A Survey of DNS Tunnelling Detection Techniques Using Machine Learning

Shiraz Yassine, Jawad Khalife, Maroun Chamoun, Hussein el Ghor

[#]*L'Institut National des Télécommunications ET de l'information, Faculty of Engineering, Saint Joseph University Beirut, Lebanon*

***Abstract*— The Domain Name System (DNS) is an essential network service translating human-friendly host names into numerical IP addresses. Prior to almost any network communication, a communication with a DNS server is, the most likely, needed. For this reason, DNS cyber-attacks are now one of the most challenging threats in the information security community due to its wide availability and the fact that it's not monitored in terms of security - not intended for data transfer.**

Particularly, DNS tunnelling embedding data in DNS queries and response is receiving a lot of attention in the research field over the last years. Recent studies have focused on DNS tunnelling detection using machine learning.

The aim of this paper is to provide a comprehensive survey of some different techniques proposed recently in the literature for detecting DNS tunnels using machine learning, while highlighting on the main findings and comparing their obtained results.

***Keywords*— Domain Name System, Cyber-attacks, Tunnelling detection, Machine Learning.**

I. INTRODUCTION

DNS translates easy memorized domain names to numerical IP addresses which is an essential service related to network and Internet Functionality. For this purpose, DNS protocol uses special message formats and types, like queries and replies. DNS and communicate on port 53 using usually UDP and TCP when the request is larger than 512 octets. RFC 1035 [1]

To determine the requested services (web pages, mail servers…), 83 DNS record types (2016) can be used. Common DNS records include: A, PTR, MX, CNAME, TXT, NS, and SOA records.

A DNS server can be authoritative – holding the DNS information - for one zone (example: domain.com) or it can be a local DNS cache serving client DNS queries. DNS queries are of two types [2]: (i) Recursive: recursion is when a DNS server query other DNS server on behalf of original DNS client for name resolution; (ii) Iterative: Forwarded to authoritative servers starting with ROOT servers. Each server refers the client to the next server in the chain, until the current server can fully resolve the request. So, the resolution of www.exampledomain.com would query a global root server, then the top-level domain "com" server and finally the "exampledomain.com" server.

From a security perspective, DNS stands out among most protocols for covert channels for several reasons.

First, because DNS is not intended for data transfer, DNS traffic is often allowed without being inspected by network security devices and almost ignored in network security policies, which makes DNS a prone for attacks and misuse.

Second, DNS includes some flexible fields used by attackers like TXT record and other.

In 1998 [3], Data transfer over DNS protocol has been discovered and was originally designed as a simple way to bypass the captive portals at the network edge and gain free Wi-fi access restricted access sites. Currently, transferring data over DNS poses a serious security risk to all organizations.

In 18 December 2017, The Etisalat UAE [4] headlined the news; the website was hacked, redirecting its users to a Chinese site through DNS tunnelling. The intent of the hacker was to steal user sensitive information. This attack shows that DNS can be used to attack well reputed organizations without referring to complex network protocols or advanced traffic obfuscation techniques.

The global DNS threat survey [5] covering three regions, has shown that the business sector is taking DNS tunnelling threats more into consideration where 38% of businesses are aware of data exfiltration through DNS (24% in 2016) but still more than the half are not aware of it. On the other hand, 22% of the organizations were affected with DNS tunnelling (11% in 2016).

The remaining of this paper is structured as follows. Section 2 deals with DNS tunnelling description, the way data can be exfiltrated and by which tools; Section 3 highlights DNS tunnelling detection techniques using machine learning. Section 4 compares the surveyed methods; and finally, section 5 finally outlines the conclusion.

II. DNS TUNNELLING

Tunnelling [3] allows transmission of data using a certain infrastructure encoding data of other programs and protocols in DNS queries and responses without alerting any firewalls or intrusion detection system. The original intention was to bypass captive portals in Wi-Fi hotspots at airports or hotels to acquire free internet access.

DNS tunnelling is a client-server model requiring a client to be compromised through malware, phishing or social engineering with the only requirement of access to internal

DNS server. At the infected DNS client level, a persistent backdoor with a DNS Tunnel will thus be established.

DNS tunnels can be used to exfiltrate important and confidential data from any organizations network (Data Exfiltration) or in form of Command and control channel [3] (C&C).

C&C is a communication channel between the target host and the command and control server. It embeds data and commands in DNS queries and responses. Also, it includes full remote access of the compromised host. In 2012 [3], at the RSA conference, it was one of the most dangerous cyber-attacks.

A lot of malware families have been discovered using DNS tunnelling to hide their communication: Morto [6], Feederbot [7], etc...

A. How does it work?

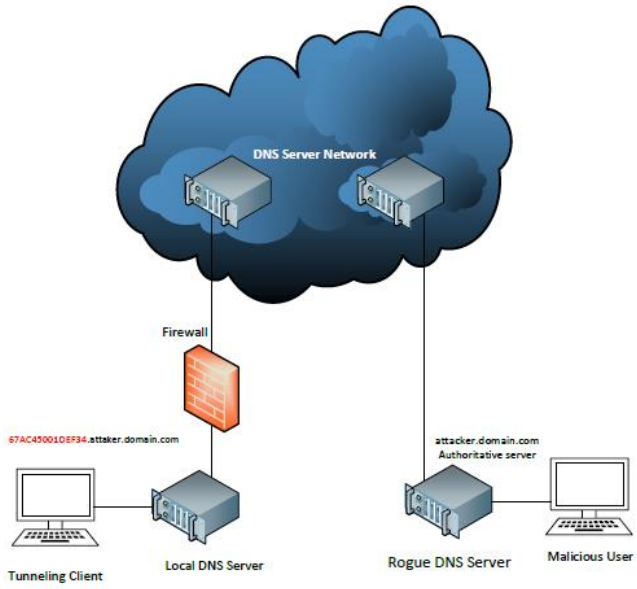

Figure 1.DNS tunnelling

As illustrated in [Fig. 1], DNS tunnelling requires a compromised client system to have external network connectivity and a Rogue DNS server controlled by the malicious user that can act as an authoritative server to execute the server-side tunnelling and data payload executable programs. After being infected by a malware, the DNS client starts issuing recursive DNS queries addressed to a domain name controlled by the threat actor. The local DNS server then forwards the queries iteratively to authoritative servers which should appear as normal to the local firewall. As shown in Figure 1, sensitive Data "67AC45001DEF34" can be easily exfiltrated through the DNS query itself back to the malicious user rogue DNS Server.

There are many tools [3], [8-10] used to embed data in DNS queries and responses between the tunnelled client and the rogue server that can then forward the data to another destination client.

B. Major DNS tunnelling tools

The Most commonly used DNS tools [7] are: DNS2tcp, tcp-over-DNS, OzymanDNS, Iodine, split brain, DNScat-P/DNScat2, DNScapy...

DNScat [3], released in 2004, is a java-based tool that allows two hosts to communicate routing all traffic through DNS.

Iodine [8], released in 2006, is a cross platform implementation of IPV4 tunnelling data through DNS server. It's written in C language and run on many environments such Linux, windows and others.

DNS2tcp [9] is a network tool able to encapsulate TCP packets over DNS tunnels. It's written in C and runs on Linux.

OzymanDNS [3] is a tool used to create a SSH tunnel over DNS or for file transfer.

Now that we highlighted on the DNS tunnelling technique and tools, detecting DNS tunnels seems to be a challenging task for researchers, as we will show in the next section.

III. DNS Tunnelling detection

As mentioned earlier, the most challenging concern in today's business is to keep ahead with the growing and changing security threats especially the massive rise in threats such "DNS Tunnelling".

DNA tunnelling detection techniques can be grouped, as per Franham [3], into two categories: Payload analysis and traffic analysis.

In payload analysis, the analysis will be for one or more requests and responses for tunnel Indicators. The attributes used are: size of request and response, entropy of hostnames, statistical analysis, uncommon record types and policy violation.

In traffic analysis, multiple requests and responses will be analysed over Time. Traffic attributes used here include among others: volume of DNS per IP address and per domain, number of hostnames per domain, geographic location of DNS server, domain history.

Recently, as a response toward the DNS tunnelling concern, researchers are tending to use Machine Learning Techniques (MLTs) to detect tunnelling. As mentioned earlier, MLTs will be highlighted the most and surveyed in this paper.

C. Machine learning

Machine learning is a subfield of artificial intelligence used to understand data structure and fit it into models that can be used by people. It allows computers to train on data inputs and statistical features.

Machine Learning is mostly used for an efficient tunnelling detection. It provides a way to define normal behaviour in a network, so it can detect anomalies that indicate the presence of DNS tunnels. Several MLTs exist: Support Vector Machine (SVM), Naïve Bayes (NB), Decision Tree (DT), K-nearest Neighbor (KNN) and others.

D. MLT Used for DNS Tunnel Detection

Different Machine Learning algorithms are used in the field of data science classified mainly into two categories: The Supervised learning and unsupervised learning.

The Supervised learning is where instances are given with known labels and it includes algorithms such logistic and linear regression, classification and support vector machine; with the latter one, the instances are unlabelled. A well-known algorithm in unsupervised learning is k-means clustering.

Maurizio Aiello et al. [11] show how basic classifiers of supervised learning are used to detect DNS tunnelling. His approach lies on Bayes classifier exploiting the statistical features of DNS Messages and detecting the presence of malicious data by analysing the entire set of DNS server exchanged information.

The performance evaluation shows that the approach is reliable and good results are obtained despite the simplicity of the mechanism.

In [12], the same work was enhanced by a monitoring mechanism using the same classifier that looks at statistical features of protocol message, such as packet inter-arrival times and of packet sizes instead of focusing on a single one and by reducing the classification time. As per the authors claim, the approach was reliable, robust and fast for DNS tunnelling detection.

Anirban Das et al. [13] addressed DNS tunnelling through a robust, end-to-end approach to deploy system for detecting malicious DNS activities.

"Logistic Regression" is the model used to detect data exfiltration with DNS tunnelling and "K-Means" for the tunnelling.

The 2 machine learning models show high detection and small false positive rate:

Logistic regression detects exfiltration with very small false positive rate of 0.189%

K-means detect Tunnelling with true positive rate of 91.68% and false positive rate of 0.40%.

Jingkun liu et al. [14] proposed a mechanism deployed on the recursive DNS using a set of features including: time-interval, request packet size, record type and subdomain entropy. The mechanism works in an off-line stage using labelled traffic to identify the existence of tunnelled traffic.

To compare the binary mechanism, the authors used 3 algorithms: Support Vector Machine (SVM), Logistical regression (LR), and Decision Tree (DT) and the results shows detection accuracy and precision of 99.96%.

Van Thuan Do et al. [15] addressed DNS tunnelling detection in mobile networks using machine learning.

Two methods have been selected: OCSVM (One Class Support Vector Machine) and K-Means. Beside the challenge of the small size of DNS dataset, the detection using OCSVM is superior to the one using K-Means especially that K-means is a cluster classifier that work better when the clusters are even which is not the case of DNS tunnelling.

OCSVM with the Radial Basis Function kernel obtained the higher and best result with 96% F-measure.

IV. COMPARING TECHNIQUES

Few papers in the literature addressed methods comparison. Nonetheless, a comparative analysis for detecting DNS tunnelling using Machine learning techniques was presented by Mahmoud Sammour et al. in [16] in order to identify the most accurate classifier. The techniques used are: Support Vector Machine (SVM), Naïve Bayes (NB) and Decision Tree (DT). SVM has outperformed the two other classifiers due to its high performance in handling multiple numbers of class labels. The two others have performed approximately the same.

SVM achieved 83% F-measure, NB 79% and 78% by DT.

Saeed Shafieian et al. [17] addressed DNS protocol exploitation that causes sensitive data exfiltration via tunnelling.

Signature-based intrusion detection isn't effective. Therefore, the authors proposed a technique that employs an ensemble of machine learning algorithms that are different in nature. The algorithms used are: Random Forests, K-Nearest (K-NN) and Multi-layer perception (MLP).

Results show the following:

- Ensemble of machine learning classifiers performs better than single one.
- The Ensemble of RF and Multi-layer perceptron have near false positive in detecting DNS tunneling.
- Weight of classifier and the combination rule affect the performance.
- Adding more classifiers can reduce the performance.
- SVM has outperformed the NB and DT by achieving the highest F-measure.
- DNS tunneling detection in Mobile networks using OCSVM is superior to the one using K-Means.
- Logistic regression and K-Means are used for data exfiltration and C&C tunnel detection with low false positive and high detection rate.
- Bayes Classifier of supervised learning can be used as a reliable and fast DNS tunnelling detector.

TABLE 1. MLT CLASSIFICATION

Paper	DNS Tunnelling Detection	
	Method(s)	**Result**
[11]	Bayes Classifier - Machine learning-based Analysis	Fast and reliable DNS tunnelling detection.
[13]	Logistic Regression, K-Means	Low false positive and high detection rate. LR: false positive rate 0.189%. K-Means: false positive rate 0.40%.
[14]	Binary-Classification compared with SVM, DT and Logistic Regression	High detection accuracy 99.96%
[15]	One Class Support Vector Machine, K-Means	OCSVM accuracy is superior to K-Means. F-Measure 99.6%
[16]	Support Vector Machine, Naïve Bayes, Decision Tree (DT).	F-measure: SVM 83% NB 79% DT 78%
[17]	Random Forests, K-Nearest, Multi-layer perception.	An ensemble of RF and Multi-layer perceptron have near-zero false positive

Based on the surveyed works in this paper, Table 1 summarizes and compares the main aspects of each methods. As shown in Table 1, methods tend to use different known algorithms. Results shows reliable DNS tunneling techniques using Bayes, K-means and logistic regression. OCSVM is better than K-Means and SVM is better than Bayes and DT.
Binary classification outperforms SVM, DT and logistic regression.
An ensemble of machine learning classifiers performs better than single one: RF and Multi-layer perceptron have near-zero false positive.

V. CONCLUSIONS

In this paper, we surveyed some of DNS tunnelling detection techniques using machine learning and the approaches cover different range of tunnelling detection to better define the scope of research. With this variety, it is a challenging task to identify the most suitable classifier, which would fit the process of detecting DNS tunnelling. Throughout this survey, we have shown several challenges for researchers in the field: Results for different machine learning method don't provide the same performance metrics and use different datasets.

REFERENCES

[1] RFC 1035, Domain Names - Implementation and Specification, P. Mockapetris, the Internet Society (November 1987)
[2] Recursive and Iterative DNS Queries. http://www.omnisecu.com/tcpip/recursive-and-iterative-dns-queries.php
[3] G. Franham and A. Atlasis, "Detecting DNS Tunneling" SANS institute InfoSec Reading Room. 2013
[4] UAE Today " Advanced DNS Protection is the need of the hour of Middle East telco operators, in the light of Etisalat website hacking " http://www.uaetoday.com/news_details.asp?newsid=52987
[5] 2017 Report - The global DNS threat Survey http://www.infosecurityeurope.com/__novadocuments/445925?v=636554315770370000
[6] C. Mullaney. Morto worm sets a (DNS) record. Technical report, Symantec, 2011. http://www.symantec.com/connect/blogs/morto-worm-sets-dns-record.
[7] Christian J. Dietrichyz, Christian Rossowz, Felix C. Freilingy, Herbert Bos, Maarten van Steen and Norbert Pohlmannz, "On Botnets that use DNS for Command and Control".
[8] Maurizio Aiello, Alessio Merlo2, Gianluca Papaleo," Performance Assessment and Analysis of DNS Tunneling Tools"
[9] Iodine. https://code.kryo.se/iodine/
[10] DNS2tcp. https://tools.kali.org/maintaining-access/dns2tcp
[11] Maurizio Aiello, Maurizio Mongelli, Gianluca Papeleo, "Basic Classifiers for DNS tunneling Detection (2013)
[12] M. Aiello, M. Mongelli, and G. Papaleo, "DNS Tunneling detection through statistical fingerprints of protocol messages and machine learning". (2014)
[13] Anirban Das, Min-Yi Shen, Madhu Shashanka and Jisheng Wang, "Detection of Exfiltration and tunneling over DNS". (2017)
[14] Jingkun Liu, Shuhao Li, Yongzheng Zhang, Jun Xiao, Peng Chang and Chengwei Peng, "Detecting DNS Tunnel through Binary-Classification Based on Behavior Features". (2017)
[15] Van Thuan Do, Paal Engelstad, Boning Feng, and Thanh Van Do, "Detection of DNS tunneling in Mobile Networks using Machine Learning. (2017)
[16] Mahmoud Sammour, Burairah Hussin, Mohd Fairuz Iskandar Othman, "comparative Analysis for detecting DNS tunneling Using Machine Learning techniques. (2017)
[17] Saeed Shafieian, Daniel Smith, and Mohammad Zulkernine, "Detecting DNS Tunneling Using Ensemble Learning". (2017)