Effective DGA Family Classification using a Hybrid Shallow and Deep Packet Inspection Technique on P4 Programmable Switches

Ali AlSabeh[1], Kurt Friday[2], Jorge Crichigno (Presenter)[1], Elias Bou-Harb[2]

[1] University of South Carolina, SC
[2] The University of Texas at San Antonio, TX

IEEE International Conference on Communications (IEEE ICC)
May 30, 2023 - Rome, Italy

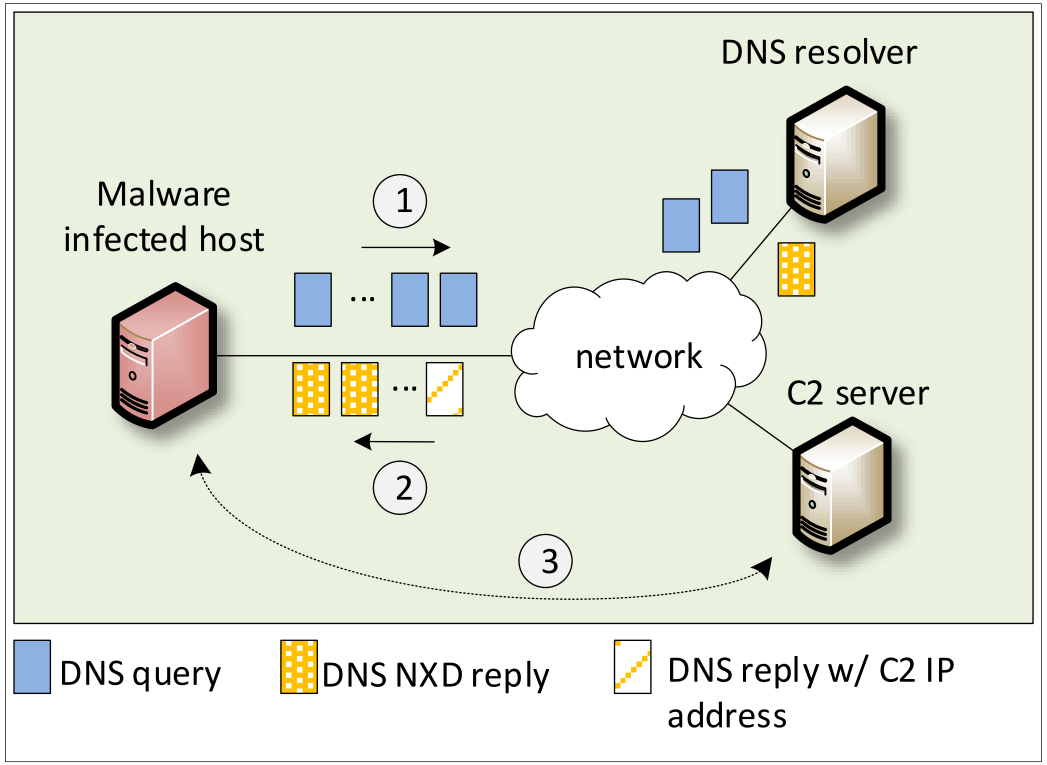
Introduction

- Attackers often use a Command and Control (C2) server to establish communication between infected host/s and bot master
- Domain Generation Algorithms (DGAs) are the *de facto* dynamic C2 communication method used by malware, including botnets, ransomware, and many others[1]

[1] "Dynamic Resolution: Domain Generation Algorithms." [Online]. Available: https://tinyurl.com/44hz9hpm.

DGA Attacks

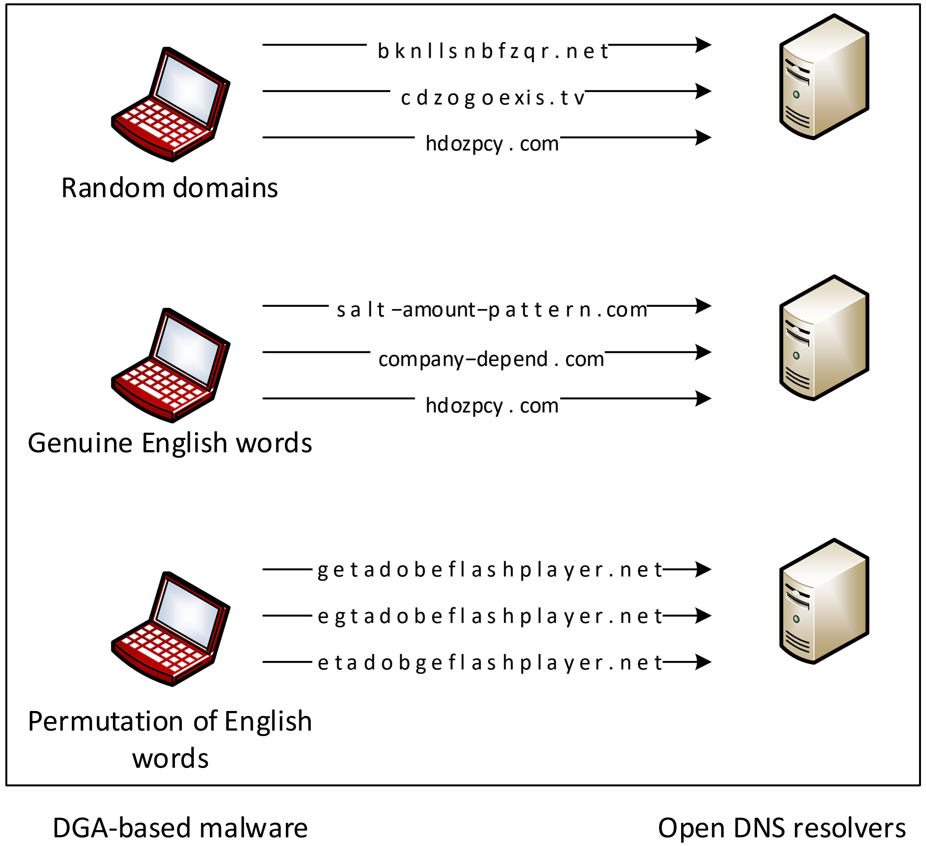
- DGAs evade firewall controls by frequently changing the domain name selected from a large pool of candidates
- The malware makes DNS queries to resolve the IP addresses of these generated domains
- Only a few of these queries will be successful; most of them will result in Non-Existent Domain (NXD) responses



(1) DNS queries. (2) (NXD) replies. (3) Eventually, a query for the actual domain is sent and malware-C2 communication starts.

DGA Attacks

- DGAs evade firewall controls by frequently changing the domain name selected from a large pool of candidates
- The malware makes DNS queries to resolve the IP addresses of these generated domains
- Only a few of these queries will be successful; most of them will result in Non-Existent Domain (NXD) responses



Existing Mitigation Strategies

- Approaches rely on contextual network traffic analysis (context-aware) or domain name analysis, without considering network traffic (context-less)
- Most research efforts focus on DGA detection, i.e., they perform binary classification in order to segregate DGAs from benign traffic
- In addition to DGA detection, it is helpful to classify DGA malware based on the family (Trojan, Backdoor, etc.)

Motivation

- Context-aware approaches analyze the network traffic behavior to fingerprint DGAs
    - Slow since they typically analyze batches of traffic offline
- Domain-name (context-less) approaches obtain high accuracy with ML models
    - The use of a general-purpose CPU/GPU may create a bottleneck due to high traffic volume
- There is a need for a system that uses both context-aware and context-less features to classify DGAs

Contribution

- Proposing a novel P4 scheme that uses a hybrid context-aware and context-less feature extraction technique entirely in the data plane
- Implementing Deep Packet Inspection (DPI) on Intel's Tofino ASIC that extracts and analyzes domain names within 3 microseconds
- Evaluating the proposed approach on 50 DGA families collected by crawling GBs of malware samples
- Highlighting the effectiveness of the proposed work in terms of accuracy, performance

Overview P4 Switches

- P4 switches permit the programmer to program the data plane
  - Customized packet processing
  - High granularity in measurements
  - Per-packet traffic analysis and inspection
  - Stateful memory processing

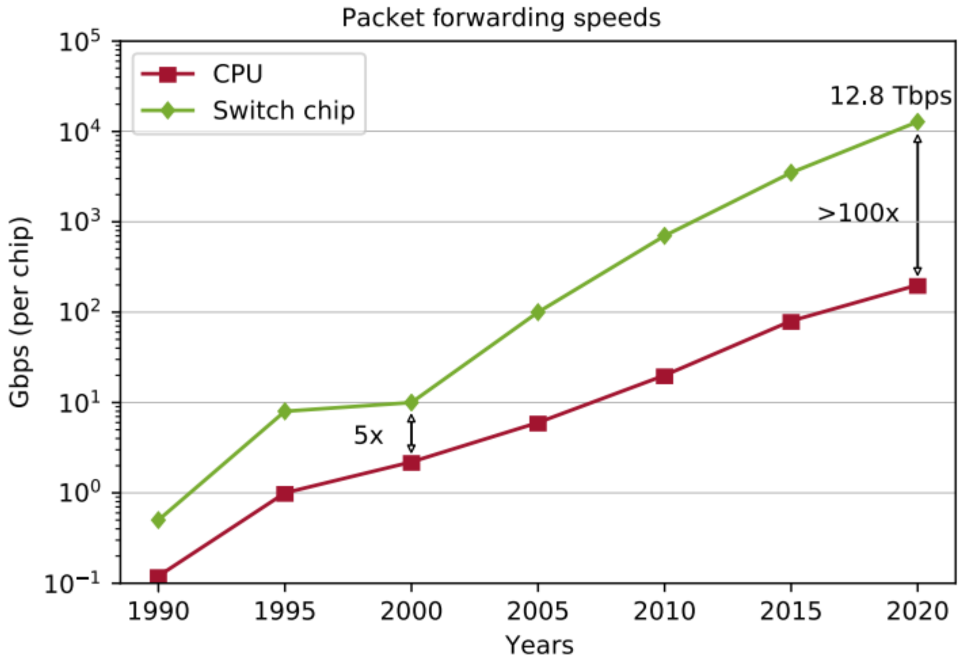
```
/*************************************************************************
*********************** P A R S E R  ***********************************
*************************************************************************/

    state parse_ethernet {
        packet.extract(hdr.ethernet);
        transition select(hdr.ethernet.etherType) {
            TYPE_IPV4: parse_ipv4;
            default: accept;
        }
    }

    state parse_ipv4 {
        packet.extract(hdr.ipv4);
        verify(hdr.ipv4.ihl >= 5, error.IPHeaderTooShort);
        transition select(hdr.ipv4.ihl) {
            5            : accept;
            default      : parse_ipv4_option;
        }
    }
```

P4 code

Programmable chip

Overview P4 Switches

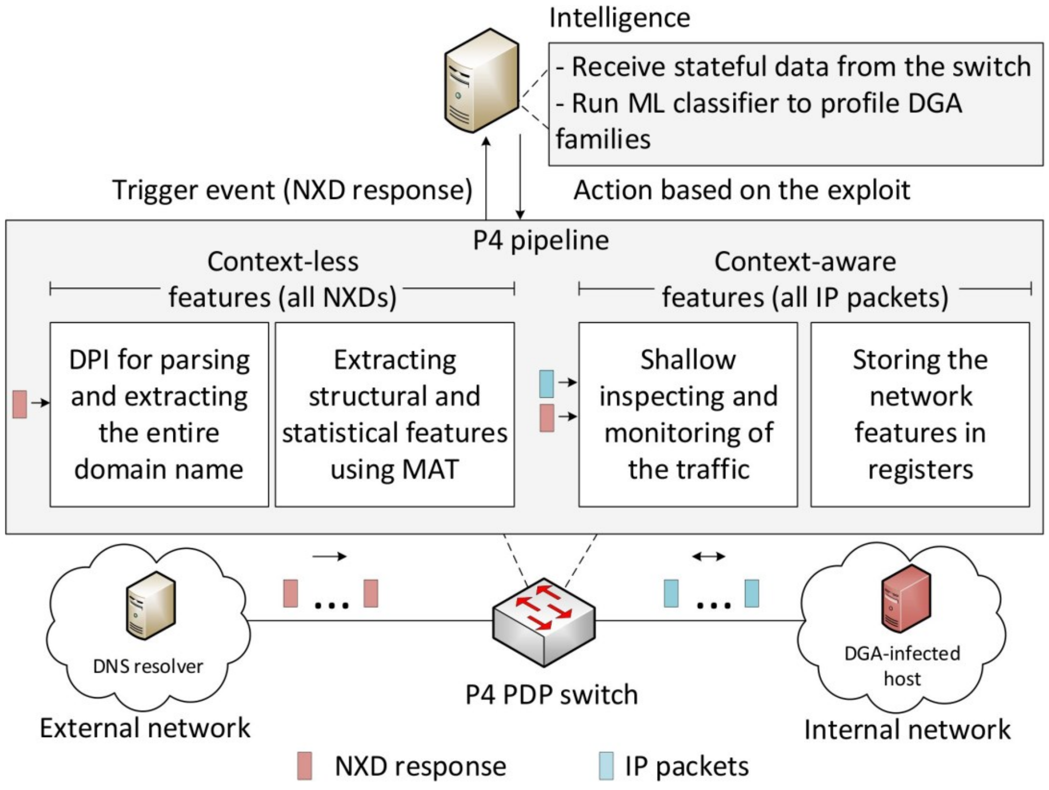
- P4 switches permit the programmer to program the data plane
  - ➢ Customized packet processing
  - ➢ High granularity in measurements
  - ➢ Per-packet traffic analysis and inspection
  - ➢ Stateful memory processing
- **If the P4 program compiles, it runs on the chip at line rate**



Reproduced from N. McKeown. Creating an End-to-End Programming Model for Packet Forwarding. Available: **https://www.youtube.com/watch?v=fiBuao6YZI0&t=4216s**

Proposed System

- The P4 PDP switch collects and stores the context-aware features of the hosts
- When an NXD response is received, the switch performs DPI on the domain name to extract domain features
- The switch sends the collected features to the control plane
- The control plane runs the intelligence to classify the DGA family and initiate the appropriate incidence response



Proposed System

- Context-aware features
  - For each host in the network, the following features are stored in the data plane:
    - Number of IP addresses contacted
    - Inter-arrival Time (IAT) between such IP packets
    - Number of DNS requests made
    - Time it takes for the first NXD response to arrive
    - IAT between subsequent NXD responses
  - Collected in the data plane

Proposed System

- Context-less features
  - It computes the bigram of the domain name; a bigram model may suffice to predict whether a domain name is a legitimate human readable domain

$$score\ (d) = \sum_{\forall\ subdomain\ s \in d} \left(\sum_{\forall\ bigram\ b \in s} f_s^b \right)$$

Where f_s^b is the frequency of the bigram b in the subdomain *s*

  - The frequency value of a bigram *b* is pre-computed and stored in a Match-Action Table (MAT)
  - The lower the score, the more random the domain name
  - Example: the bigrams of “google” are: “$g”, “go”, “oo”, “og”, “gl”, “le”, “e$”

Evaluation

- Dataset
  - Hundreds of GB of malware samples; 1,311 samples containing 50 DGA families
  - To collect DGA-based malware, only samples that receive NXD responses containing domain names generated by DGAs (based on DGArchive[1]) are considered
- Experimental setup
  - The collected dataset was used to train ML models offline on a general-purpose CPU
  - 80% of data was used for training and 20% for testing

[1] D. P LOHMANN, “DGArchive.” [Online]. Available: https://tinyurl. com/yc6whwrc.

Evaluation

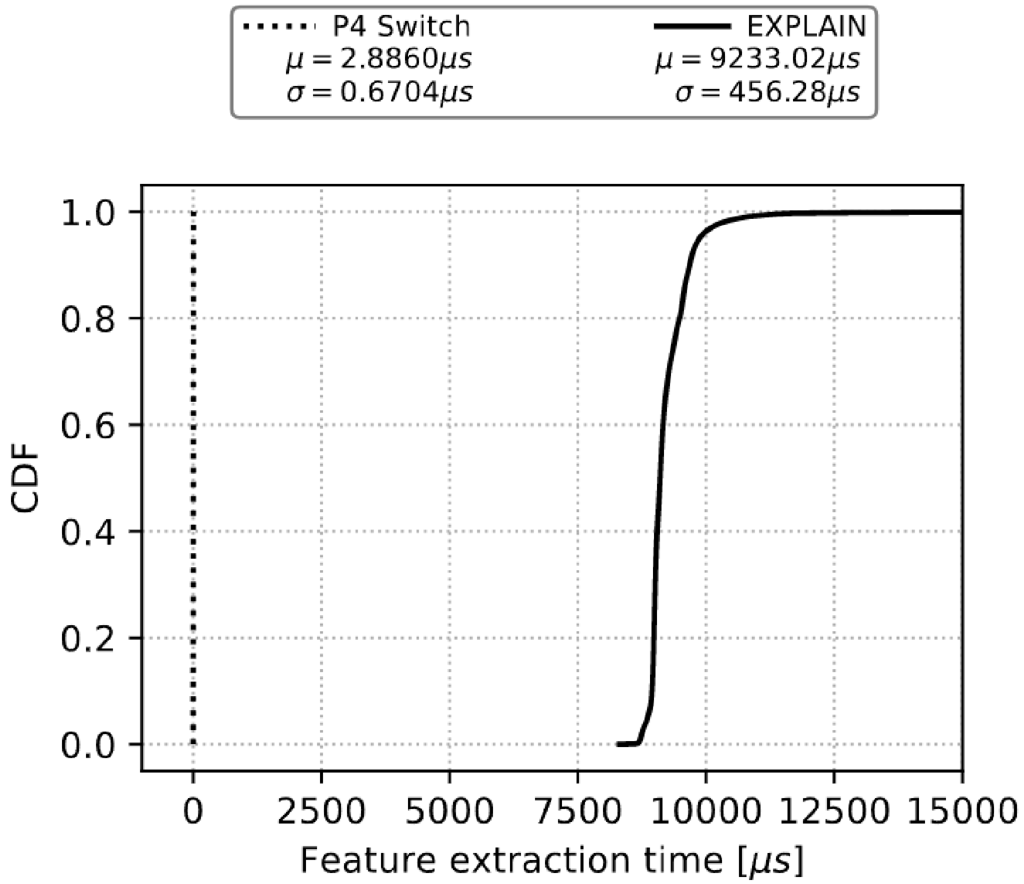
- Accuracy (Acc), F1 score, and Precision (Prec) of different ML classifiers during the first 8 NXD responses received were reported
- The Random Forest (RF) model performed best
  - ➢ The Accuracy (Acc) starts at 92% from the first NXD response received and reaches 98% by the 8th NXD response

NXD count	RF			SVM			MLP			LR			GNB		
	Acc	**F1**	**Prec**	**Acc**	**F1**	**Prec**	**Acc**	**F1**	**Prec**	**Acc**	**F1**	**Prec**	**Acc**	**F1**	**Prec**
NXD 1	0.923	0.907	0.902	0.872	0.856	0.847	0.87	0.843	0.829	0.716	0.679	0.667	0.726	0.688	0.688
NXD 2	0.951	0.943	0.943	0.899	0.893	0.893	0.904	0.897	0.9	0.76	0.741	0.747	0.727	0.701	0.707
NXD 3	0.964	0.958	0.964	0.918	0.913	0.914	0.924	0.914	0.912	0.767	0.74	0.743	0.649	0.668	0.732
NXD 4	0.966	0.961	0.963	0.906	0.905	0.912	0.916	0.909	0.915	0.79	0.765	0.758	0.633	0.635	0.692
NXD 5	0.97	0.966	0.967	0.915	0.91	0.911	0.919	0.91	0.907	0.77	0.735	0.746	0.604	0.615	0.689
NXD 6	0.975	0.972	0.973	0.914	0.911	0.912	0.922	0.915	0.918	0.794	0.767	0.783	0.617	0.627	0.716
NXD 7	0.977	0.976	0.979	0.92	0.915	0.915	0.929	0.924	0.93	0.799	0.771	0.78	0.61	0.613	0.714
NXD 8	0.98	0.979	0.981	0.917	0.912	0.914	0.93	0.923	0.921	0.764	0.73	0.735	0.631	0.618	0.65

RF: Random Forest; SVM: Support Vector Machine; MLP: Multilayer perceptron; LR: Logistic Regression; GNB: Gaussian Naive Bayes

Evaluation

- Feature extraction time of the proposed approach and EXPLAIN
- EXPLAIN's available source code was tested on a general-purposed CPU with 64 GB RAM, 2.9 GHz processor with 8 cores



Conclusion and Discussion

- In this work, we propose a hybrid feature extraction technique relying on context-aware and context-less features to classify DGA families
- Context-aware features characterize the network traffic behavior of the DGAs and require shallow packet inspection (no degradation to the throughput)
- Context-less features study the statistical and structural characteristics of the domain names relating to NXDs using DPI
- With 50 DGA families analyzed, the proposed approach achieves 92% accuracy with RF classifier from the first NXD response and reaches up to 98% by the 8th NXD response
- We plan to explore other techniques that are robust against encrypted DNS traffic, in addition to collecting more DGA families

Acknowledgement

- Thanks to the National Science Foundation (NSF)
- Activities in the CI Lab at the UofSC are supported by NSF, Office of Advanced Cyberinfrastructure (OAC), awards 2118311 and 2104273

References

[1] M. Grill, I. Nikolaev, V. Valeros, and M. Rehak, “Detecting DGA Malware using NetFlow,” in 2015 IFIP/IEEE International Symposium on Integrated Network Management (IM), pp. 1304–1309, IEEE, 2015.

[2] Y. Iuchi, Y. Jin, H. Ichise, K. Iida, and Y. Takai, “Detection and Blocking of DGA-Based Bot Infected Computers by Monitoring NXDOMAIN Responses,” in 2020 7th IEEE International Conference on Cyber Security and Cloud Computing (CSCloud)/2020 6th IEEE International Conference on Edge Computing and Scalable Cloud (EdgeCom), pp. 82– 87, IEEE, 2020.

[3] L. Bilge, S. Sen, D. Balzarotti, E. Kirda, and C. Kruegel, “Exposure: A passive DNS Analysis Service to Detect and Report Malicious Domains,” ACM Transactions on Information and System Security (TISSEC), vol. 16, no. 4, pp. 1–28, 2014.

[4] S. Schuppen, D. Teubert, P. Herrmann, and U. Meyer, “FANCI: Feature-based Automated NXDomain Classification and Intelligence,” in 27th USENIX Security Symposium (USENIX Security 18), pp. 1165– 1181, 2018.

[5] L. Fang, X. Yun, C. Yin, W. Ding, L. Zhou, Z. Liu, and C. Su, “ANCS: Automatic NXDomain Classification System Based on Incremental Fuzzy Rough Sets Machine Learning,” IEEE Transactions on Fuzzy Systems, vol. 29, no. 4, pp. 742–756, 2020.

[6] K. Highnam, D. Puzio, S. Luo, and N. R. Jennings, “Real-time Detection of Dictionary DGA Network Traffic Using Deep Learning,” SN Computer Science, vol. 2, no. 2, pp. 1–17, 2021.

[7] B. Yu, D. L. Gray, J. Pan, M. De Cock, and A. C. Nascimento, “Inline DGA Detection with Deep Networks,” in 2017 IEEE International Conference on Data Mining Workshops (ICDMW), pp. 683–692, IEEE, 2017.

[8] A. Drichel, N. Faerber, and U. Meyer, “First Step Towards Explainable DGA Multiclass Classification,” in The 16th International Conference on Availability, Reliability and Security, pp. 1–13, 2021.

[9] T. A. Tuan, H. V. Long, and D. Taniar, “On Detecting and Classifying DGA Botnets and their Families,” Computers & Security, vol. 113, p. 102549, 2022.

Thank You

- Contact info for further questions
  - aalsabeh@email.sc.edu
  - jcrichigno@cec.sc.edu

- CyberInfrastructure Lab (CI Lab) website
  - http://ce.sc.edu/cyberinfra/