Detecting Botnets communicating with transient C2 servers

Authors:
Khanh Hoang Huynh
Mathijs Visser

Date:
02-07-2020

Supervisor:
Eddie Bijnen (True.nl)

Introduction

- Botnets, Botnet Command & Control, Bot master
- In 2019 Spamhaus Project reported has doubled of in detection of Botnet C2 servers in comparison to 2017[1]
- 16% of the increase in C2 servers can be attributed to bulletproof hosting providers
- IP-addresses are no longer useful for detecting and blocking of C2 traffic

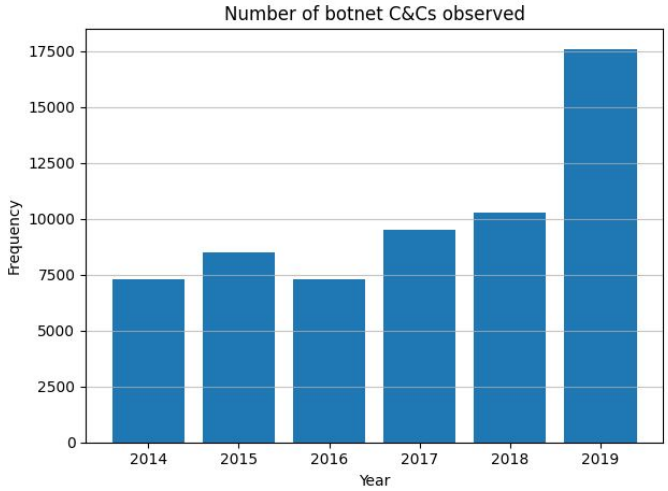


Figure 1: Observed Botnet C&Cs through recent years, reported by spamhaus[1]

1: https://www.spamhaus.org/news/images/full-2019/spamhaus-botnet-threat-report-2019.pdf

Research Goals

- Detecting Botnet traffic in a Network
- Find out what characteristics C2 domains have in common
- Find out what network characteristics C2 domains have in common
- Not using Deep Packet Inspection

Research Question

Main Research Question

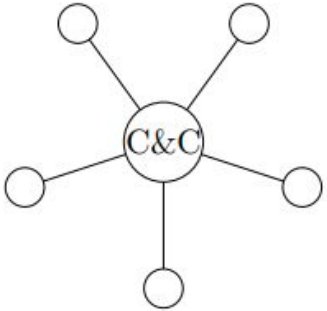
How can malicious traffic to and from transient Command and Control servers be detected using DNS and NetFlow data?

Sub Questions

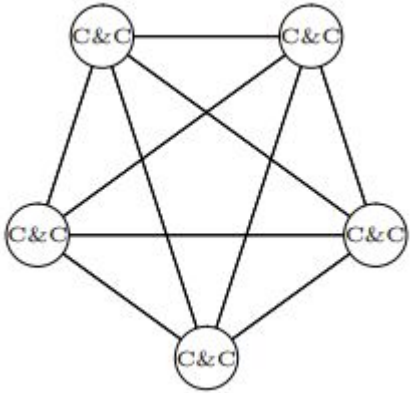
- What domain features can be used to detect transient command and control servers?
- What NetFlow features can be used to detect transient command and control servers?

Botnets

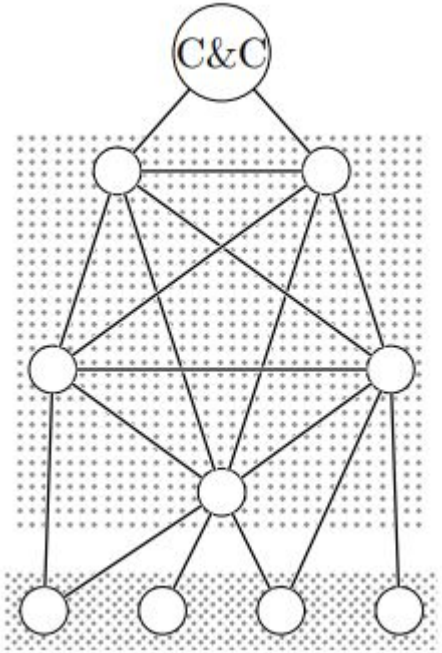
- Centralized architecture
- Peer to Peer (P2P) architecture
- Hybrid architecture



Client-server C&C



Peer-to-peer C&C



Hybrid C&C

Figure 2: The botnet variants of today

Figure 2 source: https://doi.org/10.1109/COMST.2017.2749442

Relative Entropy (Kullback-Leibler Divergence)

- Context of Digital information: **Measure of randomness** or uncertainty in data
- Shannon Entropy is normally used to measure randomness
- Relative Entropy allows you to compare random data with another distribution

Relative Entropy Formula:

$$D_{KL}(P||Q) = \sum_i p_i log(\frac{p_i}{q_i})$$

- P = Distribution you want to compare
- Q = Distribution you compare with (Baseline Distribution)
- Log base = |Q|

Example:
Hello
P = {0.2, 0.2,0.4,0.2}
Q = {0.06, 0.11, 0.04, 0.08}[1]

D_{KL}=0.2 * $\log_{26}$(0.2 / 0.06} + 0.2 * $\log_{26}$(0.2 / 0.11} + …

1: https://en.wikipedia.org/wiki/Letter_frequency

Relevance Vector Machine

- Similar to the Support Vector Machine
- Works well with high dimensional features
- Relatively efficient when compared to other algorithms
- Provides probability of classification

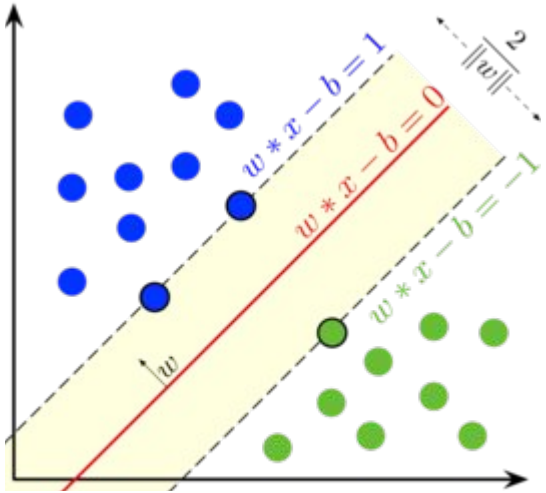


Figure 3: SVM classifying example

Figure 3 source: https://en.wikipedia.org/wiki/Support_vector_machine

Proof of Concept (PoC)

- Botnet Detector using standard network traffic monitoring data DNS and Netflow data

- Two main components:
  - **Domain Classifier using DNS**
  - Disclosure[1]: Botnet Detector using Netflow data

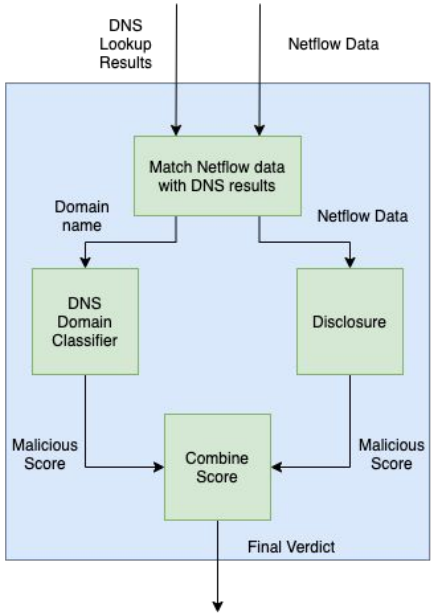


Figure 4: PoC Architecture

1: https://doi.org/10.1145/2420950.2420969

Disclosure

- Existing NetFlow detection system
- Flow sizes
- Client access patterns
- Temporal features

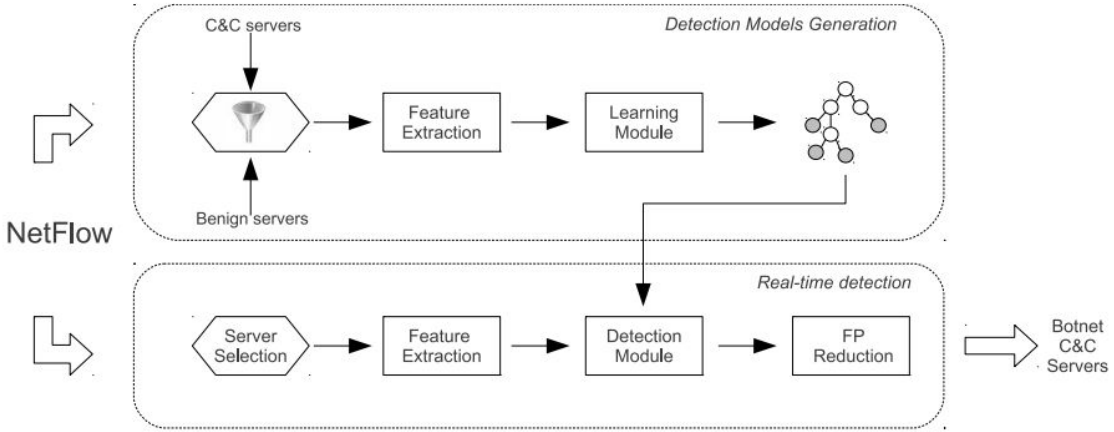


Figure 5: Architecture of Disclosure

Source Figure 1: https://doi.org/10.1145/2420950.2420969

Proof of Concept

Domain Classifier using DNS

- Features:
  - DNS
    - # NS, MX, TXT Records
  - WHOIS
    - Domain Registration Period
    - Domain Name Relative Entropy
- Example DGA[1]:
  - WWW.XN--ZALGO003446-SJGB60AIGHL2I8JC3B0A2A97FTBLL0CZA.COM
  - WWW.XN--ZALGO012841-SJGB60AIGHL2I8JC3B0A2A97FTBLL0CZA.COM
- Machine Learning: Relevance Vector Machine

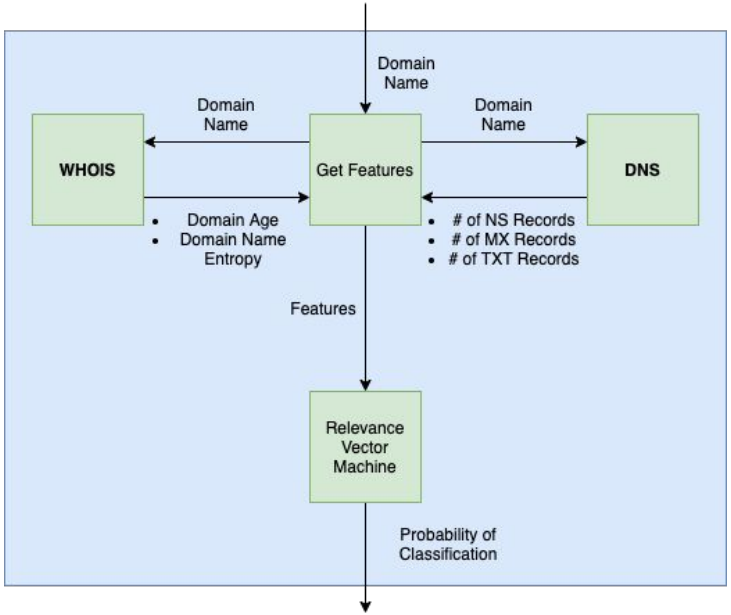


Figure 6: PoC Domain Classifier Architecture

1: http://go.cybereason.com/rs/996-YZT-709/images/Cybereason-Lab-Analysis-Dissecting-DGAs-Eight-Real-World-DGA-Variants.pdf

Experiments

- Evaluating the Features of the Domain Classifier using DNS

- Evaluating the PoC
  - Domain Classifier using DNS
  - Disclosure
  - The Whole PoC System

Creating Domain Lists

- Created Two Domain Lists
- Benign Domains lists:
  - List of Top 1 million domains from Majestic[1]
- Malicious Domains List:
  - List with combination of Recent[2] (~ max 2 weeks) and Basic Spam Domains[3] from Joewein
- Only domains that are still online are included in the list
- Domains with certain TLDs were excluded

1: http://downloads.majestic.com/majestic_million.csv
2:https://joewein.net/dl/bl/dom-bl.txt
3: https://joewein.net/dl/bl/dom-bl-base.txt

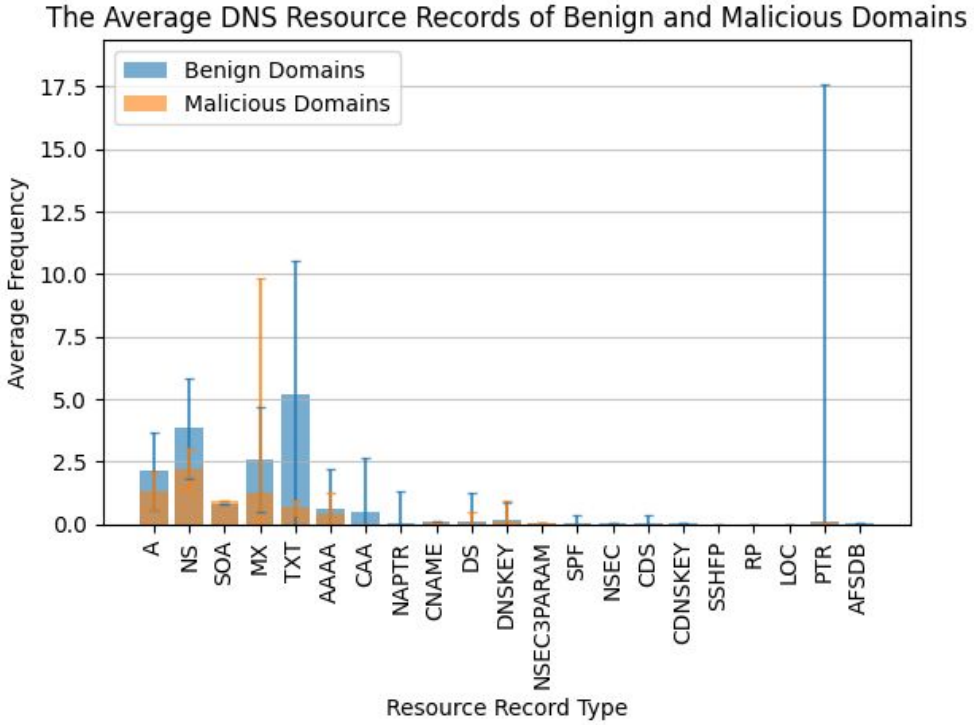
Training the PoC

- Train NetFlow system with CTU-13 dataset[1]
  - 100k NetFlow records
  - 2% malicious
- Train DNS classifier with domain lists
  - 1000 malicious and 1000 benign domains
- Prevent over/underfitting by calculating the accuracy on both the training set and evaluation set

1: https://www.stratosphereips.org/datasets-ctu13

Evaluation DNS features

- Benign domains usually have more:
  - NS records
  - TXT records
  - MX records



Evaluation WHOIS Domain Registration Period Feature

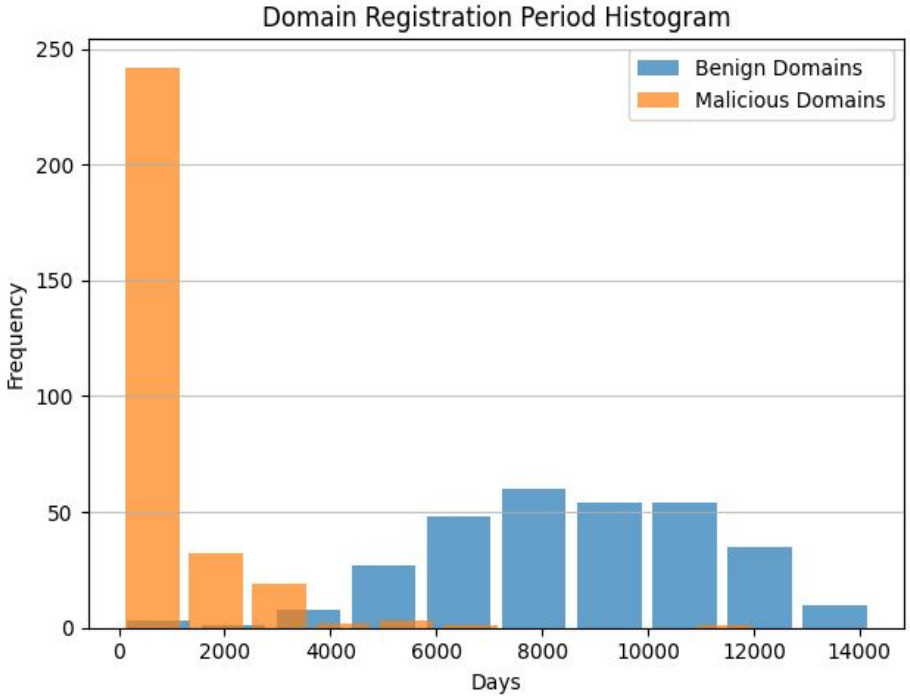
Methodology

Results

- Malicious domains usually have **lower** registration Periods
- Clear distinction between malicious and benign domains

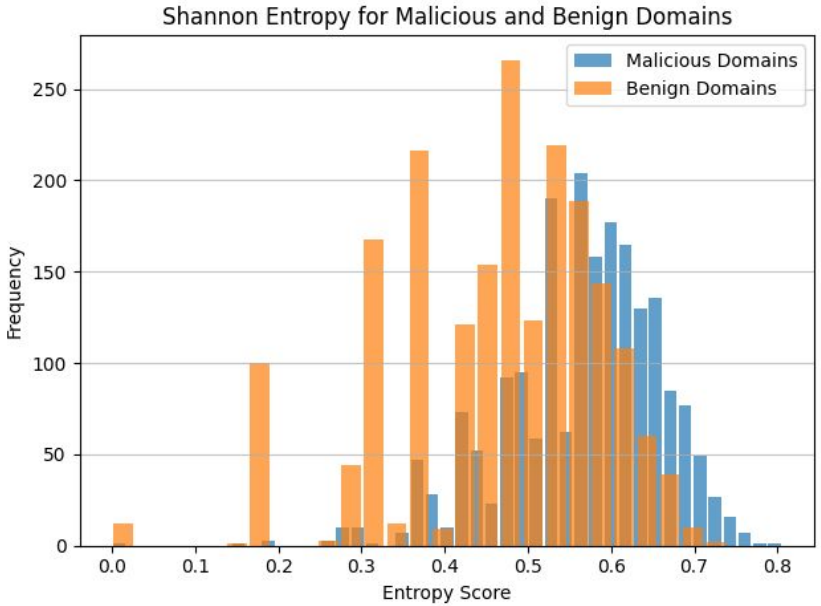
Discussion

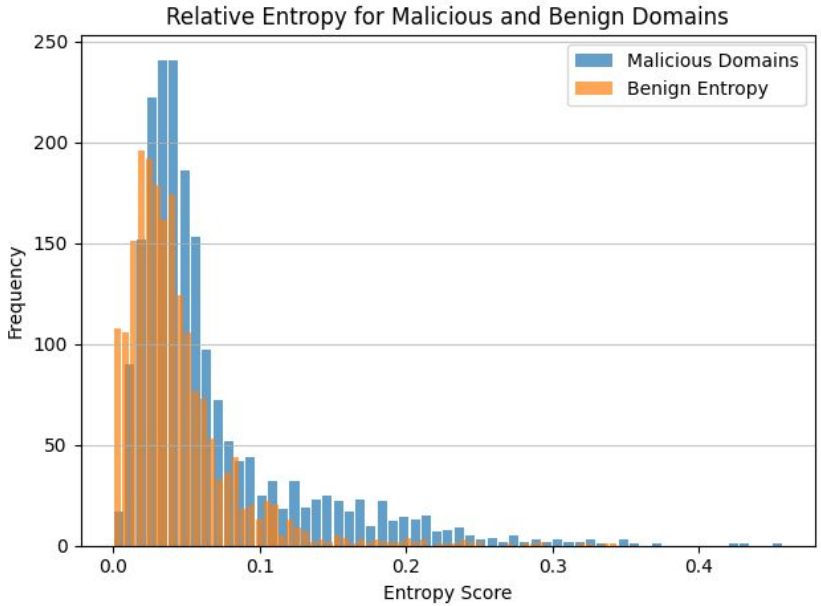
- Benign domains: Top 300 most popular websites



Evaluation WHOIS Domain Name Entropy Feature

Methodology





Evaluating PoC

- www.malware-traffic-analysis.net
- Both background and malicious traffic
- DNS system
- Disclosure
- PoC

Results

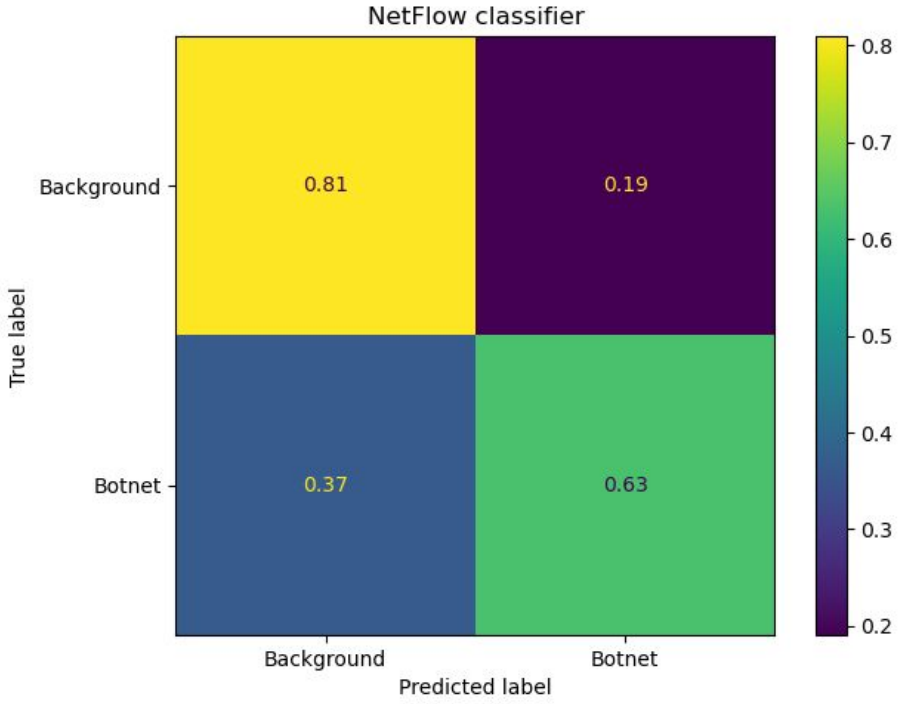
Domain Classifier Evaluation Results

- Accuracy: 97%

	True (Detection)	False (Detection)
True (Reality)	31 (TP)	7 (FN)
False (Reality)	1 (FP)	320 (TN)

Netflow Evaluation Results

- n = 10.000
- Accuracy: 71%



PoC Evaluation Results

- Accuracy: 99%

	True (Detection)	False (Detection)
True (Reality)	40 (TP)	3 (FN)
False (Reality)	0 (FP)	415 (TN)

Discussion PoC

- Limited test dataset
- NetFlow detection only works known botnets
- Limitations
  - CDNs
  - WHOIS

Future Work

- Evaluate system in a real-world environment
- Evaluate system with larger datasets
- Evaluate disclosure features
- Other features
  - Registrars
  - BGP ASN

Conclusion

How can malicious traffic to and from transient Command and Control servers be detected using DNS and NetFlow data?

- Combining two systems into one
    - One system using DNS and one system using NetFlow
- At least one system must be accurate
- Good DNS features: # of {NS, MX, TXT}, Domain Registration Period, Domain Name Relative Entropy
- Usable Netflow features: Flow size, Client Access Patterns, Temporal Behaviour

Questions?