A Word Graph Approach for Dictionary Detection and Extraction in DGA Domain Names

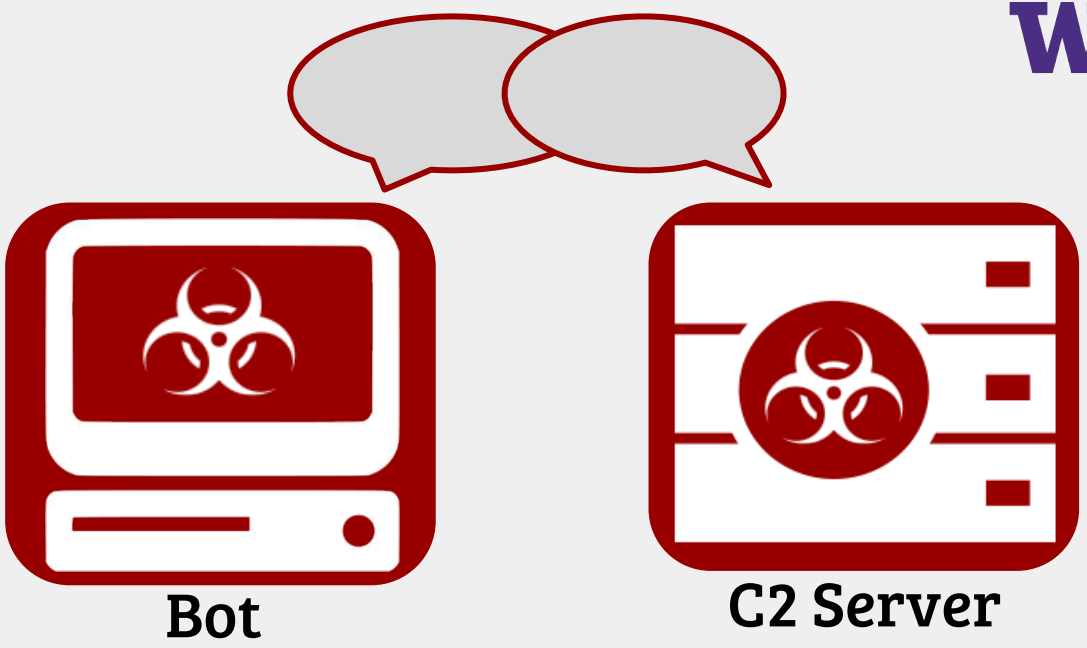
Mayana Pereira, Shaun Coleman, Bin Yu,
Martine De Cock, Anderson Nascimento

Infoblox
CONTROL YOUR NETWORK

Motivation

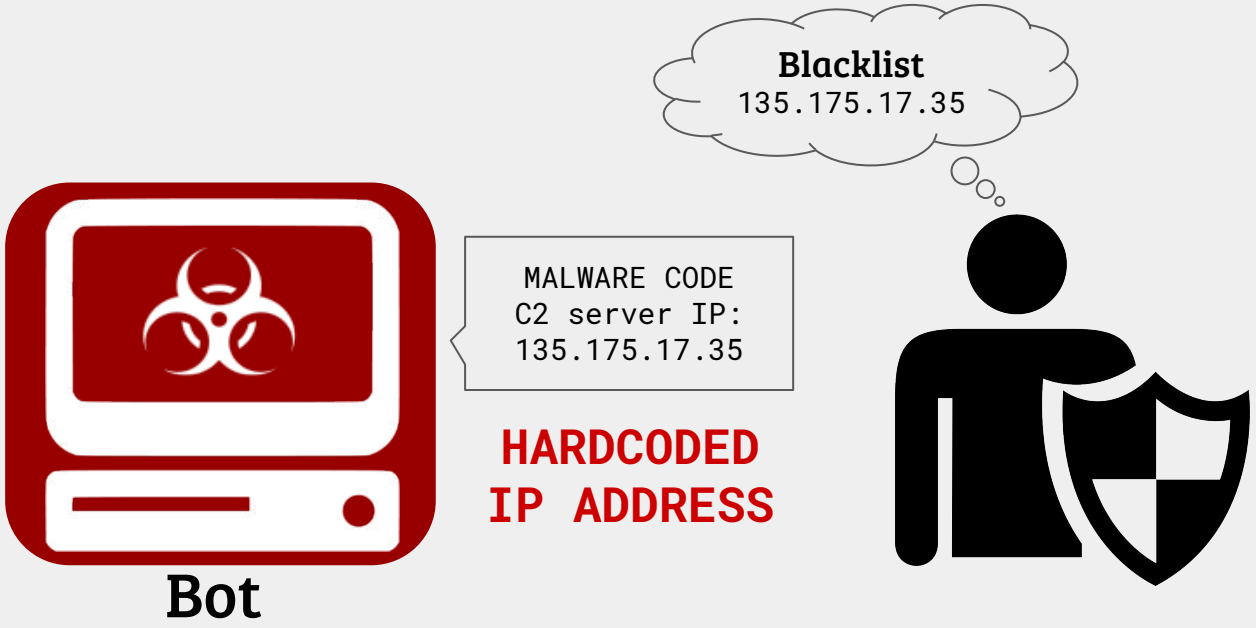
Source:
https://www.accenture.com/t20170926T072837Z__w__/us-en/_acnmedia/PDF-61/Accenture-2017-CostCyberCrimeStudy.pdf
https://www.csoonline.com/article/3153707/security/top-5-cybersecurity-facts-figures-and-statistics-for-2017.html

Communication Between bots and C2 Servers

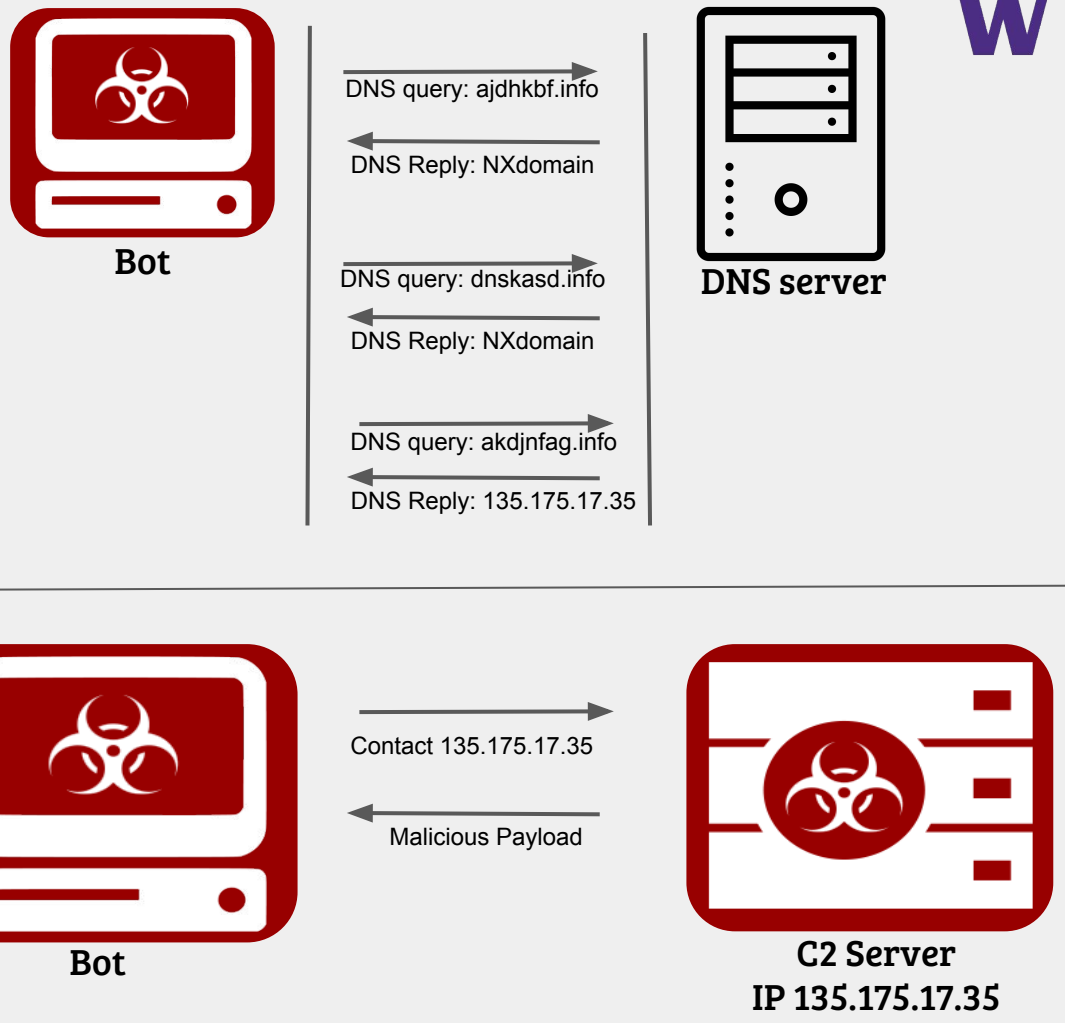


Accountability
Activation
Updates
Sending Back Stolen Information

Communication Between bots and C2 Servers



DGA: Domain Generation Algorithm



How does DGA Detection Models work?

Good - wikipedia.org

Bad - nn4rzw6r4yv4ezapuu.ru

Works by Differentiating
Characters Probability Distributions

[1] Schiavoni, Stefano, et al. "Phoenix: DGA-based botnet tracking and intelligence." *International Conference on Detection of Intrusions and Malware, and Vulnerability Assessment*. Springer, Cham, 2014.

[2] Antonakakis, Manos, et al. "From Throw-Away Traffic to Bots: Detecting the Rise of DGA-Based Malware." *USENIX security symposium*. Vol. 12. 2012.

[3] Yadav, Sandeep, et al. "Detecting algorithmically generated malicious domain names." *Proceedings of the 10th ACM SIGCOMM conference on Internet measurement*. ACM, 2010.

How does DGA Detection Models work?

Good - wikipedia.org

Bad - nn4rzw6r4yv4ezapuu.ru

Bad - wintermeasure.net

How dictionary DGA domains are formed...

jacquelynchristophers.net.
gweneverechristison.net.
christianchristianson.net.
rosalynnemottershead.net.
creightonthaddeus.net.
jacquelynjeremiah.net.
creightonnathaniel.net.
priscilladwerryhouse.net.
christinajeremiah.net.

facegone.net.
walkroad.net.
weakdont.net.
sellfool.net.
weakheat.net.
deepaunt.net.
facethey.net.
ballpull.net.
pushaunt.net.
walklift.net.
bothfive.net.
facegoes.net.

Suppobox Malware Domains

Words are used repeatedly!

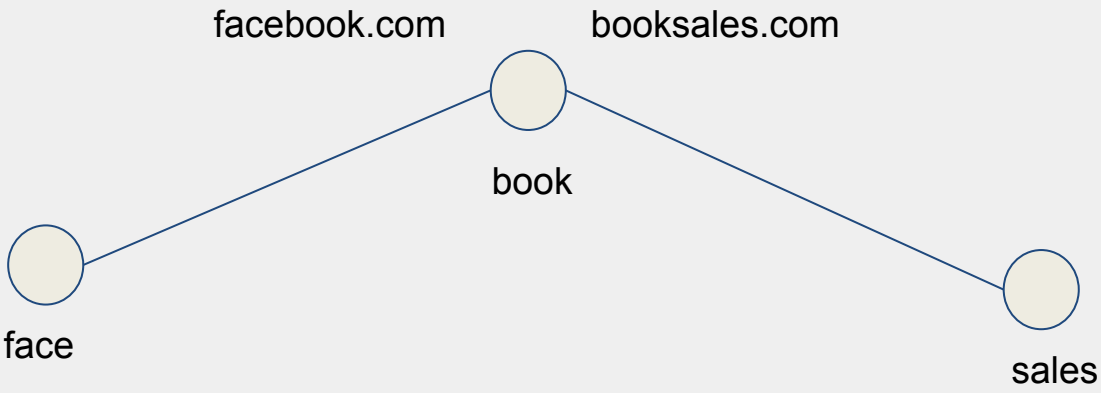
Contributions

We propose a method that:

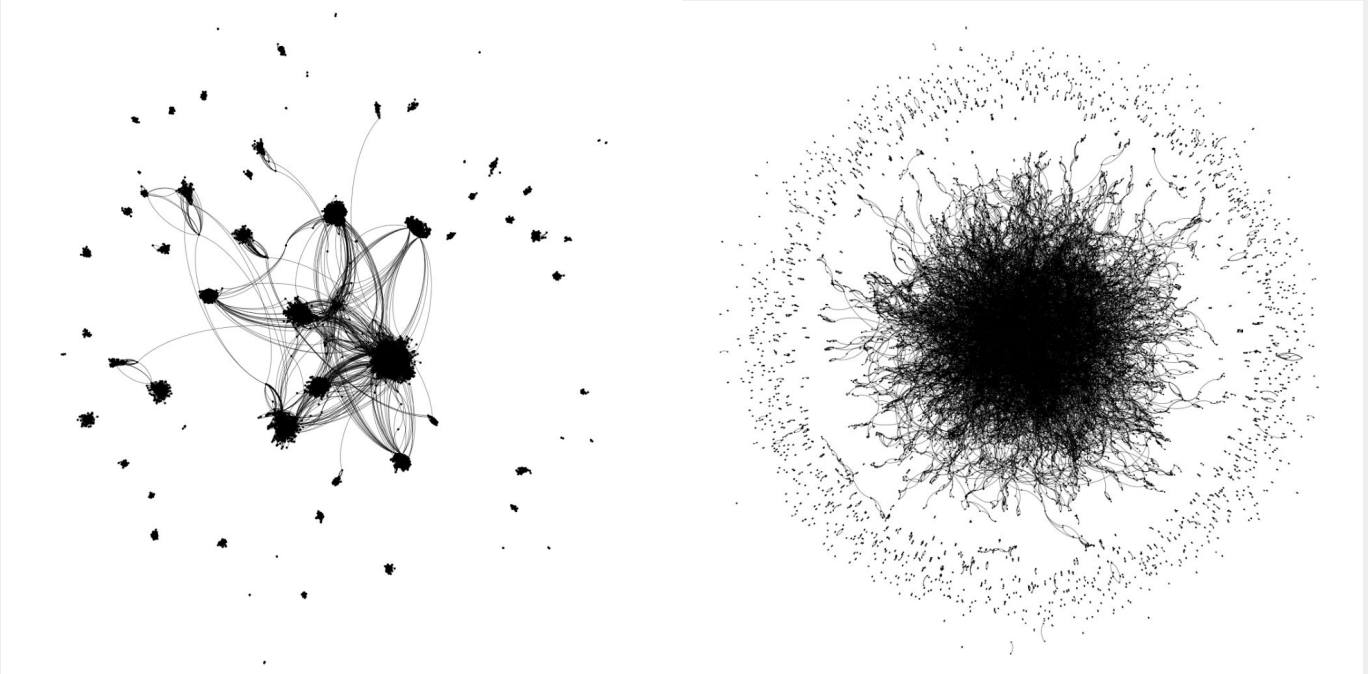
1. Detects domains and **Extracts the dictionary** from dictionary DGAs.

2. Robust **against changes** in the dictionary.

Assume there is an algorithm for finding “words” within a domain

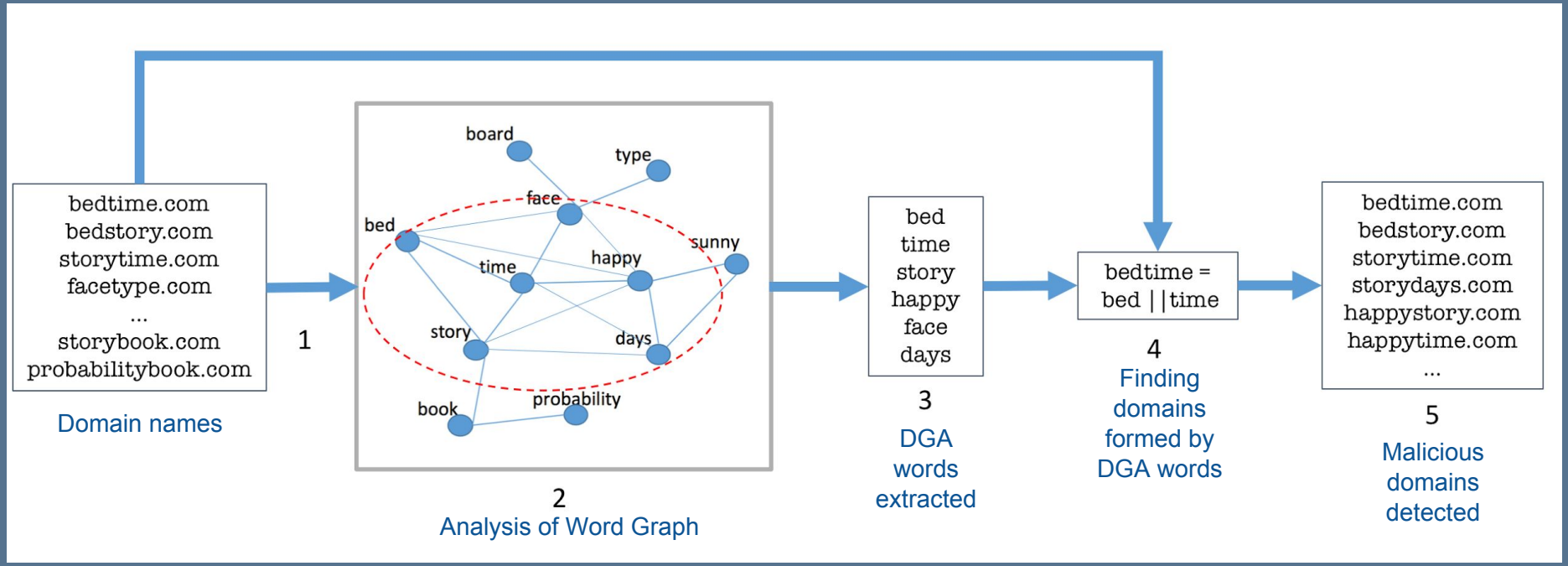


DGA words connect differently!



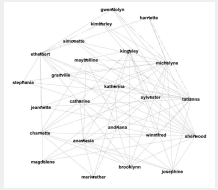
DGA

Benign

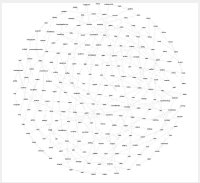


We extract the dictionaries <u>without</u> Reverse Engineering efforts!

Finding Malicious Regions in the graph



DGA



non-DGA

ID	$Feature_1$	$Feature_2$	$Feature_3$	$Feature_4$

Description vector

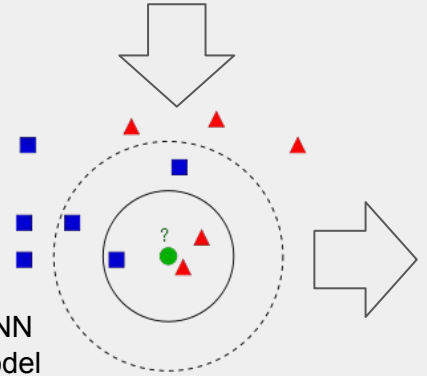
Dataset

1	$Feature_1$	$Feature_2$	$Feature_3$	$Feature_4$
2	$Feature_1$	$Feature_2$	$Feature_3$	$Feature_4$
...				
n	$Feature_1$	$Feature_2$	$Feature_3$	$Feature_4$

Filter Nodes with low degree
less than **n (n=4)**

Features from each graph component G:

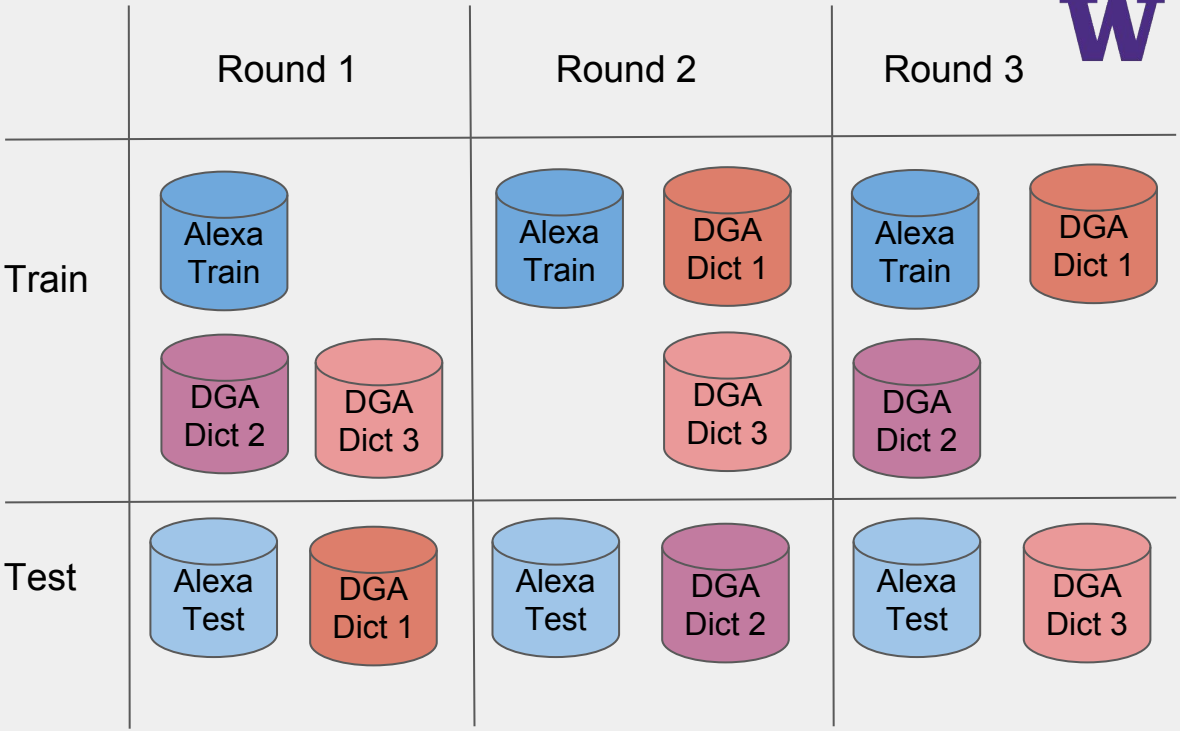
I. Average node degree
II. Maximum node degree
III. Number of cycles which form a basis of cycles of G
IV. Average cycles per node



K-NN model

DGA
non-DGA
...
DGA

Methodology



Unbalanced Dataset: DGA domains are less than 1%

Alexa dataset: Benign domains from Alexa (alexa.com) Top 120k domains. 80,000 domains for training and 40,000 domains for testing.

DGA dataset: Suppobox DGA domains. 1,020 DGA domains. Generated using 3 different dictionaries (340 domains per dictionary).

Results

Word Detection Results

	# of words used by DGA	# of detected words	Recall	FPR
Round 1	92	92	1	0
Round 2	70	64	0.91	0
Round 3	80	80	1	0

Classification Results

	Round 1			Round 2			Round 3		
Model	Precision	Recall	FPR	Precision	Recall	FPR	Precision	Recall	FPR
WordGraph	1	1	0	1	0.96	0	1	1	0
Random Forest (Baseline)	0.056	0.009	10^{-3}	0.031	0.006	10^{-3}	0.0	0.0	10^{-3}

Remarks

- The Method have been used to extract dictionaries from real traffic, extracting known and unknown dictionaries (validation being conducted by security experts).

- We have been investigating the relationship between the dictionary size and amount of domains we need to capture in order to extract dictionaries.

- Efficiency: In datasets with 2M domains entire algorithm runs in ~100 minutes. (Word Extraction + Graph Analysis)

Summary

- First Algorithm that aims at detecting dictionary DGA domains
- We are able to extract 97,5% of the used dictionary with a few hundred domains.
- Our method is completely **independent of the dictionary** that is used by the malware

QUESTIONS?

Thank you!

mpereira@infoblox.com

Infoblox
CONTROL YOUR NETWORK

Results

Word Detection Results

Words Round 1-> ['within', 'belong', 'early', 'would', 'distant', 'clothes', 'journey', 'remember', 'smell', 'safety', 'forget', 'little', 'effort', 'separate', 'ridden', 'husband', 'those', 'destroy', 'chair', 'future', 'through', 'health', 'suffer', 'increase', 'known', 'follow', 'already', 'woman', 'storm', 'fight', 'period', 'choose', 'summer', 'water', 'fresh', 'thrown', 'smoke', 'thought', 'hunger', 'gentleman', 'party', 'crowd', 'member', 'however', 'experience', 'although', 'begin', 'training', 'degree', 'morning', 'class', 'heavy', 'share', 'likely', 'history', 'order', 'weather', 'return', 'answer', 'student', 'glass', 'alone', 'shake', 'succeed', 'present', 'think', 'nearly', 'leader', 'require', 'glossary', 'strange', 'various', 'chief', 'college', 'heaven', 'often', 'twelve', 'worth', 'necessary', 'difficult', 'happen', 'rather', 'pleasant', 'amount', 'middle', 'produce', 'thick', 'heard', 'gentle', 'round', 'forward', 'between']

Words Round 2-> ['hello', 'face', 'sell', 'fish', 'lady', 'wing', 'weak', 'after', 'live', 'drive', 'queen', 'peace', 'guide', 'half', 'field', 'force', 'late', 'story', 'mine', 'name', 'house', 'tuesday', 'both', 'gift', 'month', 'least', 'serve', 'walk', 'wednesday', 'past', 'nail', 'gain', 'august', 'under', **'octover'**, 'then', 'lend', 'meat', 'case', 'raise', 'these', 'born', 'meet', 'sight', 'price', 'tried', 'with', 'duty', 'quick', 'milk', 'most', 'horse', 'food', 'cloud', 'sick', 'sunday', 'monday', 'reach', 'enjoy', 'head', 'world', 'feed', 'dark', **'croud'**]

Words Round 3 -> ['cornelius', 'christianson', 'winchester', 'christison', 'madeline', 'josceline', 'coriander', 'calanthia', 'seraphina', 'paternoster', 'johnathon', 'marigold', 'radclyffe', 'maryvonne', 'raschelle', 'trevelyan', 'columbine', 'sharmaine', 'bethanie', 'katherine', 'nathaniel', 'katheryne', 'september', 'terrence', 'madelaine', 'quintella', 'autenberry', 'summerfield', 'roosevelt', 'christmas', 'mottershead', 'michaelson', 'oliverson', 'shaniqua', 'blackburn', 'earnestine', 'alexandrina', 'bartholomew', 'anjelica', 'washington', 'richardine', 'gwendoline', 'willoughby', 'pemberton', 'maximillian', 'masterson', 'evangelina', 'mariabella', 'harmonie', 'veronica', 'evangeline', 'beauregard', 'christiana', 'wilhelmina', 'dulcibella', 'sacheverell', 'tatianna', 'winnifred', 'maybelline', 'kimberley', 'granville', 'stephania', 'anastasia', 'simonette', 'kingsley', 'harriette', 'andriana', 'catharine', 'gwendolyn', 'jeannette', 'sherwood', 'brooklynn', 'michelyne', 'ethelbert', 'josephine', 'magdalene', 'katherina', 'meriwether', 'charnette', 'sylvester']

Random Forest Features

- ent: normalized entropy of characters
- nl2: median of 2-gram
- nl3: median of 3-gram
- naz: symbol character ratio
- hex: hex character ratio
- vwl: vowel character ratio
- len: domain label length
- gni: gini index of characters
- cer: classification error of characters
- tld: TLD hash
- dgt: digital character ratio