NEXT-GENERATION OF EXPLOIT KIT DETECTION BY BUILDING SIMULATED OBFUSCATORS

Tongbo Luo & Xing Jin
Palo Alto Networks

paloalto networks

Agenda

- Background and Motivation
- Challenges
- Our Approach and Lessons We learned
- Result and Our Observation

paloalto networks.

Motivation

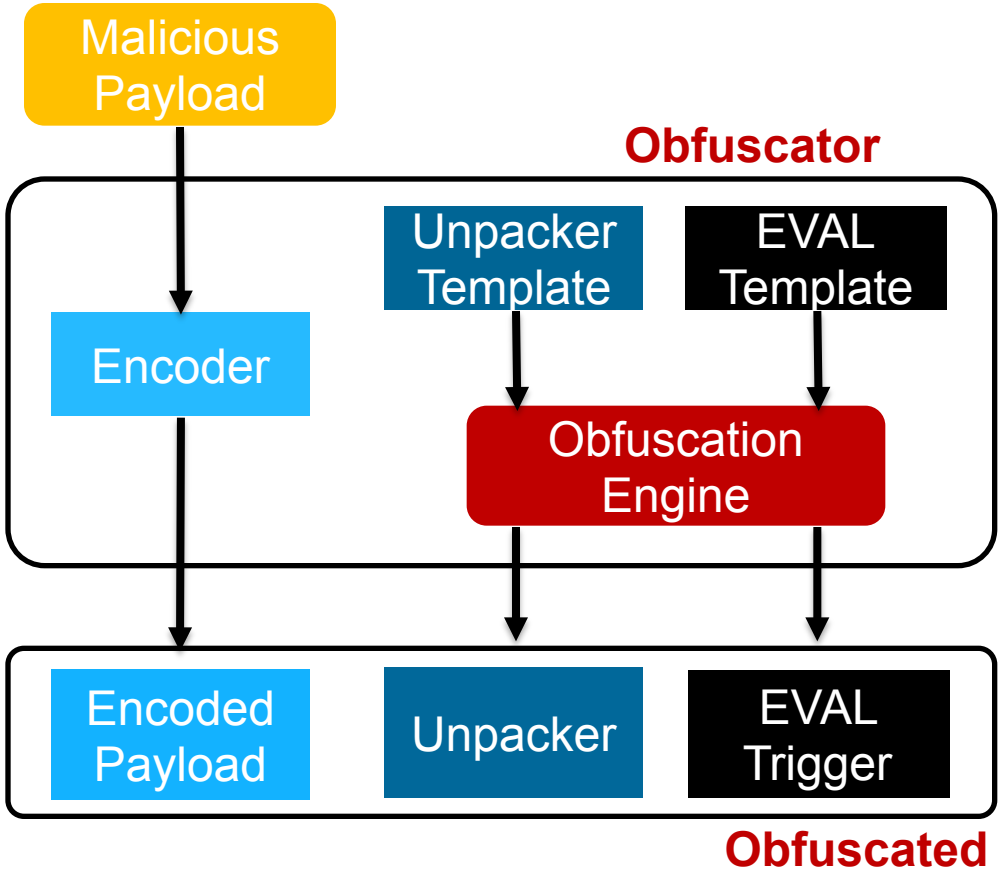
www.jsDarwin.com

Similarity-based Detection on malicious Exploit Kit JavaScript

Large Number of Samples Required

paloalto networks.

Exploit Kit Obfuscator



Reconstruct Encoder

Identify Template From Obfuscated Page

Reproduce Engine Logic

paloalto networks

Obfuscator Reverse-Engineering

```
window [‘e’+’v’+’a’+’l’]
                [wsq(xyz, fxj)];
```

```
window [‘ev’+’al’]
                [zxh(abc, sih)];
```

```
window [‘aevala’.substr(1,4)]
                [zxh(abc, sih)];
```

Obfuscation Engine
(variant)

```
“window [“ + $[eval_ctr] + “] [”
                + $v[0] +”(“ v[1]
                    + ”,” + v[2] + ”)” + “]”
```

EVAL Template
(version)

paloalto networks.

Challenges

1. Code Complexity

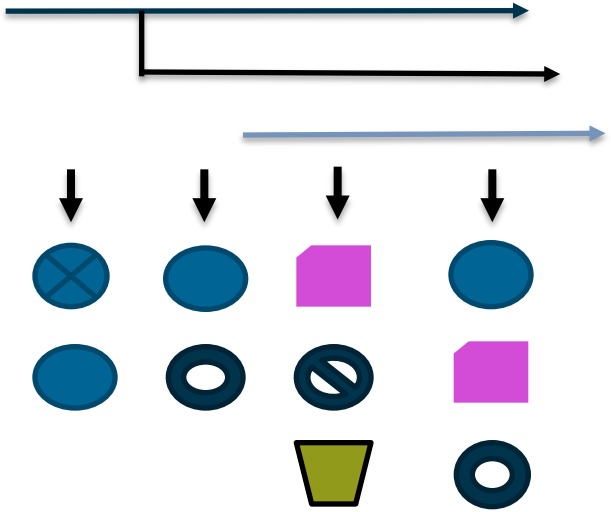
```
<script>
function pDA(AUT){ ... }
ZzF ="lnFloa";
function vzLh(Hpv){gEao="oi"+""+"n",aOC=window,SLVmg="e"+"x",
bUdR="R"+"e"+""+"g",FMwVS="o"+""+"u"+""+"te"+"rH
",HjEn="u"+"s"+""+"e"+"r",Wyc="b"+"o",Wbf="e"+"R"+""+"x";var qVa=new
aOC[bUdR+'Exp']('MSIE (\d+\.\d+);');var xngS=navigator[HjEn+'Agent'];var
prC=qVa[(Wbf+'ec').replace("R","")](xngS);var
$mrh=!prC,ryG;$mrh=!$mrh;if($mrh){ryG=prC[1]}GcSX='.*'+'>(.'+'*?)'+'<\/p';
if( ... ){GcSX='.'+ '?>('+'.*'+'?)'+'<\/p'}else { ... } VdZW =
["r"+"e"+"p"+"l"+"a"+"c"+"e"]["j"+gEao]();var UkP=document[Wyc+'dy'],
yh$f=UkP[FMwVS+'TML'],wvnV="", El=new aOC[bUdR+'Exp'](ZzF+Hpv+GcSX,'gi');wvnV=
El[SLVmg+'ec'] (yh$f);wvnV=wvnV [1];var WMK =
wvnV[VdZW](/[-]/g,"/")[VdZW](/[_]/g,"+");function asafas(a){return a}function
GKcx(s) { ... }if(!pDA(GKcx(WMK)))alert("Error");}vzLh("dNTvXuVoyq");
vzLh("xWgaeJKYcclAu");</script>
```

Hundred lines of code
Random variables

paloalto networks.

Challenges

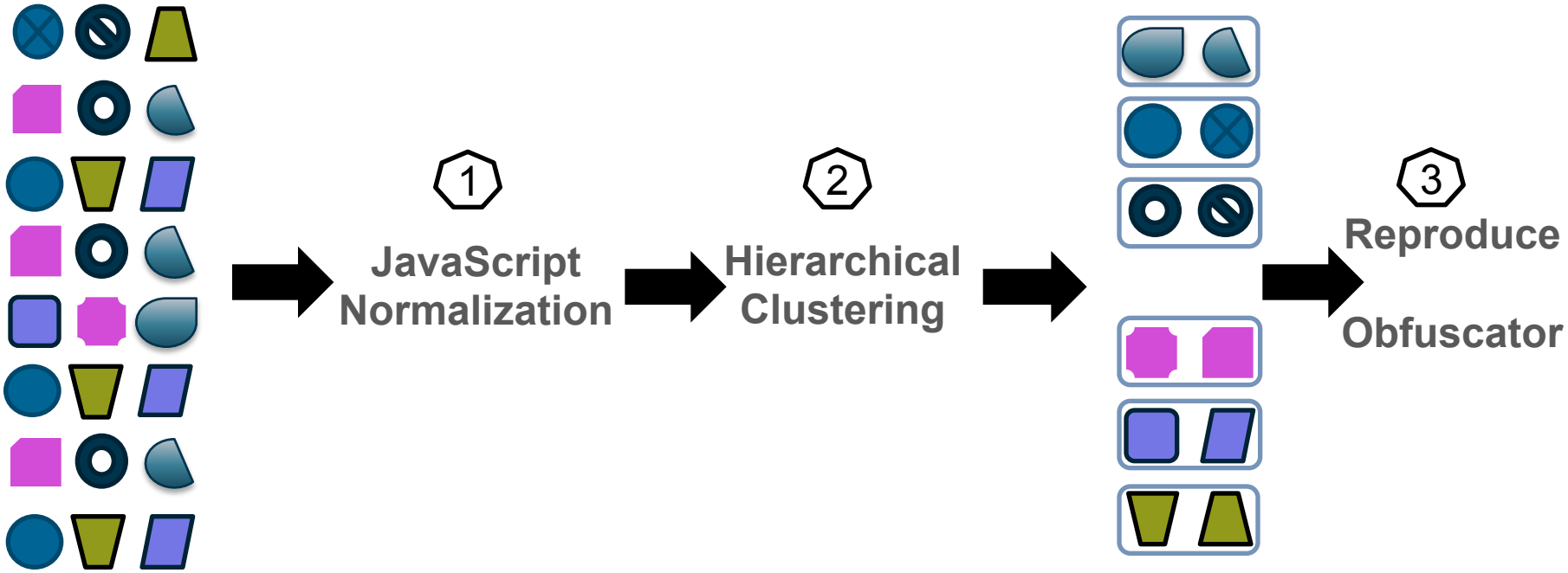
2. Data Complexity



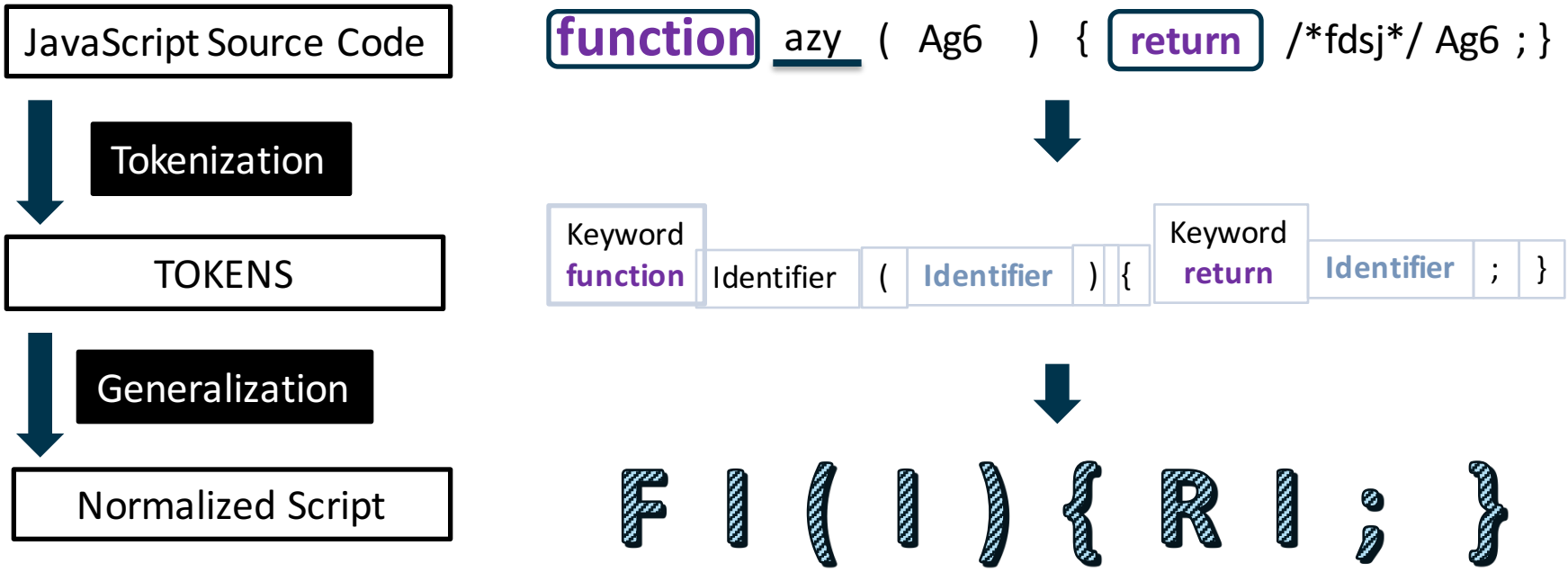
Big data set (~20000 Samples over 2 year period)
Mixed versions and variants

paloalto networks.

Overview of Our Approach



JavaScript Normalization



- Ignore Superficial Obfuscation (e.g. Randomized variable names)
- Normalized Script ⇔ Structure of the Code

Statistics

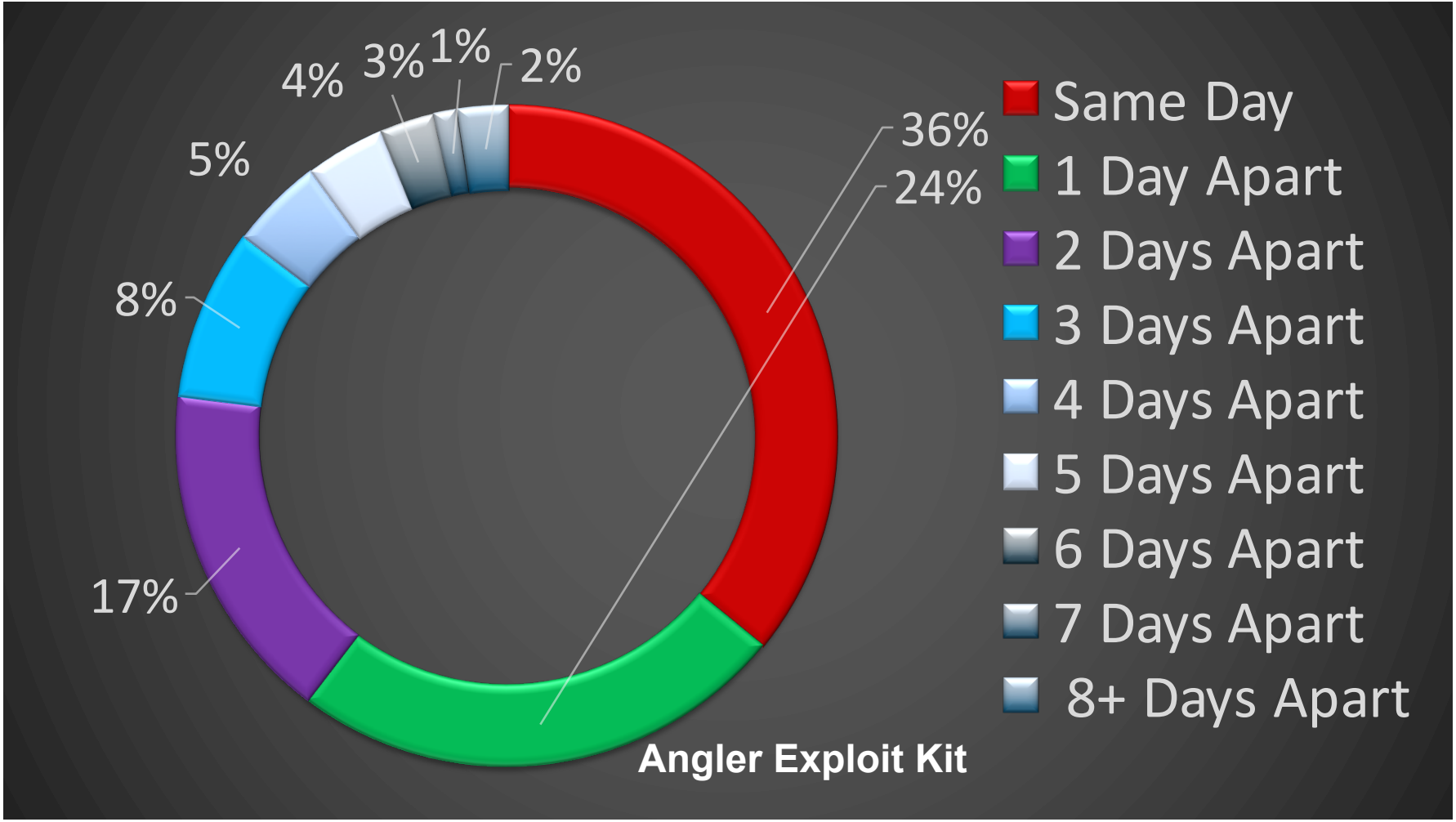
Total Number of samples we collected

Family	Angler	Nuclear	Rig	KaiXin	Fiesta
TOTAL #	7834	1303	1793	10291	79
UNIQUE #	613	107	344	1543	48

Total Number of samples
With Unique Normalized script

```
function azy ( Ag6 ) { return Ag6 ; }
function F$x3j( k2c5x ) { return k2c5x ; }
```

```
F I ( I ) { R I ; }
```

Statistics on Normalized Samples

Clustering

- Goal: Cluster Samples based on their obfuscator
- Observation: Similar Structure ➔ Generated by Similar Obfuscator
- Define Similarity.

$$SimScore(norm1, norm2) = 1 - \frac{EditDist(norm1, norm2)}{max(len(norm1), len(norm2))}$$

  - Similarity Score [0 , 1]
  - 1 = Identical, 0 = Different

paloalto networks.

Hierarchical *vs* Flat

- Flat Model: (K-Means)
  - Easy and Efficient
  - Drawbacks
- Require predefined **K** as input
  - **K** ⇔ Number of Obfuscator Version
  - Hard to Predict, lack of knowledge
  - Will be Changed over time.

paloalto networks.

Agglomerative (Bottom-Up) Hierarchical Clustering

Dendrogram

What is the proper Threshold to Identify Obfuscator Version/Variant?

[0.4 – 0.5]

Identify Obfuscator Version

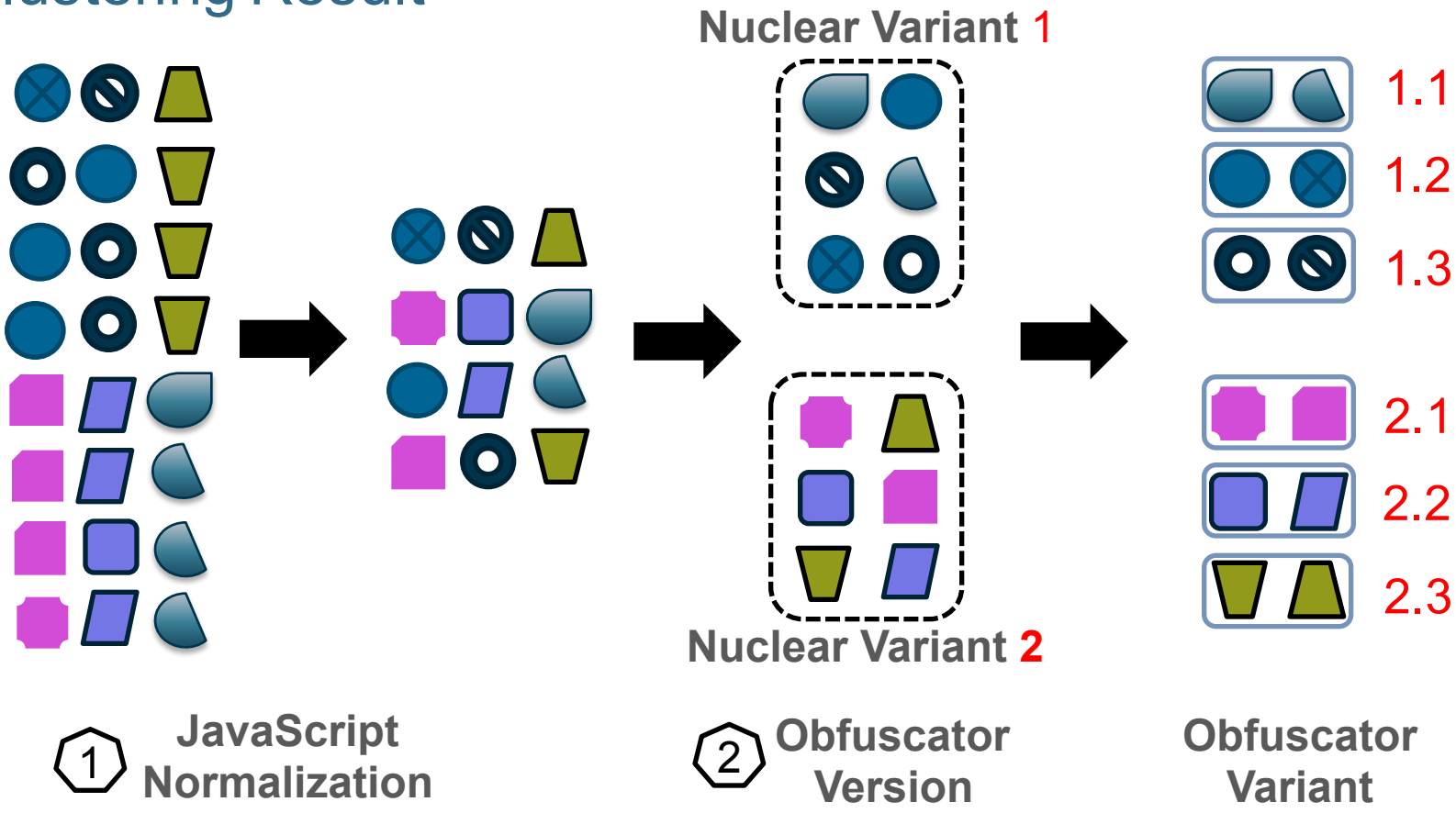
[0.78 ~ 0.85]

Identify Obfuscator Variants

Threshold vs K

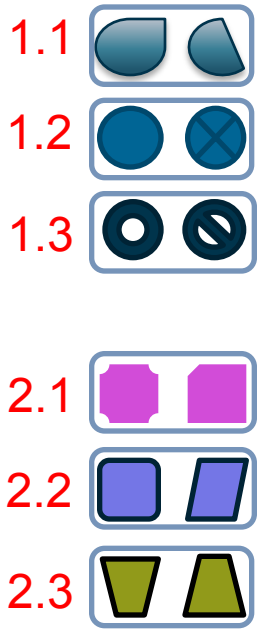
paloalto networks

Clustering Result



paloalto networks.

Reproduce Obfuscator for Each Cluster



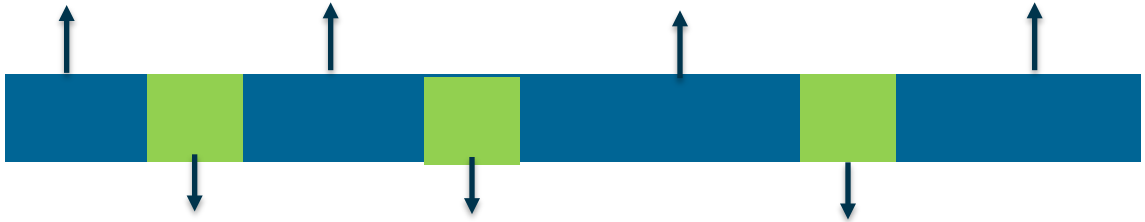
Obfuscator Variant

```
window ['e'+'v'+'a'+'l'][wsq(xyz,  fxj)];
window ['ev'+'al'] [zxh(abc, sih)];
window [l28f + i8d4][zxh(abc, sih)];
```

```
I['S'+'S'+'S'+'S'][I(I,I)];
I['S'+'S'                ][I(I,I)];
I[  I +I                 ][I(I,I)];
```

Threshold 0.80 ➔
80% Similar Within Cluster

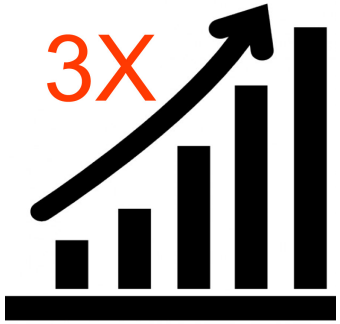
Template



Obfuscator Engine

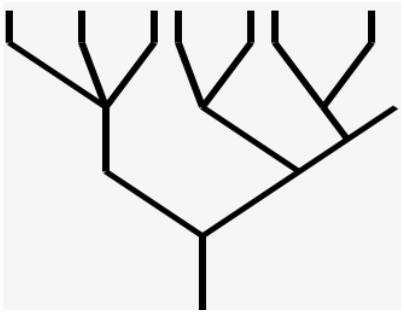
paloalto networks

Why This Research



Boost
Samples
Set

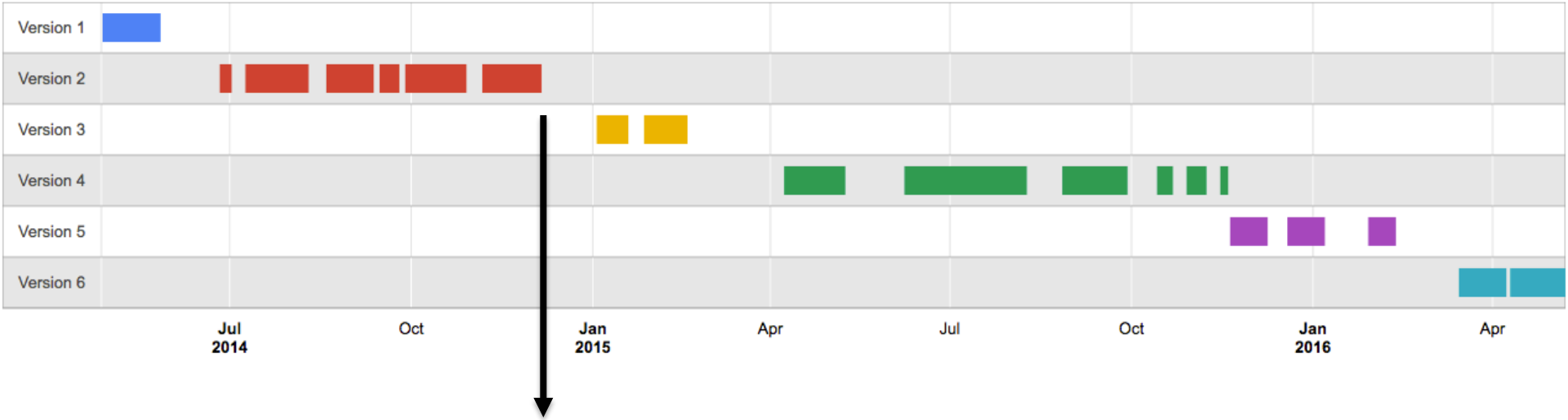
Improve
Detection
Rate



Better
Understanding
on Obfuscator

paloalto networks.

Life Cycle of Nuclear Exploit Kit Obfuscator

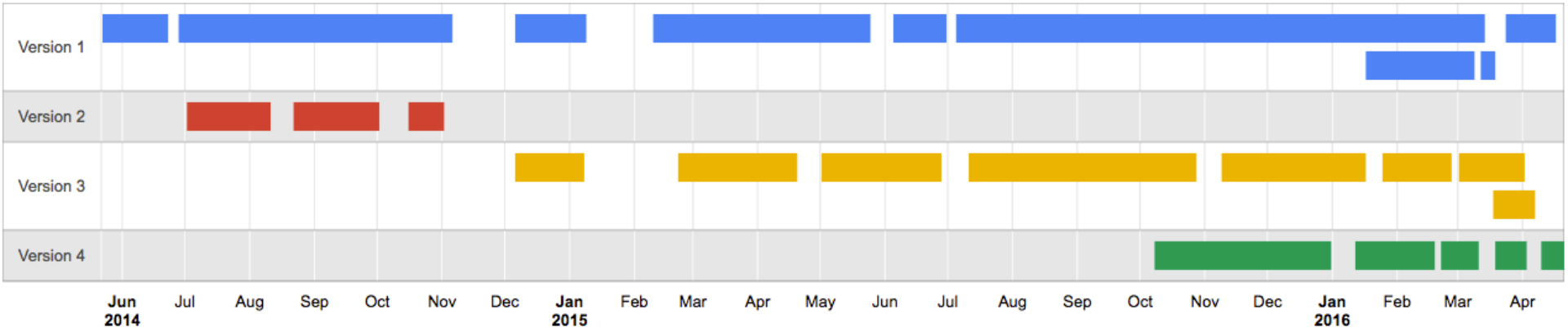


During **December 2014**, a new version of Nuclear Pack emerged. … new version will **completely replace the old** version.

http://community.websense.com/blogs/securitylabs/archive/2015/01/15/evolution-of-an-exploit-kit-nuclear-pack.aspx

paloalto networks.

Life Cycle of Angler Exploit Kit Obfuscator



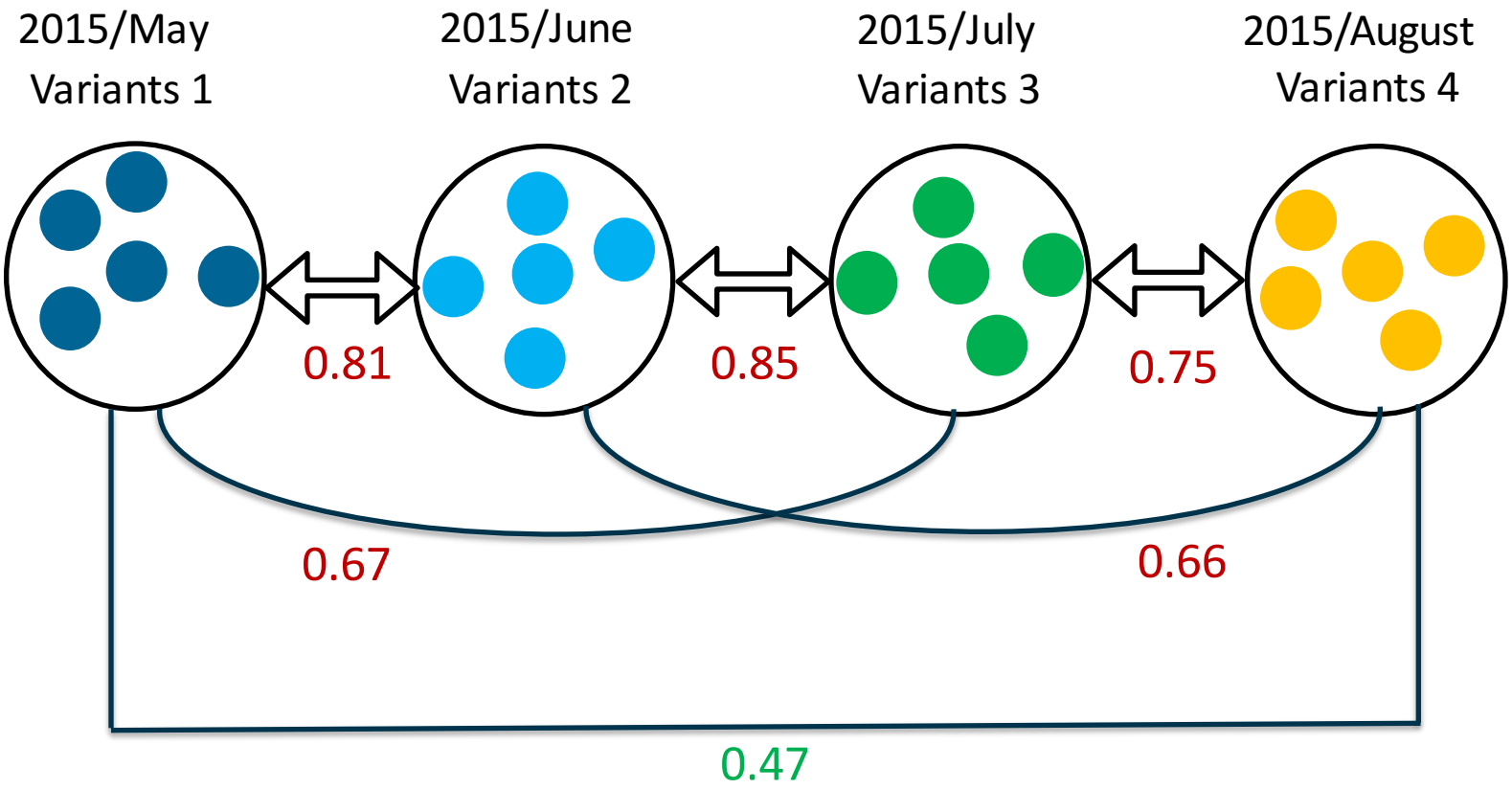
- Extremely Prevalent
- Aggressive tactics for evading detection
- **Only few Obfuscator Versions / Variants.**

# of Domains Deployed

Version 1: 519
Version 2: 89
Version 3: 753
Version 4: 68
Version 5: 234

Evolution of Variants

Nuclear Obfuscator Version 2



TAKEAWAY

- A new Angle to Explore Exploit-Kit
- The novel method to boost sample set and improve detection rate by reproducing obfuscator.
- The Evolution of Obfuscator in the wild

paloalto networks.

Q & A

https://github.com/irobert-tluo/rebuild_obfuscator

paloalto networks.