NTNU – Trondheim
Norwegian University of
Science and Technology

Detecting DNS tunneling using machine learning

Terje Kristoffer Skow

Submission date: November 2015
Responsible professor: Than Van Do, ITEM
Supervisor: Hai Ngyuen, Telenor Research

Norwegian University of Science and Technology
Department of Telematics

Abstract

Hello, here is some text without a meaning. This text should show what a printed text will look like at this place. If you read this text, you will get no information. Really? Is there no information? Is there a difference between this text and some nonsense like “Huardest gefburn”? Kjift – not at all! A blind text like this gives you information about the selected font, how the letters are written and an impression of the look. This text should contain all letters of the alphabet and it should be written in of the original language. There is no need for special content, but the length of words should match the language.

This is the second paragraph. Hello, here is some text without a meaning. This text should show what a printed text will look like at this place. If you read this text, you will get no information. Really? Is there no information? Is there a difference between this text and some nonsense like “Huardest gefburn”? Kjift – not at all! A blind text like this gives you information about the selected font, how the letters are written and an impression of the look. This text should contain all letters of the alphabet and it should be written in of the original language. There is no need for special content, but the length of words should match the language.

And after the second paragraph follows the third paragraph. Hello, here is some text without a meaning. This text should show what a printed text will look like at this place. If you read this text, you will get no information. Really? Is there no information? Is there a difference between this text and some nonsense like “Huardest gefburn”? Kjift – not at all! A blind text like this gives you information about the selected font, how the letters are written and an impression of the look. This text should contain all letters of the alphabet and it should be written in of the original language. There is no need for special content, but the length of words should match the language.

After this fourth paragraph, we start a new paragraph sequence. Hello, here is some text without a meaning. This text should show what a printed text will look like at this place. If you read this text, you will get no information. Really? Is there no information? Is there a difference between this text and some nonsense like “Huardest gefburn”? Kjift – not at all! A blind text like this gives you information about the selected font, how the letters are written and an impression of the look. This text should contain all letters of the alphabet and it should be written in of

the original language. There is no need for special content, but the length of words should match the language.

Hello, here is some text without a meaning. This text should show what a printed text will look like at this place. If you read this text, you will get no information. Really? Is there no information? Is there a difference between this text and some nonsense like “Huardest gefburn”? Kjift – not at all! A blind text like this gives you information about the selected font, how the letters are written and an impression of the look. This text should contain all letters of the alphabet and it should be written in of the original language. There is no need for special content, but the length of words should match the language.

Preface

Hello, here is some text without a meaning. This text should show what a printed text will look like at this place. If you read this text, you will get no information. Really? Is there no information? Is there a difference between this text and some nonsense like “Huardest gefburn”? Kjift – not at all! A blind text like this gives you information about the selected font, how the letters are written and an impression of the look. This text should contain all letters of the alphabet and it should be written in of the original language. There is no need for special content, but the length of words should match the language.

Contents

List of Figures

List of Tables

List of Algorithms

List of Acronyms

DNS Domain Name System.

DPI Deep Packet Inspection.

RR Resource Record.

TTL Time to live.

VPN Virtual Private Network.

Chapter 1
DNS

1.1 Introduction

Domain Name System (DNS) is an important part for the internet. It is a system of distributed databases which contains the information about all the domains. In the mid and late 1980s did the previous system, `HOST.TXT`, encounter problems [MD88] which lead to the creation and standardizing called DNS. Since that has the DNS system been updated and configured many times. It needed to be able maintain a fast response time as the database grew larger, this was solved by using a hierarchical set up. This means that each server only has a limited information and sends the request to a new server until it reaches the correct server. It started with one root server, which has expanded to 13 today. The each layer of the hierarchy is called a zone, and it delegates the responsibility for underlying zones delimited by the `dot` in the request name Figure 1.1.

1.2 Structure

The data in the databases are called RR and contains the information about what the server do with the request. It has the following fields [Moc83]:

- NAME – the owner name of the record.
- TYPE – what type of record this is, name-to-IP (A) or IP-to-name (PTR).
- CLASS – define the class of the record, usually `IN` for internet. It is not widely use and not important for this paper.
- TTL – an integer which says how long the record should be cached by the server receiving the response.
- RDLENGTH – Specifies the length of the payload in number of octets. One octet is one octet of bits which corresponds to one character

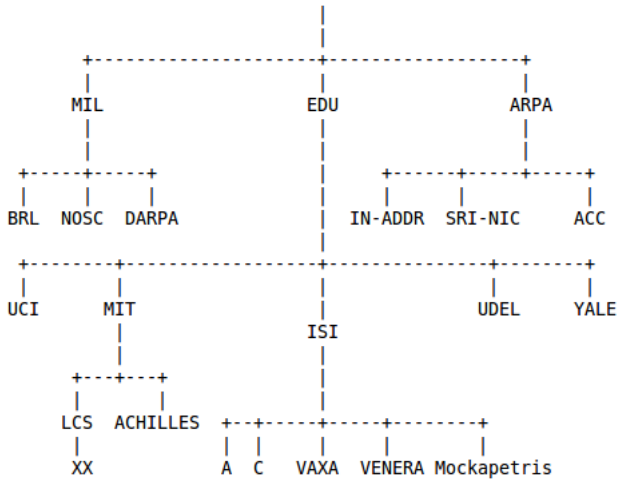

Figure 1.1: Example of name spaces of a root with MIL, EDU and ARPA as immediate subdomains. Each leaf is a domain [Moc87].

– RDATA – the payload of the record. The format and length varies depending on the TYPE and CLASS of the RR.

DNS was first implemented with around 15 different RR TYPE, which has now increased to over 30 [Far13] as a result of the development of the internet. The most notable values for TYPE are:

– A – the payload will contain the ipv4 address of the hostname requested. This is the most used TYPE

– AAAA – contains the ipv6 address of the hostname.

– CNAME – canonical name, respond with the correct alias of the hostname.

– MX – the mail exchange for the domain

– TXT – a text response with large payload, can be used in many ways and are an important type in DNS tunneling.

– PTR – pointer record. Used to map a hostname to an IP-address, commonly known as a reverse lookup.

– NS – authoritative name server for the domain

The 'A' type RR for telenor.no at the name server will look something like this:

Field	Value
NAME	telenor.no
TYPE	A
CLASS	IN
TTL	300
RDLENGTH	15
RDATA	153.110.156.145

Table 1.1: Example of RR for telenor.no

1.3 How it works

To explain of DNS works is an example the easiest way

When a request goes through DNS it starts in the root zone, where it sent down the hierarchy to the `.no` zone.

Normally a DNS server in an enterprise does not send requests directly to the internet, but use an internal DNS server instead. If you are the owner of the authoritative server for a domain, you can control the responses. This is what a DNS tunnel exploits, which will be explained more in the next section.

Chapter 2
DNS Tunneling

DNS tunneling was first used by people who exploited that DNS was not monitored in network you had to pay to use, e.g. hotels and cafés. It was used as an Virtual Private Network (VPN) tunnel. In later years it has been discovered that in enterprises the DNS are not monitored as much as other traffic on the network. People has therefore figured out that it is a good way to ex filtrate data in secure networks. DNS could also be used for a "command and control" attack, where commands are sent over DNS.

The way DNS works it that if you control the authoritative DNS server for a domain you can easily send commands.

With the increase of smartphones it has been discovered that DNS tunneling could again be used as the it started, to use the network without having to pay for it. Carriers can not start charging for regular queries since just regular use of a the internet produces a lot of DNS traffic. Which an user would not see and it would be hard for the carrier to explain for an user what he has been charged for.

Chapter 3
DNS Tunneling Detection

There has been done some research in detecting DNS tunneling over the years, but as it is still a problem no one has found a solution that is cost efficient. The best way for detecting tunnels is still Deep Packet Inspection (DPI) which is slows down the DNS requests as the amount of requests increase. DPI looks into each request and response for payload information which can indicate a DNS tunnel. For instance if requests maximizes the size of the labels and the overall name it should be looked at [Far13], this since tunnels would try to minimize the number of packages and maximize speed. Looking at the hostname should also be an indication since regular DNS names is dictionary words or have some meaning, while an encoded name would be meaningless. Traffic analysis is the other main alternative to detecting tunnels. Looking at volume, frequency and other attributes of DNS traffic could give indication of a tunnel. Earlier research has covered different techniques, looking at the volume of DNS traffic from a IP address or the volume of DNS traffic to a specific domain [Far13]. The overarching way of detecting tunnels with traffic analysis is looking for anomalies and stand out cases.

3.1 Traffic analysis

Data that is tunnelled through DNS is normally limited to 512 bytes per request, which leads to clients to send and receive lots of requests and responses. If the server should have the possibility to send data to the client will the client have to constantly send requests to get the data as a response from the server. All this leads to lots of DNS traffic which is not similar to normal use.

Chapter 4
Machine Learning

Chapter 5
Results

Chapter 6
Conclusion

References

[Far13] Greg Farnham. Detecting dns tunneling. *InfoSec Reading Room*, 2013.

[MD88] P. Mockapetris and K. J. Dunlap. Development of the domain name system. In *Symposium Proceedings on Communications Architectures and Protocols*, SIGCOMM '88, pages 123–133, New York, NY, USA, 1988. ACM.

[Moc83] Paul V Mockapetris. Domain names: Implementation specification. 1983.

[Moc87] Paul V Mockapetris. Domain names-concepts and facilities. 1987.