The ISP Column
A monthly column on things Internet

April 2019
Geoff Huston

DNS Privacy at IETF 104

From time to time the IETF seriously grapples with its role with respect to technology relating to users' privacy. Should the IETF publish standard specifications of technologies that facilitate third party eavesdropping on communications or should it refrain from working on such technologies? Should the IETF take further steps and publish standard specifications of technologies that directly impede various forms of third party eavesdropping on communications? Is a consistent position from the IETF on personal privacy preferred? Or should the IETF be as agnostic as possible and publish protocol specifications based solely on technical coherency and interoperability without particular regard to issues of personal privacy?

This is not a new question for the IETF. Going back some twenty years the IETF was working on a standardization of a suite of media gateway protocols when the request was raised to make the protocols compliant with the US Communications Assistance for Law Enforcement Act (CALEA). This excited passions both within the IETF and in the broader circle of observers and commentators. The Electronic Privacy Information Centre communicated to the IETF its position, which resonated with many IETF participants at the time: "We are writing to urge the IETF not to adopt new protocols or modify existing protocols to facilitate eavesdropping. [..] we believe that such a development would harm network security, result in more illegal activities, diminish users' privacy, stifle innovation, and impose significant costs on developers of communications." [https://www.epic.org/privacy/internet/letter_to_ietf.html]. After much angst and debate, the IETF refused to act on this request, and documented its position in RFC 2804, "IETF Policy on Wiretapping" "The IETF has decided not to consider requirements for wiretapping as part of the process for creating and maintaining IETF standards." [RFC 2804, May 2000].

> To put this into some context, the telephone networks that preceded the Internet typically operated under a framework of interception capability and this capability was a mandatory requirement for licensed service operators for both their voice and data services. For the IETF to place interception capabilities out of scope for their standards work was not only a strong break from an establish public carriage function, but it threw into some confusion how vendors and operators could define an interoperable standard for interception requests. ETSI has evidently filled this gap with a set of standards for lawful interception (https://www.etsi.org/technologies/lawful-interception).
>
> However, this still presents real issues to both network operators and law enforcement agencies. One interesting approach in the New Zealand networking community was to support the development of a tool called "OpenLI", an open source implementation of the ETSI protocols (https://openli.nz) for use by local network operators.

The IETF's position of refusal to standardise surveillance-enabling architecture modifications did not settle the matter. It didn't settle the matter then and hasn't settled it now. Code and standard specifications of network protocols do not necessarily usurp our laws, and code, law and markets are all elements in a political tussle over what ultimately determines social policies and practices.

The ongoing situation over the ensuing decade was an uneasy standoff between the IETF, as the most visible body associated with the Internet's code base, and various public bodies. This situation changed in response to the revelations in the documents leaked by Edward Snowden in 2013. Snowden's disclosures of mass surveillance by the NSA (evidently working in close cooperation with related agencies in Australia, the United Kingdom and Canada) prompted the IETF to take a very strong public position: "Pervasive monitoring is a technical attack that should be mitigated in the design of IETF protocols, where possible." [RFC 7258, May 2014]. This is the IETF crossing into the second of the above questions. Rather than simply refusing to work on interception technologies, as espoused in RFC 2804, this later RFC advocates that the IETF should publish standard specifications of technologies that directly impede third party eavesdropping on communications.

It's a noble position that the IETF has taken, but it is perhaps a rather unworldly one in the light of subsequent concerns on the extent of corporate activities in this same area, an activity now with it's own name: "surveillance capitalism." The world of the Internet is now a world where surveillance dominates every aspect of the Internet's environment. The online market for good and services is distorted by the presence of "free" products and services that are funded through a back flow of advertising revenue based on a thorough and comprehensive knowledge of individual users, gained only by using thorough and comprehensive surveillance frameworks that target each and every user. The Internet is largely dominated, and indeed driven, by surveillance capitalism and pervasive monitoring is a feature, not a bug. Indeed, perhaps the only debate left today is one over the respective merits and risks of surveillance by private actors or surveillance by state-sponsored actors. The pronouncement of the IETF denouncing stats-sponsored surveillance can only generate a wry smile in retrospect. Sadly, pervasive monitoring is what generates the revenue that propels today's Internet, and the IETF is a coerced fellow traveller, despite the occasional bursts of sometime hysterical rhetoric that attempts to disavow any such relationship. We have come a very long way from this lofty moral stance on personal privacy into a somewhat tawdry and corrupted digital world, where "do no evil!" has become "don't get caught!"

It has been five years since RFC 7258 was published and the privacy issue refuses to go away. It seems that the IETF is heading into this turgid and complex field of privacy once more, this time because of the Domain Name System.

DNS Privacy and DNS over TLS

The DNS has always been a fertile field of opportunity for both surveillance and access control. The basic DNS name resolution protocol has always worked in a totally unencrypted mode, so that queries and responses are available to any party who can see these transactions on the wire. The wire protocol has no authentication, so a network operator can intercept DNS queries addressed to any IP address and provide a response in their name, while the querier is none the wiser. It's also the case that every single Internet transaction starts with a DNS name resolution query. The DNS is a timely and accurate indicator of everyone we do online, and it's an entirely unprotected and open protocol. What a rich minefield! Little wonder that many service operators and many nation states for that matter use the DNS for all kinds of purposes relating to both surveillance and access control.

The intersection of RFC 7258 and the DNS has generated the topic of "DNS Privacy!" complete with an IETF Working Group and a worthy collection of drafts of ideas of how to improve the privacy aspects of the DNS.

The first steps in this activity were to look at the interaction between the end client and their chosen recursive resolver. This is a critical element of the larger picture, as this is the only part of the DNS resolution service where the IP address of the end client is contained in the query. Once the query is passed within the DNS infrastructure the query contains no direct identifying link to the client.

As an aside it is worth a mention about EDNS(0) Client Subnet extension to DNS queries and the tension between privacy and performance levers that are accessible with such end user information leakage (RFC 7871).

The rise of Content Distribution Networks and multiple points of presence has led to a technique, commonly used by Akamai today as well as some others, where the assumed geolocation of the DNS resolver posing the question is a reasonable facsimile to the location of the end client. The concurrent rise of the use of open DNS resolvers, most notably the 8.8.8.8 service from Google, negated this assumption.

In response to the frustrations on the CDN side of misdirected users and woefully inefficient content delivery, the IETF standardised a mechanism to attach the subnet of the end client to the query, RFC 7871. The attachment was down through the use of the EDNS extensions mechanism, and the idea was to put the IP address of the end client making the query (or an IP prefix) into an attachment to the query that both survived recursive resolver hand-offs and was used as a distinguishing label in local cache lookups.

Semantically a bridge is being crossed here. Previously the DNS could be thought of as an invariant distributed database. No matter who poses a name query the response is always the same. Client Subnet is an overt admission that some folk want the DNS to be inconstant, such that the value of the response may depend on the identity of the querier. More importantly, a major privacy bridge is also being crossed. Previously, authoritative name servers were not exposed to the identity of the original client making the query. The DNS queries did not expose the original querier. With Client Subnet the authoritative server is aware of the original client. Interception and eavesdropping undertaken at the server end will enjoy a richer view of which end clients are expressing some level of interest in the names served by this authoritative server.

Perhaps in deference to RFC 7258, it should be noted that the IETF appeared to be reluctant to go there when specifying this Client Subnet extension, but nevertheless they ended up doing it!

I'll quote here Section 2 of RFC 7871, which is a good description of the level of compromise and discomfort that lies just beneath the surface of this DNS privacy debate in the IETF:

> "If we were just beginning to design this mechanism, and not documenting existing protocol, it is unlikely that we would have done things exactly this way.
>
> "The IETF is actively working on enhancing DNS privacy [DPRIVE_Working_Group] and the reinjection of metadata [METADATA] has been identified as a problematic design pattern.
>
> "As noted above however, this document primarily describes existing behavior of a deployed method to further the understanding of the Internet community.
>
> "We recommend that the feature be turned off by default in all nameserver software, and that operators only enable it explicitly in those circumstances where it provides a clear benefit for their clients. We also encourage the deployment of means to allow users to make use of the opt-out provided. Finally, we recommend that others avoid techniques that may introduce additional metadata in future work, as it may damage user trust.

> "Regrettably, support for the opt-out provisions of this specification are currently limited. Only one stub resolver, getdns, is known to be able to originate queries with anonymity requested, and as yet no applications are known to be able to indicate that user preference to the stub resolver."
>
> [RFC7871, Client Subnet in DNS Queries, May 2016]

The Transport Layer Security (TLS) protocol is capable of both encrypting the communication between a client and a server and also providing some assurance to the client that the server is operated under the authority of the named entity that the client intended to connect to. In much the same manner as TLS is used to protect HTTP sessions and provide some assurance that the service point is an authorised agent of the named service, this protocol can also be used in the DNS context between the end user's client stub resolver and their chosen recursive resolver service.

The DNS Privacy Working Group has worked on DNS over TLS [RFC 7858 and RFC 8310] and we are now seeing a number of DNS recursive resolver services that support DNS over TLS. Resolver code for Unbound, PowerDNS, and Knot exists, and Bind can be configured with TLS use though a stunnel configuration. So if you are prepared to leave the reservation and set up your own DNS resolution environment on your device you can bypass the open DNS resolution system provided by your ISP and use a DNS over TLS service which will hide your DNS queries and responses from your ISP, and any interested onlookers.

However, it has to be said that it's a highly qualified form of privacy. It's not a solution for everyone. Adding DNS over TLS support to your platform may require the installation of a third-party app on your device (which may or may not be possible in your device), and in any case the number of users who are willing to alter their device's DNS configuration is very limited. Even when the packing of the TLS service is quite seamless, such as in Android Pie's DNS privacy option, it probably still will not be broadly used. In Android's case it's an esoteric feature buried a few levels deep in menus, it is not necessarily supported on all Android platforms, and unless you already know about it you will probably never stumble over in when poking around in your device.

But configuring the client is only half the story. Who are you going to talk to? Which recursive resolvers support client connections using DNS over TLS?

It's an important question. While you are stopping others from looking over your shoulder at your online DNS activity, you are still telling your chosen resolver your complete DNS profile. Today, there are open DNS resolver servers being provided by Google, Cloudflare and Quad9.

Sharing your secrets with Google may sound a bit like dancing with the devil. Google's ad platform generates comprehensive user profiles and their ad support systems are certainly expert and capable practitioners of the art of surveillance capitalism! In their defence, I must note that Google clearly state that they do not use their public DNS service to reap user profile data and exercise strict controls over access to DNS data, but that itself raises the question of how such unilateral undertakings are enforced within the company. Google does not open itself up for any form of third-party compliance inspection. While their DNS practice statement is an excellent statement of noble intent, how can a user be assured that Google is thoroughly and completely committed to every detail in the practice statement?

Look at it from the user's perspective. Once you leave the reservation and configure your system to use a third-party open DNS resolver, you may also be leaving aside your local national regulatory framework. It's a mixed package, as you may be circumventing what you might think of as onerous national content controls, including DNS censorship, but at the same time you may also be circumventing any rights and protections you may have under these same national regulatory structures. Once you are outside of any national jurisdiction then who is left to oversee that service providers adhere to their stated practices in providing the service?

It's not just trust in the service provider at the other end of the TLS connection. Even accessing such a privacy-oriented service may present an issue. In its wisdom the IETF's DPRIVE Working Group standardised DNS over TLS over TCP Port 853. This is not Port 443 as used by TLS in supporting HTTP. Any network operator can prevent users from using this DNS overlay service by simply blocking all traffic to TCP port 853.

DNS over TLS represents a specialised service accessible to just a few. It's a service that is readily blocked. It's a service that may prevent surveillance on the wire, but still ends up sharing your DNS activity with the DoT service provider of your choice. You may well still be compromised in terms of assured privacy protection, but does it make you feel better having a choice as to which service operator you choose to expose yourself to?

DNS over HTTPS

What caused all the current fuss at the DNS sessions at the IETF was a variant of this DNS over TLS approach, termed DNS over HTTPS (DOH).

In terms of the carriage of DNS on the wire there is almost nothing that differs between DNS over TLS and DNS over HTTPS. Both take wire format DNS messages, encrypt them using TLS and use a TCP session between the client and resolver. In protocol terms of packets on the wire the only difference between the two approaches is that DoH uses the same TCP port number as HTTPS, namely port 443. It may sound like a cosmetic change but there are two very fundamental differences that transcend this simple protocol tweak.

Firstly, DOH very difficult to detect. It looks like HTTPS traffic and uses the same port as HTTPS traffic. One could make assumptions in the opening TLS handshake where the name of the server is carried in the clear, but work on encrypted SNI in TLS 1.3 is proceeding, and it's reasonable to believe that even this small aperture of visibility will be sealed up in the near future. If you also add TLS padding to the mix, then even traffic profile analysis would not necessarily reveal that it is a DNS session within the TLS stream.

If privacy is the goal, then what's to complain with this picture? Surely DoH offers the end user a package of encryption, mimicry and obfuscation that hides the DNS to all but the endpoints of the session.

The answer to this question leads to the second fundamental difference between DNS over TLS and DOH. We are no longer talking about an esoteric feature knob that requires a dedicated, or even foolhardy, user to turn it on. The DNS session may look like just another HTTPS session to the network, but it also looks like just another HTTPS session to a host platform. In other words, a browser may just turn on DoH. That's not the user turning it on, and not the platform turning it on, but the browser itself. There is no special configuration that needs to be in place by either the platform of the local network to support the operation of DOH. If a browser chooses to use DoH then there is little that the platform or the network can do to prevent it. If a browser has installed DoH support, then control over the DNS name resolution function has passed from the user to the browser provider, and rather than being an esoteric function enabled by a handful of users, it becomes a "mainstream" service used by potentially billions of end users. For example, it appears that Google's Chrome browser enjoys a 60% market share of browsers (http://gs.statcounter.com/browser-market-share). If Chrome enable DoH by default, then what would that mean for the entire DNS? Would it literally disappear from sight?

The second concern is the choice of DoH server. Instead of using a locally configured DNS resolver service provided by the ISP, DOH switches the situation to use a service configured by the browser. The early implementation of this service in Firefox require the explicit configuration of a trusted recursive resolver, in a manner similar to the configuration of the DNS over TLS server in Android Pie. What if the DOH resolver is configured by the browser by default?

Let's just pause for a second to think about this. DOH has the capability to place the control of the privacy setting for DNS queries into the hands of the browser, bypassing both the user and the local internet infrastructure, and can do so in a way that intertwines secure web services with secure DNS service. In privacy terms it sounds very enticing. The downside is that the user's browser is now sharing all of its local activity with the configured DOH server. To put it a different way, what part of "sharing your entire personal profile with Google" is consistent with our traditional concepts of privacy?

There is a second concern here as well. This ability for a browser and a DoH resolver to combine and thereby effectively dominate the Internet's namespace is a legitimate concern. Few companies are in such a position, but there are few companies left in the Internet's ecosystem. A very small number of digital behemoths inhabit the core of the Internet and these entities would be entirely capable of taking advantage of such an opportunity, were it offered to them. Google is the dominant provider of the platform in Android, the browser in Chrome and the DNS resolver in the 8.8.8.8 service. Would this be a case of a single corporate entity being in a position of overarching control of the Internet's entire namespace? Netflix already fielded an app that used its own DNS resolution mechanism independent of the platform upon which the app was running. What if the Facebook app included DOH? What if Apple's iOS used a DOH resolution mechanism to bypass local DNS resolution and steer all DNS queries from Apple's platforms to a set of Apple-operate name resolvers?

Can such positions be regulated? How can we be assured that transactions which now have disappeared from sight, and from any meaningful form of oversight, are still conducted with all due integrity? We have already seen many national regimes struggle with very real questions concerning the limitations of being able to impose constraints upon the actions of these entities. Are the concerns of Louis Brandeis in the first half of the twentieth century over the rise of industrial and financial behemoths that in his view were too big to effectively regulate come full circle?

What does DOH mean?

Here is the core of the collective angst and disquiet that was evident at IETF 104 when considering the implications of DOH and the "centrality" of Internet infrastructure.

We are attempting to actively withhold the DNS from the traditional forms of inspection and interception using access carriers and wire-based mechanisms. In so doing we are looking to counter what was perceived as a state-based surveillance operation that had assumed too much capability. That was the message I took from RFC 7258.

But in the case of the DNS have we over-achieved? In withholding our DNS secrets from one party, have we instead handed the entire plate to another? Have we now provided the private surveillance framework with a whole new trove of personal data to mine by ruthlessly exploiting the DNS in a manner that is entirely out of sight? Once the browsers and even the apps direct their name queries through encrypted channels to resolvers operated by the same browser and app providers, then have we dealt a body blow to any efforts to safeguard personal privacy on the Internet?

At least RFC7871 on Client Subnet had the decency to include an admonition to turn it off and a tacit apology for specifying a tool that had serious issues relating to erosion of user privacy in the DNS infrastructure. The DOH specification in RFC 8484 contains no such considerations. It omits to mention the security and privacy issues were a browser to invisibly co-opt the name resolution function and pass all its DNS traffic in a secure encrypted tunnel to a cooperating resolver using DOH that faithfully mimics conventional content transactions. It omits to mention the risks of increasing the "centrality" of the Internet when the DNS name resolution is forcibly sucked into the browser and application space and then concealed behind a veil of strong encryption.

It’s incredibly challenging to make the case that DOH enhances personal privacy. It probably doesn’t. It’s easier to sustain a case that DOH has the potential to change the parties whom you bring into your trust circle by virtue of being able to be privy to your private profile, and not necessarily in a good way. In and of itself such a substitution of trust should not necessarily be of concern. But now it's your browser that can make the decision as to whom you are trusting with your personal data, not you. And the parties who are looking to be your DOH trust partner are the same parties who have a direct and overbearing interest in selling you to the highest bidding advertiser.

It appears that the original disquiet on the part of the IETF was not that state-sponsored intelligence agencies collected intelligence, as, after all, that is their role, but a perception that the public accountability of some of these agencies had, in the IETF’s view, failed. It is ironic that the IETF’s response appears to literally hand the keys to an encrypted DNS over to a handful of private sector entities who appear to have no public accountability whatsoever. No wonder there was a pervasive sense of unease from many DNS folk at IETF 104 over the hasty standardisation of the DOH specification in 2018. It just doesn’t smell right!

Disclaimer

The above views do not necessarily represent the views or positions of the Asia Pacific Network Information Centre.

Author

Geoff Huston B.Sc., M.Sc., is the Chief Scientist at APNIC, the Regional Internet Registry serving the Asia Pacific region.

www.potaroo.net