0103423
\\ StringFileInfo \\ %s \\ FileVersion	0.08549147692307693
%08X	0.0253423
\\ VarFileInfo \\ Translation	0.09178884
_^Ã‹L$	0.01462979999999998
SHELL32.DLL	0.04659895454545454534
SHGetFolderLocation	0.1073442631578947 3
State	0.07335308
_^][	0.01
3É,`|	0.0125
3É,`|	0.0125
3É,`|	0.0125

String character frequency histogram per sample

Bucketsize of 0.01

Count of strings per bucket

0.04 is a reasonable edge

Resilient to little changes

Subset of Sofacy

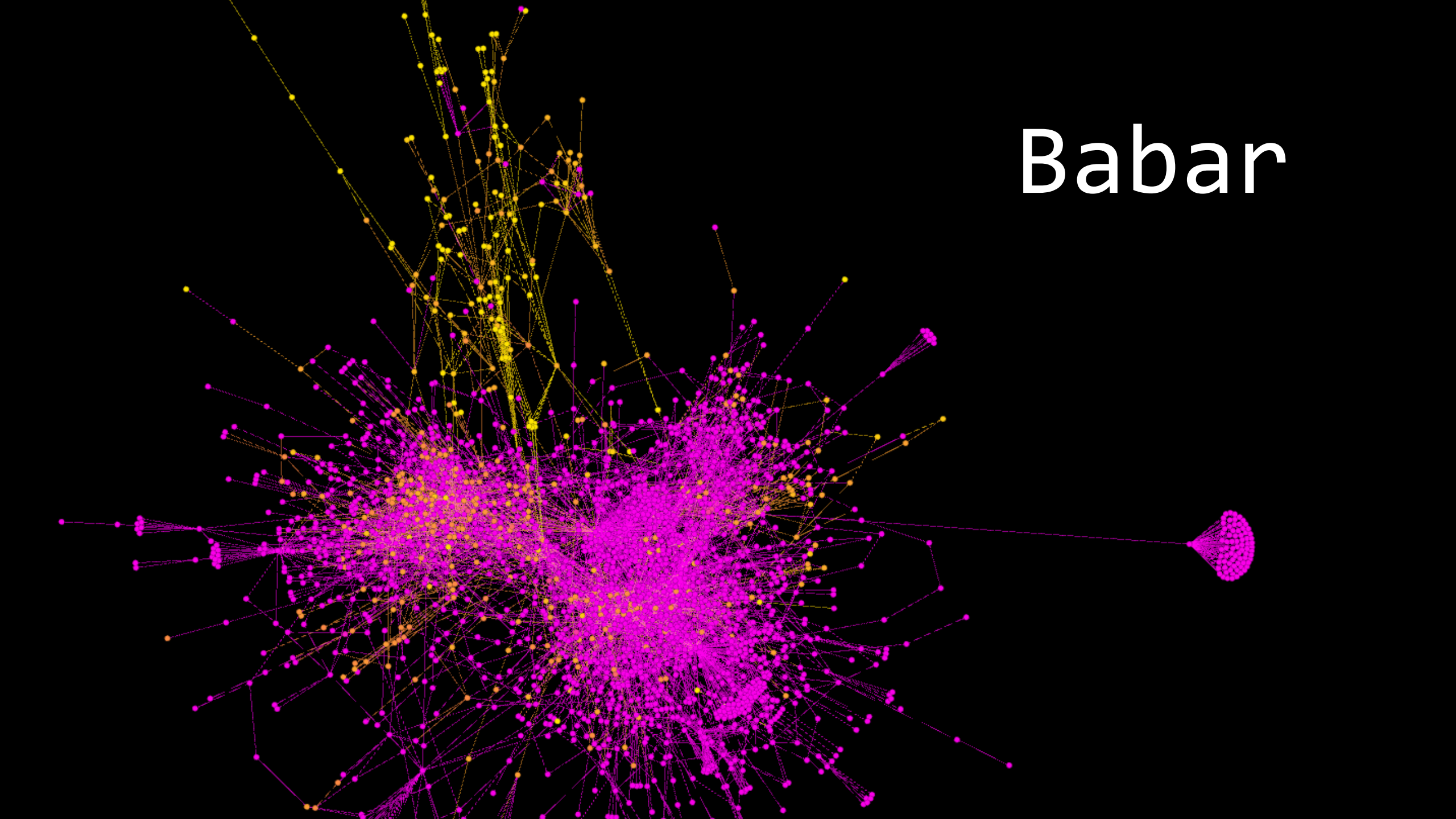
2-0-7-9-31-0-0-3-30
2-2-7-12-37-1-0-4-38
2-8-8-11-39-1-0-4-38
2-4-7-13-37-5-0-3-34
3-5-7-16-40-6-0-4-38
2-5-7-14-36-5-0-3-38
3-6-7-12-35-4-0-3-30
2-4-7-13-29-5-0-3-29
2-4-7-7-27-0-0-3-29
3-4-7-10-27-0-0-3-29
3-4-7-12-27-4-0-3-29
13-233-274-464-276-1381-1895-265-190
13-233-274-464-276-1381-1895-265-190
2-2-5-11-25-1-0-4-46
2-2-5-11-25-1-0-4-46
2-2-5-11-25-1-0-4-46
2-2-5-11-25-1-0-4-46
2-2-5-11-25-1-0-4-46
3-0-3-8-13-0-1-3-2
3-1-3-8-13-0-1-3-2
3-1-3-8-13-0-1-3-2
12-195-121-175-177-769-1319-75-49
12-195-122-175-177-784-1324-76-50
12-194-123-163-184-786-1308-81-49
12-195-120-156-188-781-1308-76-47
12-195-121-158-163-785-1323-73-43
12-195-122-157-187-770-1255-76-48
12-195-123-156-183-769-1324-73-49
9-193-101-134-160-757-1277-76-48
12-195-121-160-189-786-1304-81-49

Node‘s mnemonic distribution

Arithmetic instructions as indicator for cryptography, compression or codecs

Leveraging radare2‘s instruction type

```
shl
shr
mul
div
rol
ror
sar
load
store
```


Babar

Feasibility

Files >2mb problematic

Radare out-of-the-box PE analysis needs overhaul

Parsing time <1s – ~1h

Still on POC stage

Parse once, derive features foreva

Fully open-source, scales well

functionstotal	refslocal	refsglobalvar	refsunknown	apitotal	apimisses	stringsreferenced	stringsdangling	stringsnoref	ratiofunc	ratioapi	ratiostring
2675	13282	22	4	2443	20	562	0	18069	5.097179878048781	4.655106707317073	1.0708841463414636
2675	13282	22	4	2443	20	562	0	31476	5.097179878048781	4.655106707317073	1.0708841463414636
2675	13282	22	4	2443	20	562	0	18652	5.097179878048781	4.655106707317073	1.0708841463414636
2675	13282	22	4	2443	20	562	0	19025	5.097179878048781	4.655106707317073	1.0708841463414636
2618	12995	37	1	2540	35	536	12	15504	4.974008998054474	4.825814688715953	1.0183608949416343
2613	12988	37	1	2540	36	536	12	17475	4.964509362840467	4.825814688715953	1.0183608949416343
2613	12988	37	1	2540	36	536	12	40555	4.964509362840467	4.825814688715953	1.0183608949416343
2613	12988	37	1	2540	36	536	12	17207	4.964509362840467	4.825814688715953	1.0183608949416343
2347	11991	23	4	2432	165	662	24	16962	4.6256149091826435	4.79313824419778	1.3047111503531785
223	339	0	2	114	0	21	1	9939	1.2373490767045456	0.6325461647727273	0.11652166193181819
32	1	0	0	0	0	0	0	4240	0.1151012891344383	0.0	0.0
25	1	0	0	23	0	1	0	7642	0.11967677696078433	0.11010263480392157	0.004787071078431373
20	1	0	0	16	0	1	0	4561	0.8138020833333334	0.6510416666666666	0.040690104166666664
18	2	0	0	20	3	1	0	5569	0.54931640625	0.6103515625	0.030517578125
12	8	0	0	0	0	0	0	8022	0.07659313725490197	0.0	0.0
9	2	0	0	23	13	1	0	5854	0.732421875	1.8717447916666665	0.08138020833333333
6	0	0	0	0	0	0	0	14900	0.02215264650283554	0.0	0.0
2	0	0	0	0	0	0	0	18707	0.007384215500945179	0.0	0.0
0	0	0	0	0	0	0	0	1238	0.0	0.0	0.0
0	0	0	0	0	0	0	0	1234	0.0	0.0	0.0
0	0	0	0	0	0	0	0	1875	0.0	0.0	0.0

PackRat

And the packers?

DATA

Testdata for APT28, Packrat, Animalfarm and others soon to be released on github

	AE	AF	AG	AH	AI	AJ	AK	AL	AM	AN	AO	AP	AQ	AR	AS	AT	AU	AV	AW	AX
1	functionstotal	refslocal	refsglobalva	refsunknown	apitotal	apimisses	stringsreferenced	stringsdangling	stringsnoref	ratiofunc	ratioapi	ratiostring	getprocaddress	memallocation	createthread	ctshortestpath	callbackcount	cbaveragesize	cblargestsize	stringsrefhisto
2	124	715	1	0	183	3	30	2	264	9.6875	14.296875	2.34375	0	88	3	2	2	467	612	2-0-2-4-1-4-2-5-1
3	543	1730	3	1	437	0	100	0	1178	7.798138786764706	6.275850183823529	1.4361213235294117	11	8	11	2	11	145	556	2-8-2-17-17-6-13-17-14
4	1611	4311	4	1	601	1	100	0	2646	8.620505136986301	3.2159674657534247	0.5351027397260274	12	10	8	2	8	181	551	4-2-5-10-35-1-0-4-36
5	1218	3091	3	1	409	0	84	0	1739	9.079794847328245	3.0489623091603053	0.6261927480916031	11	5	7	2	7	196	551	2-0-7-9-31-0-0-3-30
6	1712	4431	4	1	584	1	106	0	2666	8.845899470899472	3.017526455026455	0.5477017195767195	11	10	9	2	8	180	551	2-2-7-12-37-1-0-4-38
7	1650	4317	4	1	583	0	114	0	2786	8.733485772357724	3.0858316395663956	0.6034044715447154	11	10	9	2	8	180	551	2-8-8-11-39-1-0-4-38
8	1503	3825	3	1	563	3	107	0	2220	8.86872167673716	3.3220827039274923	0.6313727341389728	15	10	9	2	8	170	469	2-4-7-13-37-5-0-3-34
9	1788	4649	4	1	598	0	123	0	2736	8.774340452261306	2.934594849246231	0.6036039572864321	13	10	9	2	8	170	469	3-5-7-16-40-6-0-4-38
10	1678	4331	4	1	530	0	115	0	2868	8.739583333333334	2.7604166666666665	0.5989583333333334	11	10	8	2	8	163	469	2-5-7-14-36-5-0-3-38
11	1304	3331	3	1	425	0	102	0	1950	8.93640350877193	2.9125548245614037	0.699013157894737	11	5	7	2	7	184	469	3-6-7-12-35-4-0-3-30
12	1513	3082	3	1	384	1	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]752767526	0.6774677121771218	14	5	4	2	4	118	219	2-4-7-13-29-5-0-3-29
13	1436	2921	3	1	371	0	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]346153846	0.6084735576923077	12	5	4	2	4	118	219	2-4-7-7-27-0-0-3-29
14	1445	2936	3	1	374	7	[illegible]	[illegible]	2537	[illegible]	[illegible]2911877	0.6435584291187739	12	5	4	2	4	118	219	3-4-7-10-27-0-0-3-29
15	1511	3095	3	1	376	0	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]235294	0.6534352022058824	12	5	4	2	4	118	219	3-4-7-12-27-4-0-3-29
16	4255	20499	21	30	690	2	[illegible]	21	[illegible]	[illegible]	[illegible]967213	3.2876536885245904	63	5	2	5	2	119	185	13-233-274-464-276-1381-1895-265-190
17	4255	20499	21	30	690	2	[illegible]	21	[illegible]	[illegible]	[illegible]967213	3.2876536885245904	63	5	2	5	2	119	185	13-233-274-464-276-1381-1895-265-190
18	3624	6273	3	1	869	0	117	0	4252	10.70820726172466	2.5677240922844176	0.3457119894099849	66	4	3	2	1	173	173	2-2-5-11-25-1-0-4-46
19	3623	6272	3	1	866	0	117	0	4239	10.72147253787878	2.562736742424242	0.3462357954545454	66	4	4	2	1	173	173	2-2-5-11-25-1-0-4-46
20	3638	6696	3	1	875	0	117	0	6986	8.102016818700115	1.9486708950969214	0.2605651368301026	66	3	4	2	2	119	173	2-2-5-11-25-1-0-4-46
21	3639	6698	3	1	873	0	117	0	6995	[illegible]	[illegible]9420024202733486	[illegible]	66	3	3	2	2	119	173	2-2-5-11-25-1-0-4-46
22	3639	6698	3	1	873	0	[illegible]	0	[illegible]	[illegible]	[illegible]733486	[illegible]	66	3	3	2	2	119	173	2-2-5-11-25-1-0-4-46
23	295	859	6	1	305	2	[illegible]	0	[illegible]	[illegible]	[illegible]738462	[illegible]	16	1	5	1	161	161	161	3-0-3-8-13-0-1-3-2
24	247	720	6	1	296	0	38	0	1245	8.614676339285714	10.323660714285714	1.3253348214285714	15	21	1	5	1	161	161	3-1-3-8-13-0-1-3-2
25	246	699	6	1	289	0	38	0	1245	8.579799107142858	10.079520089285714	1.3253348214285714	15	21	1	5	1	161	161	3-1-3-8-13-0-1-3-2
26	3940	17932	15	30	627	1	2950	27	24859	2.989[illegible]1895881896	[illegible]7576121794[illegible]	2.238429972804973	63	5	1	5	1	125	125	12-195-121-175-177-769-1319-75-49
27	3950	17779	15	30	627	1	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	1	5	1	125	125	12-195-122-175-177-784-1324-76-50
28	3572	15520	15	30	589	0	2935	[illegible]	17[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	6	1	7	1	101	101	12-194-123-163-184-786-1308-81-49
29	3573	15503	15	30	591	0	2919	14	187[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	6	1	5	1	101	101	12-195-120-156-188-781-1308-76-47
30	3700	16162	15	31	536	2	2908	1	19165	4.638358472400514	0.6719351732991014	3.6454990372272142	35	6	1	6	1	101	101	12-195-121-158-163-785-1323-73-43
31	3475	15342	15	30	578	0	2856	49	19078	[illegible]7431775137741	0.7774836432506887	3.8416838842975207	35	6	1	5	1	101	101	12-195-122-157-187-770-1255-76-48
32	3486	15385	15	30	567	0	2919	13	18[illegible]	[illegible]	[illegible]	[illegible]	35	[illegible]	1	5	1	101	101	12-195-123-156-183-769-1324-73-49
33	3551	15594	15	29	526	0	2[illegible]78	5[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	35	[illegible]	1	5	1	101	101	9-193-101-134-160-757-1277-76-48
34	3573	15686	15	30	591	24	2[illegible]93	16	18[illegible]	[illegible]	[illegible]	[illegible]	33	[illegible]	1	5	1	101	101	12-195-121-160-189-786-1304-81-49
35	434	1247	3	1	318	0	37	0	991	5.927666083916084	4.3433129370629375	0.505354020979021	10	5	1	6	1	101	101	2-1-2-10-5-0-0-3-9
36	430	1244	3	1	317	0	39	0	1003	5.873033216783217	4.329654720279721	0.5326704545454546	10	5	1	6	1	101	101	2-1-2-10-7-0-0-3-9
37	823	2836	4	2	669	0	1[illegible]	0	1467	5.00754[illegible]81931464	4.070531542056075	0.5[illegible]2803738	10	6	1	6	1	101	101	2-7-25-22-10-6-4-20-27
38	823	2836	4	2	669	0	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	6	1	6	1	101	101	2-7-25-22-10-6-4-20-27
39	238	937	0	0	158	0	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	1	2	2	1	100	100	11-15-9-2-4-1-0-0-1
40	234	829	0	0	291	0	31	0	583	8.161272321428571	10.149274553571429	1.0811941964285714	3	1	3	2	2	69	92	4-11-6-4-5-0-0-0-0
41	234	829	0	0	291	0	34	0	568	8.161272321428571	10.149274553571429	1.1858258928571428	3	1	3	2	2	78	92	5-11-6-5-5-0-0-0-0
42	234	829	0	0	291	0	34	0	568	8.161272321428571	10.149274553571429	1.1858258928571428	3	1	3	2	2	78	92	5-11-6-5-5-0-0-0-0
43	235	830	0	0	295	0	38	0	560	8.196149553571429	10.288783482142858	1.3253348214285714	3	1	3	2	2	78	92	5-9-11-5-6-0-0-0-0
44	225	781	0	0	138	0	29	0	480	10.21984011627907	6.268168604651163	1.3172238372093024	0	1	3	2	2	69	92	7-9-4-2-4-1-0-0-0
45	1299	3376	4	1	417	0	96	0	2350	8.513789848993289	2.733064177852349	0.6291946308724833	11	10	4	2	4	67	86	3-1-6-11-32-1-0-3-36
46	1298	3371	4	1	415	0	106	0	2391	8.53587962962963	2.7291140572390575	0.6970749158249159	11	10	4	2	4	67	86	3-4-13-11-32-1-0-3-36
47	222	780	0	0	118	0	33	0	487	10.32366071428571	5.48735119047619	1.5345982142857142	0	1	1	2	1	85	85	5-16-7-0-3-0-0-0-0
48	214	840	0	0	255	0	24	0	458	8.359375	9.9609375	0.9375	3	1	3	2	2	68	84	3-4-7-4-4-0-0-0-1
49	196	768	0	0	255	0	29	0	477	8.144946808510639	10.596742021276595	1.2051196808510638	3	1	3	2	2	68	84	3-5-12-5-2-0-0-0-1
50	195	765	0	0	260	0	33	0	477	7.9345703125	10.579427083333334	1.3427734375	3	2	3	2	2	68	84	3-5-12-5-4-1-0-0-0
51	228	915	0	0	298	0	40	0	536	7.952008928571429	10.393415178571429	1.3950892857142856	3	1	3	2	2	82	84	4-8-13-6-5-1-0-0-1
52	229	909	0	0	140	0	30	0	460	9.318033854166666	5.696614583333333	1.220703125	0	1	1	2	1	57	57	4-5-4-13-2-0-0-1-0
53	229	909	0	0	140	0	30	0	460	9.318033854166666	5.696614583333333	1.220703125	0	1	1	2	1	57	57	4-5-4-13-2-0-0-1-0
54	228	922	0	0	121	0	34	0	457	9.474734042553191	5.0282579787234045	1.4128989361702127	0	1	1	2	1	57	57	4-8-5-12-2-1-0-1-0

Discussion:
Cost, Scalability, Resilience, Reliability

Cost: parsing time and setup, cost to deploy, cost to maintain

Scalability: ease of feature extraction and adaption

Resilience: robustness against changes in binaries/infrastructure

Reliability: correctness of data

Feature integration with MISP

Feature integration with MISP

1. Objects to group indicators as one entity
2. Feasible way to extract the indicators from binaries & graphs
3. Organise, store & display everything
4. Means for object interconnection & correlation

MASTERPLAN

Object definition which can be plugged into MISP

PE & graph feature extraction

Mapping of features to object definition

Generate a JSON file in MISP Object format

Implementation of objects in MISP core

Objects for other file formats

Soon-ish: string search, automatic correlation on per-instance basis

Later-ish: behaviour gadget search, straight from the graphs

Money is a factor, always

Attacker success doesn‘t correlate with attacker sophistication

Attacker sophistication is correlated to attacker budget

No attacker has infinite budget

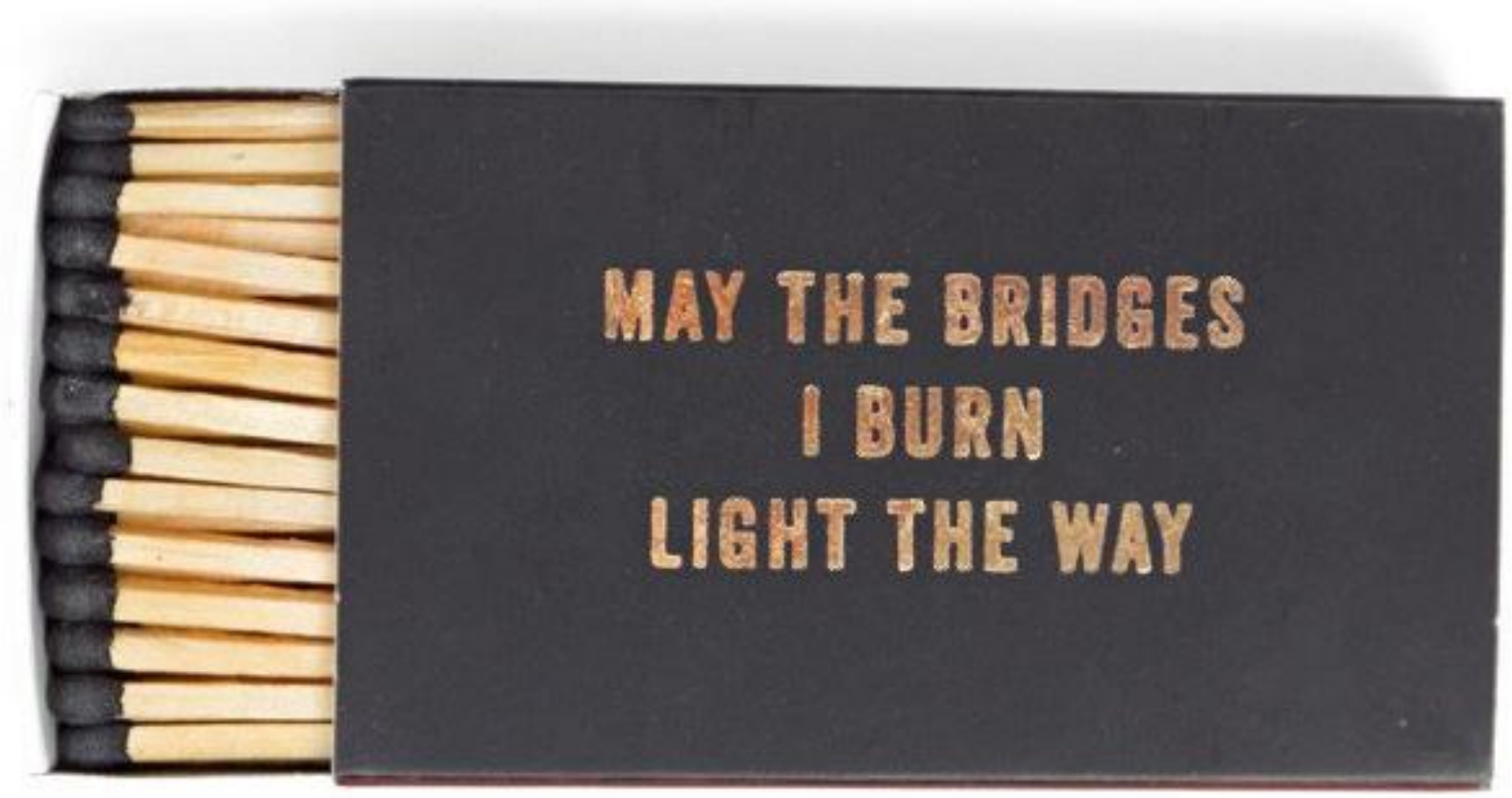
Reuse is a weak spot

Threat detection relies on reuse

Driving down defense cost to some extent drives up attack cost

Wrapping it up

thank you



Marion Marschalek
@pinkflawd

Eireann Leverett
@blackswanburst