Putting DNS in Context

Mark Allman
International Computer Science Institute

IMC 2020
October 2020

“I was drivin’ through the misty rain
Searchin’ for a mystery train
Boppin’ through the wild blue
Tryin’ to make a connection with you”
Connection-Oriented DNS to Improve Privacy and Security

Abstract

Connection-Oriented DNS (CoDNS) is a protocol that allows DNS clients to establish a secure, connection-oriented channel with a DNS server. This channel is used to transfer DNS queries and responses securely, providing enhanced privacy and security for DNS transactions. CoDNS leverages established cryptographic techniques to protect DNS communications, ensuring that sensitive information is not exposed to potential eavesdroppers.

DNS Resolvers Considered Harmful

Abstract

The Domain Name System (DNS) is a critical component of the Internet infrastructure, but its reliance on DNS resolvers to handle queries exposes the system to potential security vulnerabilities. This paper explores the implications of these vulnerabilities and their impact on the overall security of the Internet. It discusses strategies to mitigate these risks, including the development of more secure DNS resolver implementations.

Internet Engineering Task Force (IETF)
Request for Comments: 8489
Category: Standards Track
ISSN: 2070-1721

DNS Queries over HTTPS (DoH)

Abstract

This document defines a protocol for sending DNS queries and getting DNS responses over HTTPS. Each DNS query-response pair is mapped into an HTTP exchange, ensuring that DNS queries are encrypted and protected from eavesdropping.

Internet Engineering Task Force (IETF)
Request for Comments: 7626
Category: Standards Track
ISSN: 2070-1721

Specification for DNS over Transport Layer Security (DTLS)

Abstract

This document describes the use of Transport Layer Security (TLS) to provide privacy for DNS. Encryption provided by TLS eliminates opportunities for eavesdropping and on-path tampering with DNS queries in the network, such as discussed in RFC 7626. In addition, this document specifies two usage profiles for DNS over TLS and provides advice on performance considerations to minimize overhead from using TCP and TLS with DNS.

This document focuses on forming stub-to-recursive traffic, as per the charter of the OPRIVE Working Group. It does not prevent future applications of the protocol to recursive-to-authoritative traffic.
Textbook DNS Usage

1.2.3.4  Client  Resolver

A www.icir.org?

1.2.3.4  www.icir.org

SYN

FIN+ACK
Previous Work

Demystifying Page Load Performance with WProf

Xiao Sophia Wang, Aruna Balasubramanian, Arvind Krishnamurthy, and David Wetherall
University of Washington

Abstract

Web page load time is a key performance metric that many techniques modern Web pages makes it difficult to identify performance bottlenecks. We present WProf, a lightweight browser plugin that produces a detailed line-by-line graph of the activities that make up a page's dependency tree. These activities are then analyzed post-mortem to identify performance bottlenecks. Existing Web performance tools are insufficient to capture the full picture of a Web page's load time.

Dissecting DNS Stakeholders in Mobile Networks

Mario Almeida, Alessandro Emardone, Diego Perino, Naeve Vanlina-Rodriguez, Matteo Varvello
Università Politecnica di Catalunya, Téléfónica Research, TIMPRA Network, INET

Abstract

The infrastructure of mobile apps involves a large number of protocols and services, with the Domain Name System (DNS) serving as a central point of contention. Despite the increasing importance of DNS, its performance is still not well understood. In this paper, we present a study of DNS traffic in mobile networks and its implications on app performance.

1. Introduction

According to recent estimates, mobile traffic is expected to be a $30 billion industry by 2021 [1]. As mobile networks are transitioning from 3G/4G to 5G/6G, understanding the performance of DNS is crucial for optimizing end-user experience. In this paper, we present a comprehensive study of DNS traffic in mobile networks and its implications on app performance.
Dataset From Case Connection Zone

≈100 homes

Bro Monitor
Feb 2019; 7 days

1Gbps

Internet

9.2M DNS Transactions

<table>
<thead>
<tr>
<th>ISP</th>
<th>72.8%</th>
</tr>
</thead>
<tbody>
<tr>
<td>Google</td>
<td>12.9%</td>
</tr>
<tr>
<td>OpenDNS</td>
<td>9.4%</td>
</tr>
<tr>
<td>Cloudflare</td>
<td>3.9%</td>
</tr>
</tbody>
</table>

11.2M App. Transactions

<table>
<thead>
<tr>
<th>Protocol</th>
<th>Percentage</th>
</tr>
</thead>
<tbody>
<tr>
<td>TCP</td>
<td>88%</td>
</tr>
<tr>
<td>UDP</td>
<td>12%</td>
</tr>
</tbody>
</table>
Methodology: Pairing

1.2.3.4

Client

Resolver

www.icir.org=1.2.3.4

www.psg.com=4.3.2.1

www.isi.edu=2.4.6.8

DN-Hunter
Bermudez, et.al.
IMC 2012
Methodology: Blocking

How do we determine whether a connection is “blocked”?

1.2.3.4

Blocked

Not Blocked

Client

Resolver

SYN

SYN
Methodology: Blocking

CDF

Blocked
Not Blocked

≈ 20 msec

100 msec

Time Between DNS and Connection (sec)

1
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0
0.001 0.01 0.1 1 10 100 1000 10000
DNS Record Origin

- We’ll now use our two primitives—pairing and blocking—to categorize application transactions based on where the corresponding DNS information comes from.
No DNS

1.2.3.4

Client

Resolver

FIN+ACK

SYN

- 7.2% of conns.
- 82% are p2p
- the remaining cases seem to largely represent hard-coded IPs
Prefetched

1.2.3.4

Client

Resolver

First conn. to use DNS lookup

Not Blocked (> 100msec)

SYN
Prefetched

- 7.8% of conns. (875K conns.)
- 3.1M DNS lookups never used
- ≈22% of speculative lookups are used
Both connections pair with same DNS lookup

Not Blocked (> 100msec)
Local Cache

- 42.9% of conns.
- 22.2% of conns. using local cache use expired records!
Shared Resolver Cache

- 26.3% of connections
- Cache hit rate: ≈ 63%

Block (<= 100msec)
Auth Resolution Required

1.2.3.4  Client  Resolver  ADNS

- 15.7% of connections
- we cannot determine how many lookups the resolver does
- likely still benefits from caching

> minRTT

SYN

Blocked (<= 100msec)
## DNS Record Origin

<table>
<thead>
<tr>
<th></th>
<th>% Connections</th>
</tr>
</thead>
<tbody>
<tr>
<td>No DNS</td>
<td>7.2</td>
</tr>
<tr>
<td>Local Cache</td>
<td>42.9</td>
</tr>
<tr>
<td>Prefetched</td>
<td>7.8</td>
</tr>
<tr>
<td>Shared Resolver Cache</td>
<td>26.3</td>
</tr>
<tr>
<td>Auth Resolution Required</td>
<td>15.7</td>
</tr>
</tbody>
</table>

57.9% of connections have no direct DNS costs.

42.1% of connections don’t have needed DNS information on hand.
Absolute DNS Cost

75th perc \( \approx 20\text{msec} \)

\( \approx 97\% < 100\text{msec} \)

Shared cache hits
Relative DNS Cost

\[ \text{DNS\%} = 100 \times \frac{D}{T} \]
Relative DNS Cost

- Median: 0.03%
- DNS is <= 1% for ≈80% of the application transactions
- DNS is <= 10% for ≈92% of the transactions
DNS Cost

• We say application connections requiring a DNS lookup experience “significant DNS cost” when both the following criteria are met:

  • DNS lookup duration $> 20\text{msec}$
  • DNS percentage contribution $> 1\%$
### DNS Cost

<table>
<thead>
<tr>
<th>DNS Percent</th>
<th>DNS Duration</th>
<th>&lt;= 20msec</th>
<th>&gt; 20msec</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;= 1%</td>
<td></td>
<td>64.0</td>
<td>15.9</td>
</tr>
<tr>
<td>&gt; 1%</td>
<td></td>
<td>11.5</td>
<td>8.6</td>
</tr>
</tbody>
</table>

- 64% of transactions show an insignificant DNS cost using both criteria.
- 27% of transactions show an insignificant DNS cost using one criteria.
- <9% of transactions show an significant DNS cost using both criteria.
Also In The Paper ...

- More methodological details
  - including ethical considerations in appendix

- Analysis as a function of resolver
  - ISP vs. Google vs. OpenDNS vs. Cloudflare

- Brief analysis of how to improve DNS further
Questions? Comments?

Mark Allman, mallman@icir.org
https://www.icir.org/mallman/
@mallman_icsi
EXTRA SLIDES