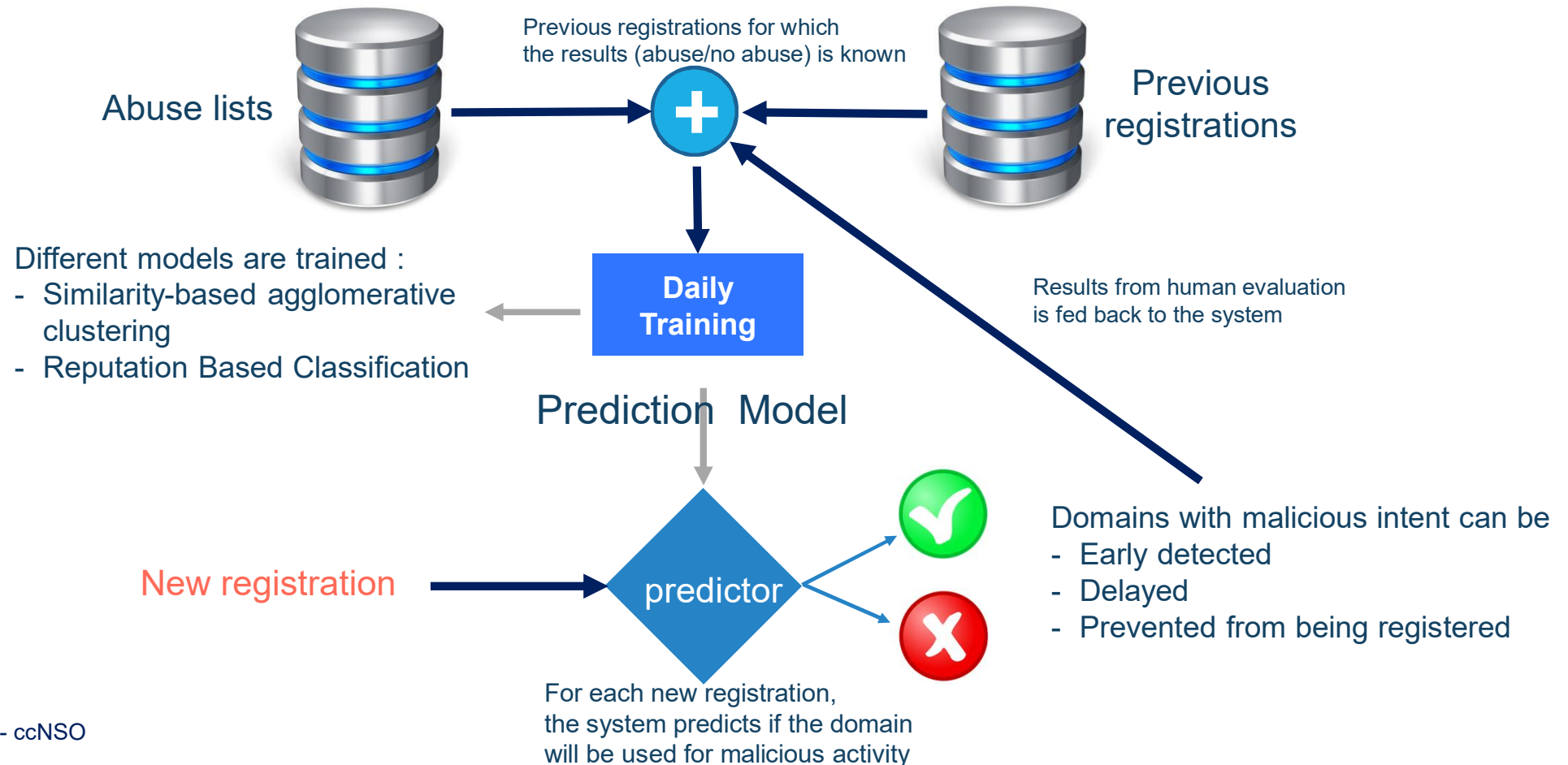


Abuse Prevention and Early Warning System (APEWS)

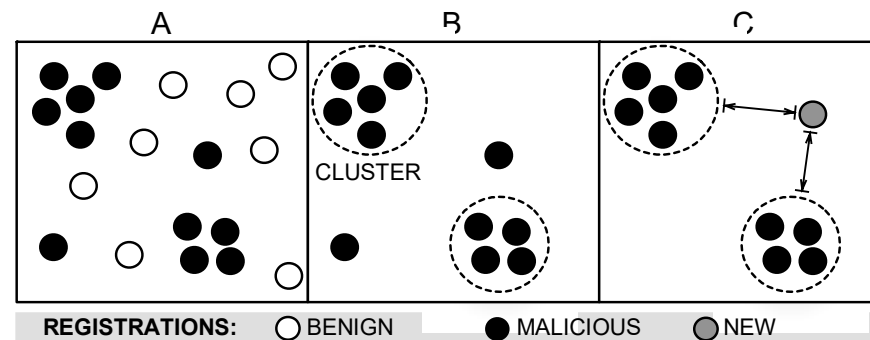
Predictive Model

Objective : Predict at time of registration whether a DN will be used abusively



Similarity Based Clustering

- Rationale : Domains belonging to the same campaign have very similar registration data
- For all malicious registrations in the past period, the similarity with other malicious registrations is calculated and expressed as a metric
- Based on the inter-registration similarity, registrations are clustered into clusters of 'very similar' registrations, i.e. 'campaigns'
- For each new registration, the distance to the malicious clusters is calculated



Results test phase

		Prediction	
		Abuse	No Abuse
Reality	Abuse	True Positives (TP)	False Negatives (FN)
	No Abuse	False Positives (FP)	True Negatives (TN)

Results

$$Accuracy = \frac{TP + TN}{TP + FP + TN + FN}$$

$$Recall = \frac{TP}{TP + FN}$$

$$Precision = \frac{TP}{TP + FP}$$

$$False\ Positive\ Rate = \frac{FP}{FP + TN}$$

How many did we find ?
(of the category we were looking for)

How many were correct ?
(of those we predicted as a hit)

How many were incorrectly
classified as a hit ?
(of those that were not abusive)

Optimization

What is most important ?

- Find all the cases (recall \uparrow) with low precision ?
- Predict correctly (precision \uparrow) and miss a lot of cases ?
- As accurate as possible ?

Results test phase

	TP	FP->TP	FP	TN	FN	Recall	Prec.	FPR
10/01/2019 - 02/01/2019	64	254	248	28045	60	84.13%	56.18%	0.88%
02/06/2018 - 10/01/2019	1575	3919	1311	334821	1759	75.75%	80.73%	0.39%
02/04/2018 - 20/06/2018	1996	1301	488	93023	378	89.71%	87.11%	0.52%
28/03/2018 - 24/04/2018	643	1085	222	37504	140	92.51%	88.62%	0.59%
10/01/2018 - 28/03/2018	4055	24	1089	80551	867	82.47%	78.93%	1.33%

$$Accuracy = \frac{TP + TN}{TP + FP + TN + FN}$$

How accurate is our prediction ?

$$Recall = \frac{TP}{TP + FN}$$

How many did we find ?
(of the category we were looking for)

$$Precision = \frac{TP}{TP + FP}$$

How many were correct ?
(of those we predicted as a hit)

$$False\ Positive\ Rate = \frac{FP}{FP + TN}$$

How many were wrong ?
(on total benign)

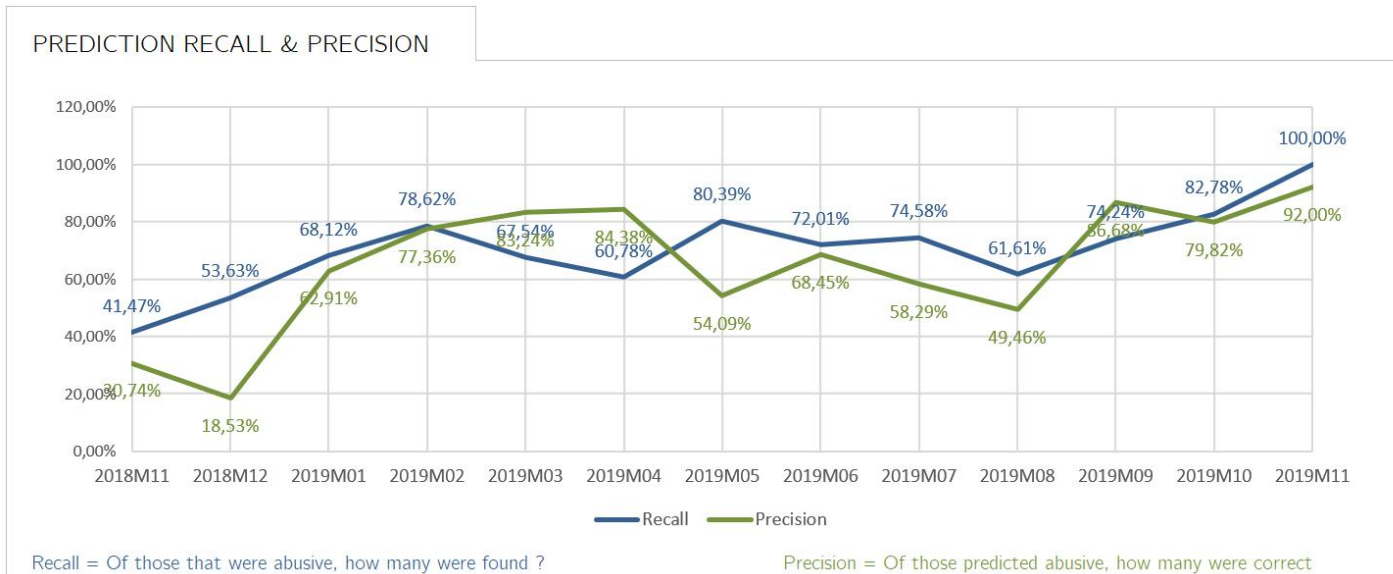
Average

TPR : **82.32%**
(pct reported abuses found)

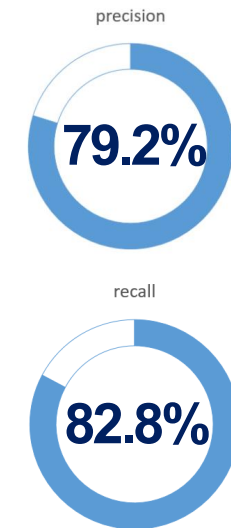
Precision: **81.62%**
(pct correct on predicted abuses)

FPR : **0.58%**
(abuses predicted on total benign)

Production phase (no delay)



RESULTS OKT 2019



$$\text{Recall} = \frac{TP}{TP + FN}$$

How many did we find ?
(of the category we were looking for)

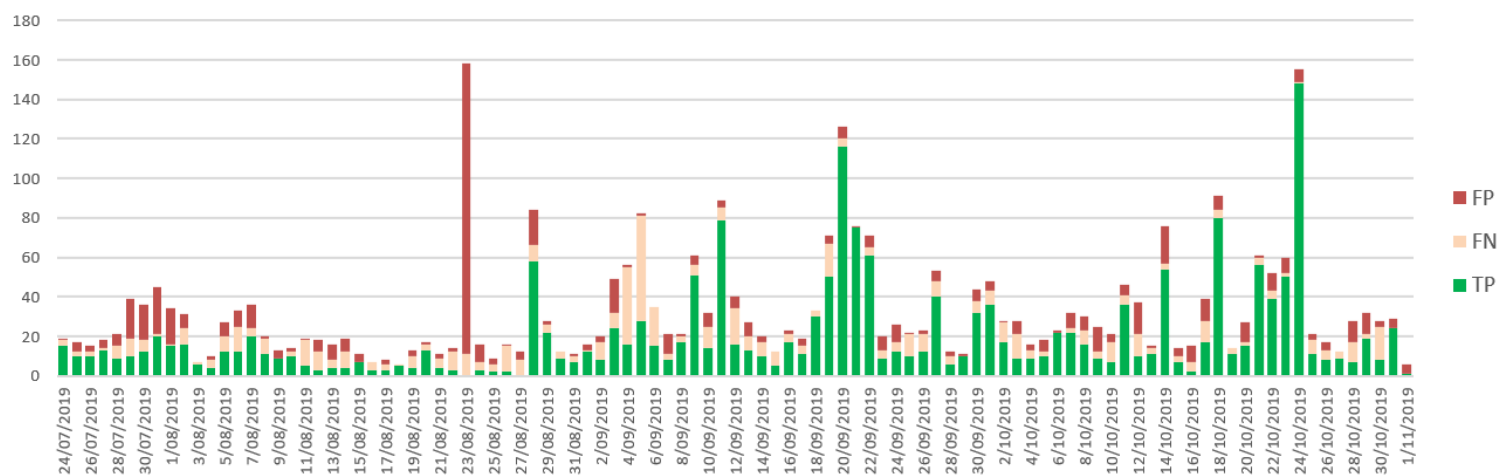
$$\text{Precision} = \frac{TP}{TP + FP}$$

How many were correct ?
(of those we predicted as a hit)

What is most important ?

- Find all the cases (recall ↑) with low precision ?
- Predict correctly (precision ↑) and miss a lot of cases ?
- As accurate as possible ?

PREDICTION CORRECTNESS (2)



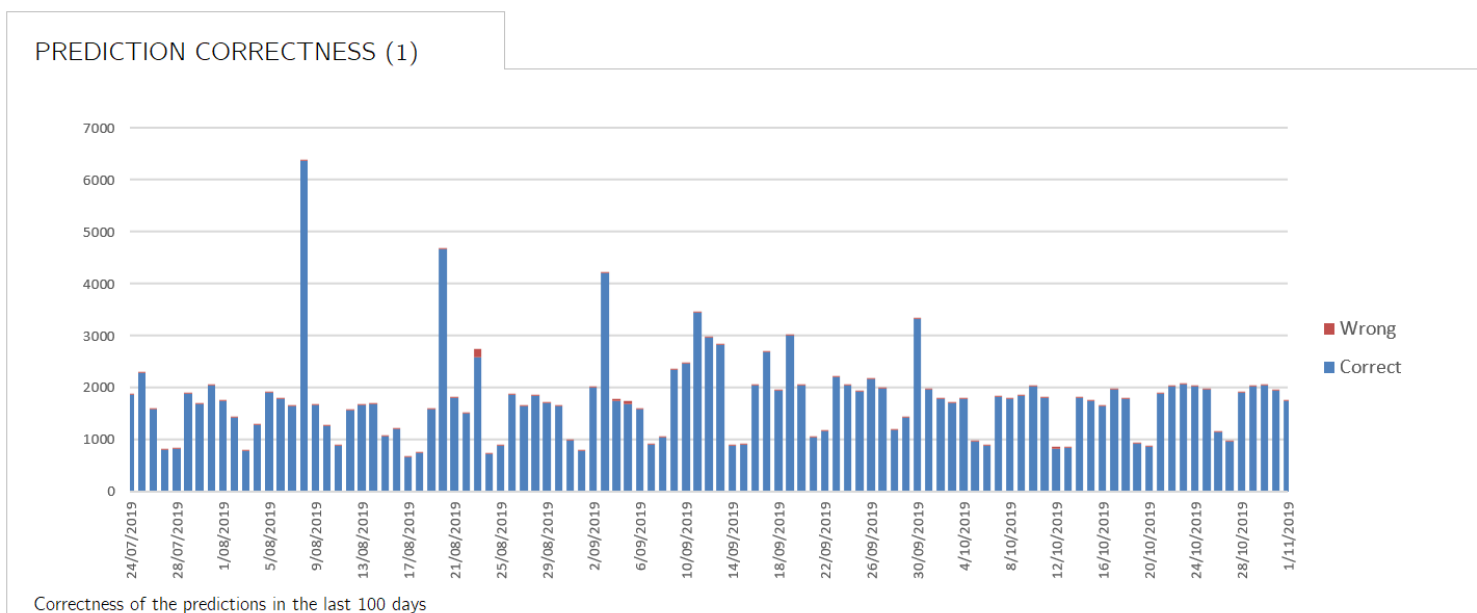
TP : Nbr of DNS that were correctly predicted as abusive in the last 100 days

FN : Nbr of DNS that were incorrectly predicted as not abusive in the last 100 days (= missed cases)

FP : Nbr of DNS that were incorrectly predicted as abusive in the last 100 days (= wrongly delayed)

Note that the FP may still turn out to be TP in the future. It just means that at the time of the report, they were not yet captured as abusive by the monitoring systems.

The Accuracy trap

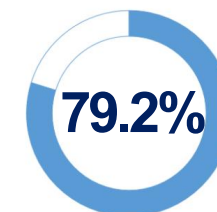


Pct of the prediction that was correct : 99.33%

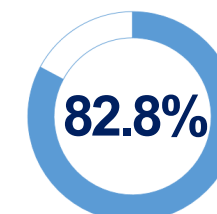
But ... if we would always predict *no abuse*, accuracy would be 98.53% !
Typical for unbalanced data.

RESULTS OKT 2019

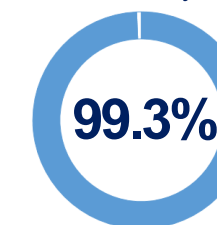
Precision



Recall



Accuracy



Effectiveness

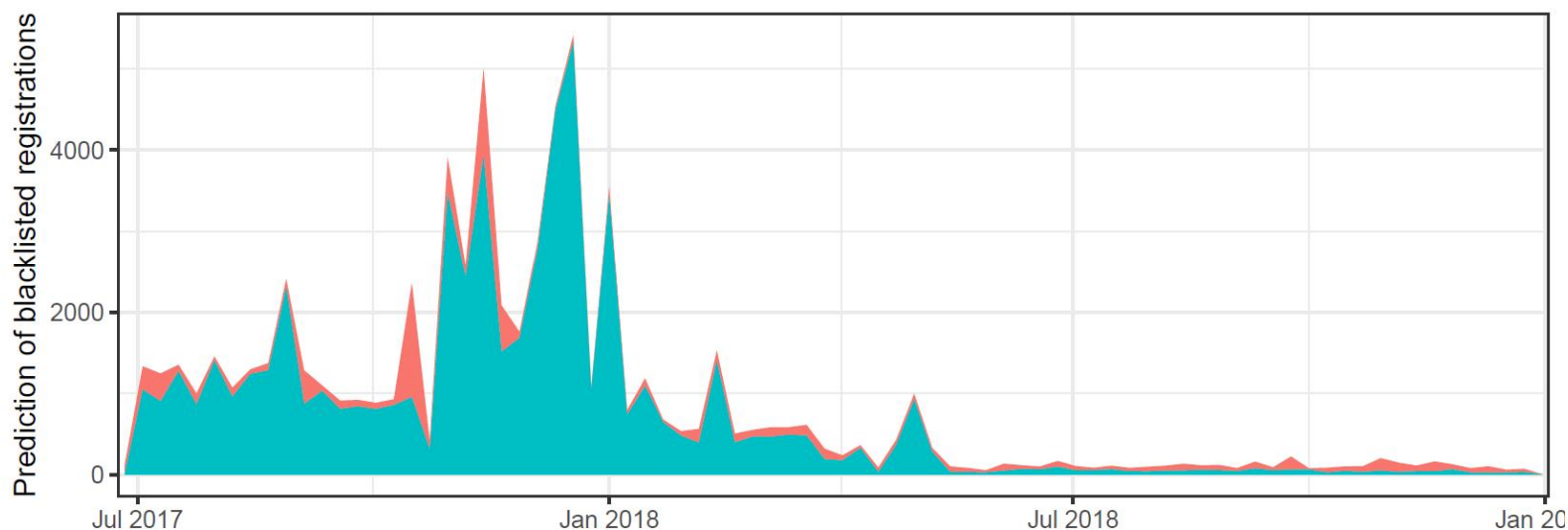


Figure 8: The weekly prediction of blacklisted registrations for the selected ensemble predictor during operations. The red area plots the total number of blacklisted registrations on that week, whereas the green area represents the predictions.

Delayed Delegation

Predict at time of registration whether a DN will be used abusively

Status :

- Running in production without delayed delegation
- Currently 80% Recall and 80% Precision

Next Steps :

- Improve algorithms (add categorisation)
- Explore to include other abuse lists
- **Start delaying**



More information

Exploring the ecosystem of malicious domain registrations in the .eu TLD

Thomas Vissers¹, Jan Spooren², Pieter Agten³, Dirk Jaumart², Peter Jaansen², Marc Van Wesemael², Frank Pessens², Wouter Joosen², and Lieven Desmet¹

¹ imec-DistriNet, KU Leuven, Belgium
{firstname.lastname}@cs.kuleuven.be
² EURid VZW, Belgium
{firstname.lastname}@eurid.eu

Abstract. This study extensively examines 11 months of registration data to identify large-scale malicious campaigns present in the .eu TLD. We explore the ecosystem and motives operators of elaborate cybercriminal entities that repeatedly register large amounts of domains for one-shot, malicious use. Although these malicious domains are short-lived, by incorporating registrant information, we establish that at least 90.6% of them can be traced to 20 large campaigns with varying duration and intensity. We further report on insights in the operational aspects of this business and observe, amongst other findings, that their processes are only partially automated. Finally, we apply a post-factum clustering process to validate the campaign identification process and to automate the ecosystem analysis of malicious registrations in a TLD zone.

Keywords: malicious domain names, campaigns, DNS security

1 Introduction

The Domain Name System (DNS) is one of the key technologies that has allowed the web to expand to its current dimensions. Virtually all communication on the web requires the resolution of domain names to IP addresses. Malicious activities are no exception, and attackers constantly depend upon functioning domain names to execute their abusive operations. For instance, phishing attacks, distributing spam emails, botnet command and control (C&C) communications and malware distribution: these activities all require domain names to operate.

Widely-used domain blacklists are created and used to stop malicious domain names¹ shortly after abusive activities have been observed and reported. As a consequence, attackers changed to a hit-and-run strategy, in which malicious domain names are operational for only a very small time window after the initial registration, just for a single day in 60% of the cases [1]. Once domain names

¹We use the term malicious domain name whenever we refer to a domain name that is registered to be bound to a malicious service or activity.

Detection of Algorithmically Generated Domain Names used by Botnets: A Dual Arms Race.

Jan Spooren¹, Davy Preuvencers¹, Lieven Desmet¹
imec - DistriNet - KU Leuven imec - DistriNet - KU Leuven imec - DistriNet - KU Leuven
Hervele, Belgium Hervele, Belgium Hervele, Belgium
jan.spooren@cs.kuleuven.be davy.preuvencers@cs.kuleuven.be lieven.desmet@cs.kuleuven.be

Peter Jaansen², Wouter Joosen²
EURid VZW, Belgium imec - DistriNet - KU Leuven
Diegem, Belgium Hervele, Belgium
Peter.Jaansen@eurid.eu wouter.joosen@cs.kuleuven.be

ABSTRACT. Malware typically uses Domain Generation Algorithms (DGAs) as a mechanism to contact their Command and Control server. In recent years, different approaches to automatically detect generated domain names have been proposed, based on machine learning. The first problem that we address is the difficulty to systematically compare these DGA-detection algorithms due to the lack of an independent benchmark. The second problem that we investigate is the difficulty for an adversary to circumvent these classifiers when the machine learning model backing these DGA-detectors are known. In this paper we compare two different approaches on the same set of DGAs: classical machine learning using manually engineered features and a ‘deep learning’ recurrent neural network. We show that the deep learning approach performs consistently better on all of the tested DGAs, with an average classification accuracy of 95.7% versus 93.9% for the manually engineered features. We also show that one of the dangers of manual feature engineering is that DGAs can adapt their strategy based on knowledge of the feature used to detect them. To demonstrate this, we use the knowledge of the used feature set to design a new DGA, which makes the random forest classifier powerless with a classification accuracy of 59.9%. The deep learning classifier is also (abit less) affected, reducing its accuracy to 83.5%.

CCS CONCEPTS

• Security and privacy → Malware and its mitigation, • Computing methodologies → Neural networks, Classification and Regression trees

KEYWORDS

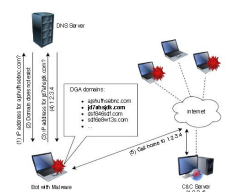
Malware Detection, Domain Generation Algorithms, Machine Learning

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyright for this work, unless otherwise stated, falls under the Creative Commons Attribution 4.0 International license. For more information, contact the publisher or the copyright owner. For all other uses, permission should be sought from ACM. For Request Permissions from permissions.acm.org. ACM 2019. Copyright is held by the author(s). Publication rights licensed to ACM. ACM ISBN 978-1-4503-5978-2/19/0000...\$15.00
© 2019 Association for Computing Machinery
ACM ISBN 978-1-4503-5978-2/19/0000...\$15.00
https://doi.org/10.1145/3257819.3257847

ACM Reference Format:
Jan Spooren, Davy Preuvencers, Lieven Desmet, Peter Jaansen, and Wouter Joosen. 2019. Detection of Algorithmically Generated Domain Names used by Botnets: A Dual Arms Race. In The 2019 ACM/IEEE Symposium on Applied Computing (SAC '19), April 8–12, 2019, Limassol, Cyprus. ACM, New York, NY, USA, Article 4, 8 pages. https://doi.org/10.1145/3257819.3257847

1 INTRODUCTION

The Internet connects billions of devices, ranging from servers and personal computers to tablets, mobile phones, household appliances, and many more. Malicious actors are constantly scanning the Internet for vulnerable devices which could be compromised, or are tricking users into unknowingly installing malware on their devices. Once this malware is present on a machine, it can be used to attack other machines, send unsolicited or phishing emails, oversteer an communication, steal e-mail addresses, encrypt the contents of the machine originating from the user a ransom for the ability to decrypt, and many more malicious schemes. Large pools [17] of infected machines, called botnets [4] exist, which are controlled from Command and Control (C&C) servers (as depicted in Figure 1).



Thanks

marc.vanwesemael@eurid.eu