MENU

text/plain

ericlaw talks about the web and software in general

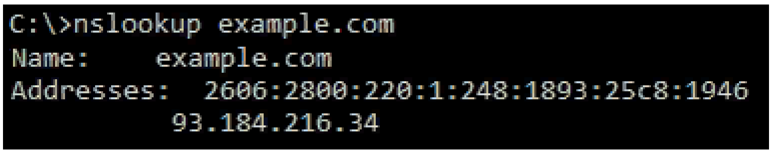
2019-11-06

Thoughts on DNS-over-HTTPS

Type https://example.com in your web browser's address bar and hit enter.

What happens?

Before connecting to the example.com server, your browser must convert "example.com" to the network address at which that server is located.



```
C:\>nslookup example.com
Name:    example.com
Addresses:  2606:2800:220:1:248:1893:25c8:1946
          93.184.216.34
```

It does this lookup using a protocol called "DNS." Today, most DNS transactions are conducted in plaintext (not encrypted) by sending UDP messages to the DNS resolver your computer is configured to use.

There are a number of problems with the 36-year-old DNS protocol, but a key one is that the unencrypted use of UDP traffic means that network intermediaries can see (and potentially modify) your lookups, such that attackers can know where you're browsing, and potentially even direct your traffic to some other server.

The DNS-over-HTTPS (DoH) protocol attempts to address some of these problems by sending DNS traffic over a HTTPS connection to the DNS resolver. The encryption (TLS/QUIC) of the connection helps prevent network intermediaries from knowing what addresses your browser is looking up– your queries are private between your PC and the DNS resolver that is providing the answers. The expressiveness of HTTP (with request and response headers) provides interesting options for future extensibility, and the modern HTTP2 and HTTP3 protocols aim to provide high-performance and parallel transactions with a single connection.

Try It

Support for DNS-over-HTTPS is coming to many browsers and operating systems (including a future version of Windows). You can even try DoH out in the newest version of Microsoft Edge (v79+) by starting the browser with a special command line flag. The following command line will start the browser and instruct it to perform DNS lookups using the Cloudflare DoH server:

```
msedge.exe --enable-features="DnsOverHttps<DoHTrial" --force-fieldtrials="DoHTrial/Group1" --force-fieldtrial-params="DoHTrial.Group1:Fallback/false/Templates/https%3A%2F%cloudflare-dns.com%2Fdns-query"
```

You can test to see whether the feature is working as expected by visiting https://1.1.1.1/help. Unfortunately, this command line flag presently only works on *unmanaged* PCs, meaning it doesn't do anything from PCs that are joined to a Windows domain.

Some Thoughts, In No Particular Order

Long-time readers of this blog know that I want to "HTTPS ALL THE THINGS" and DNS is no exception. Unfortunately, as with most protocol transitions, this turns out to be very very complicated.

SNI

The privacy benefits of DNS-over-HTTPS are predicated on the idea that a network observer, blinded from your DNS lookups by encryption, will not be able to see where you're browsing.

Unfortunately, network observers, by definition, can *observe* your traffic, even if the traffic encrypted.

The network observer will still see the IP addresses you're connecting to, and that's often sufficient to know what sites you're browsing.

Worse, they are usually still able to tell what *specific* HTTPS site you're visiting on that IP address. That's because one of the current limitations of HTTPS is that the browser sends, in unencrypted form a Server Name Indication (SNI), the hostname it expects to see in the server's certificate as a part of the ClientHello HTTPS handshake message. Closing this hole requires implementation of Encrypted SNI (ESNI) and this feature is not yet implemented in Chromium.

Privacy From Observers, Not the Resolver

If your Internet Service Provider (say, for example, Comcast) is configured to offer DNS-over-HTTPS, and your browser uses their resolver, your network lookups are protected from observers on the local network, but *not* from the Comcast resolver.

Because the data handling practices of resolvers are often opaque, and because there are business incentives for resolvers to make use of lookup data (for advertising targeting or analytics revenue), it could be the case that the very actor you are trying to hide your traffic from (e.g. your ISP) is exactly the one holding the encryption key you're using to encrypt the lookup traffic.

To address this, some users choose to send their traffic not to the default resolver their device is configured to use (typically provided by the ISP) but instead send the lookups to a "Public Resolver" provided by a third-party with a stronger privacy promise.

However, this introduces its own complexities.

Public Resolvers Don't Know Private Addresses

A key problem in the deployment of DNS-over-HTTPS is that public resolvers (Google Public DNS, Cloudflare, Open DNS, etc) cannot know the addresses of servers that are within an intranet. If your browser attempts to look up a hostname on your intranet (say MySecretServer.intranet.MyCo.com) using the public resolver, the public resolver not only gets information about your internal network (e.g. now Google knows that you have a server called MySecretServer.intranet) but it also returns "Sorry, never heard of it." At this point, your browser has to decide what to do next. It might fail entirely ("Sorry, site not found") or it might "Fail open" and perform a plain UDP lookup using the system-configured resolver provided by e.g. your corporate network administrator.

This fallback means that a network attacker might simply block your DoH traffic such that you perform *all* of your queries in unprotected fashion. Not great.

Even *alerting* the user to such a problem is tricky: What could the browser even say that a human might understand? "Nerdy McNerdy Nerd Nerd Nerd Nerd Nerd Address Nerd Resolution Nerd Geek. Privacy. Network. Nerdery. Geekery. Continue?"

Centralization Isn't Great

Centralizing DNS resolutions to the (relatively small) set of public DNS providers is contentious, at best. Some European jurisdictions are uncomfortable about the idea that their citizens' DNS lookups might be sent to an American tech giant.

Some privacy-focused users are primarily *worried* about the internet giants (e.g. Google, Cloudflare) and are very nervous that the rise of DoH will result in browsers sending traffic to these resolvers by default. Google has said they won't do that in Chrome, while Firefox is experimenting with using Cloudflare by default in some locales.

Content Filtering

Historically, DNS resolutions were a convenient choke point for schools, corporations, and parents to implement content filtering policies. By interfering with DNS lookups for sites that network users are forbidden to visit (e.g adult content, sites that put the user's security at risk, or sites that might result in legal liability for the organization), these organizations were able to easily prevent non-savvy users from connecting to unwanted sites. Using DoH to a Public DNS provider bypasses these types of content filters, leaving the organization with unappealing choices: start using lower-granularity network interception (e.g. blocking by IP addresses), installing content-filters on the user's devices directly, or attempting to block DoH resolvers entirely and forcing the user's devices to fall back to the filtered resolver.

Geo CDNs and Other Tricks

In the past, DNS was one mechanism that a geographically distributed CDN could use to load-balance its traffic such that users get the "best" answers for their current locale. For instance, if the resolver was answering a query from a user in Australia, it might return a different server address than when resolving a query from a user in Florida.

These schemes and others get more complicated when the user isn't using a local DNS resolver and is instead using a central public resolver, possibly provided by a competitor to the sites that the user is trying to visit.

Don't Despair

Despite these challenges and others, DNS-over-HTTPS represents an improvement over the status quo, and as browser and OS engineering teams and standards bodies invest in addressing these problems, we can expect that deployment and use of DoH will grow more common in the coming years.

DoH will eventually be a part of a more private and secure web.

-Eric Lawrence

Share this:

Twitter Facebook

One thought on “Thoughts on DNS-over-HTTPS”

larrylaca says:
2019-11-06 at 23:13

Eric: Excellent. Thx. Can you say more about ongoing efforts to develop an open privacy model, how this is filtering into the standards organizations, etc. Would also like to hear more about geo-political thrusts. You mentioned some EU concerns but the splinternet impacts of Russia, China, etc. are hard to read between the lines. Yet another need for a trusted, open privacy model.

Reply →

Leave a Reply

Email (required) (Address never made public)

Name (required)

Website

□ Notify me of new comments via email.

□ Notify me of new posts via email.

POST COMMENT

ABOUT

ERICLAW

Impatient optimist. Dad. Author/speaker. Created Fiddler & SlickRun. PM @ MSFT '01-'12, and '18-, presently working on Microsoft Edge. My words are my own.

View all posts

CATEGORIES

BROWSERS PRIVACY SECURITY WEB

TAGS

DNS DOH HTTPS PRIVACY PROTOCOL

PREVIOUS POST

bye: FTP Support Is Going Away

NEXT POST

AppOrWeb-to-WebApp Communication: Custom Scheme Handlers

BLOG AT WORDPRESS.COM.