Understand Umbrella DNS with QNAME Minimization

Contents

Introduction

This document describes how to use the Cisco Umbrella Domain Name System (DNS) with QNAME minimization.

Prerequisites

Requirements

There are no specific requirements for this document.

Components Used

The information in this document is based on Cisco Umbrella

The information in this document was created from the devices in a specific lab environment. All of the devices used in this document started with a cleared (default) configuration. If your network is live, ensure that you understand the potential impact of any command.

Overview

In June 2019, Cisco Umbrella added support for query name minimization (RFC7816). QNAME minimization is a privacy oriented feature in DNS which aims limit the sending of the full domain destination to the root nameservers. As a result, the flow of DNS queries to determine the DNS query response is modified.

QNAME Minimization is a worldwide topic. The Internet Systems Consortium has an introduction article on QNAME Minimization. Mozilla Firefox requires resolvers to use QNAME Minimization for DNS over HTTPS implementations and has an article on this topic.

Understand Query Minimization

Query minimization is a new data privacy-centric approach to DNS authoritative queries. To explore what query minimization is, start with an explanation of how a DNS request currently works today.

Since most human interaction with the Internet begins with a DNS query, big data on where users are going is invaluable information, which can be considered private data.

For this example, you are looking to visit umbrella.cisco.com. You need a DNS query to determine where this server is located, so Umbrella sends that query to a recursive DNS server to find the answer from the authority using these steps:

1. User query to the recursive DNS resolver: umbrella.cisco.com

2. Recursive DNS server queries the answer from the root nameservers: where can I find umbrella.cisco.com to root > answer for .com

3. Query at the .com name servers: umbrella.cisco.com to .com > gets location of cisco.com nameservers

4. Query to cisco.com name servers: umbrella.cisco.com to cisco.com > Answer provided

In many cases, this can continue with several more iterations to different nameservers until an A-record is located. In steps 1-2, Umbrella is only actively seeking the location of the .com nameservers. However, the full umbrella.cisco.com domain is sent to the root and .com nameserver. The same goes for the cisco.com nameserver receiving the full query.

With query minimization, the algorithm shifts to only asking for the required level of detail in the upstream queries:

1. User query to the recursive DNS resolver: umbrella.cisco.com

2. Recursive DNS server queries the root nameservers: where can I find .com > answer for .com

3. Query at the .com name servers: cisco.com to .com > location of cisco.com

4. Query at the cisco.com nameservers for umbrella.cisco.com > Answer

This works great in most cases, and allows the answer to be located without revealing the unique query being made to the root or TLD nameservers.

This privacy is even more important for domains that make use of EDNS Client Subnet, where the DNS authority is informed of the user's source C-Block (/24) when querying. Without QNAME minimization, the root and .com (in this example) nameservers know your general location as well as where exactly you are going. With QNAME Minimization, the roots only know that someone is looking for .com and the privacy of the requester is maintained. They do not require the level of detail provided to them today without QMIN privacy protections.

Potential Side Effects

QNAME minimization works without issue in most cases. However, it is subject to additional sources of failure compared to a direct query. Since the full destination is not revealed until the last step of the process to the authoritative nameserver, breaks in the DNS chain can break resolution of the domain. For example, here is a long fictional name - umbrellas.in.the.rain.umbrella.cisco.com. This can result in these queries:

1. What is the nameservers for .com to the root servers .

2. What is the nameservers for cisco.com to the .com servers

3. What is the nameservers for umbrella.cisco.com to the cisco.com nameservers

4. What is the nameservers for rain.umbrella.cisco.com to the umbrella.cisco.com nameservers.

5. What is the nameservers for the.rain.umbrella.cisco.com to the rain.umbrella.cisco.com nameservers

6. What is the nameservers for in.the.rain.umbrella.cisco.com to the rain.umbrella.cisco.com nameservers: SERVFAIL

7. What is the nameservers for umbrellas.in.the.rain.umbrella.cisco.com to the rain.umbrella.cisco.com nameservers (not queried due to SERVFAIL earlier)

8. What is the answer for umbrellas.in.the.rain.umbrella.cisco.com to the umbrellas.in.the.rain.umbrella.cisco.com nameservers that were found earlier (not queried due to SERVFAIL earlier)

Since the roots are not given the full query, if one of the levels of the domain returns a NXDOMAIN, SERVFAIL, the IP of a RFC-1918 internal nameserver, or other poor response, the query can fail to receive a successfully upstream authoritative response. For example, if the sixth step earlier (bold, underlined) were to fail, the query for umbrellas.in.the.rain.umbrella.cisco.com can fail to resolve. To resolve these issues, the domain owner must ensure that each level has a valid public response.