September 2019
Geoff Huston

DNS Resolver Centrality

At various times the Internet has been touted as a triumph of the power of open markets and competition. This line of argument says that unfettered by the often regressive and stultifying hand of government regulation, open markets are able to react to the needs of consumers. The rigors of a competitive supply regime will wean out those providers who don't produce what consumers want, and also remove those who are less efficient than their competitors. However, this view represents a somewhat narrow perspective on the evolutionary pressures that we see in the Internet, and the actions of markets are but one part of the larger story here.

There one aspect of market operation that is conveniently set aside in arguing that the Internet represents a triumph of open markets, namely that of market failure. Markets can fail for many reasons but one of the more consistent forms of failure is the emergence of monopolies or cartels. When a single producer or small clique of colluding producers emerge in a market, and when those producers can erect barriers to competitive entry that shut out not only current competition but future competition as well, then the enticements to extract monopoly rental from the captive market are all but irresistible. At that point, according to conventional economic theory, some form of external intervention is required to restore the market to a place where competitive pressures can once more motivate production efficiencies and product relevance.

In looking at the Internet, it is possible to divide up the various roles and look at each component activity as a market in its own right. From the design and manufacture of microprocessor chips to the operation of online search engines, web hosting and browser software, there are a myriad of markets, and each is subject to various pressures relating to competition and nascent monopolisation. In this article I'll look at the resolution of domain names in the Internet, or DNS resolvers, and measure to what extent this environment is showing signs of incipient monopolisation, or to use a common terms in DNS conversations these days, to what extent the DNS resolution environment is *centralised*.

Why are we asking this question of the DNS at all? For many years DNS name resolution services were part of the portfolio of services provided by the ISP. Indeed, DNS resolution configuration found its way into the DHCP protocol (*dynamic host configuration* protocol), where the client system is automatically loaded with the IP addresses of DNS resolvers as part of the DHCP transaction. As with many aspects of the Internet, this has changed over time. The ISP-provided DNS resolver service has been changing with the increasing use of so-called open DNS resolvers. These systems offer an alternative to the ISP-provided resolver. They may be faster, or may give certain undertakings regarding privacy, or may selectively censor certain DNS names, or determinedly resolve all DNS names without any form of censorship whatsoever. In other words they attempt to differentiate themselves from the default DNS offering and in so doing meet particular user requirements or expectations that may not be met by the ISP-provided service.

Interestingly, the cost of the DNS resolution service is conventionally bundled into the ISP's access service and the user is not aware of an incremental cost. In other words, the user perspective is that the ISP-provided service is a free service. In providing an alternative DNS

service, the open DNS service providers have found it challenging to provide a fee-based service, and in general these alternate DNS services are provided free of cost to the user.

Such DNS service are not free to operate, and if usage increases so do their service operation costs. Each provider of such free services needs to generate revenue in some manner, and the revenue problem is exacerbated the more successful they are in their endeavours of providing a free name resolution service. In some cases, service revenue is based on promotion of other paid-for services and the free service is funded through some form of marketing budget. In other cases, and perhaps more disturbingly for privacy-sensitive users, the revenue stream comes from selling the DNS query stream to third parties, which, when combined with querier identity data becomes a form of monetisation of the end users of the service. Various forms of altruism also appear to exist in this space, and some providers fund their open DNS resolver services through cross-subsidisation from revenue gathered from other service offerings simply because they appear to believe strongly enough in a privacy-preserving honest high-performance DNS service that operates with integrity.

The general observation is that there is a considerable diversity of motives behind these free open DNS providers and it appears that not all them are entirely innocent and entirely altruistic in nature.

These open resolvers operate in the same way as any other DNS recursive resolver, in that they accept traditional DNS queries over UDP port 53, resolve the queried name and pass the result back to the requestor. These open resolvers apparently gained some level of popularity in locales where certain online services are censored in locally operated DNS resolvers. One of the attributes on some of these open DNS resolvers, notably Google's Public DNS, is that it faithfully resolves all DNS names. Others differentiate their DNS offering by make a point of censoring DNS names based on particular policy settings, whether its anti-malware or censorship of certain names that are associated with criminal, anti-social or otherwise harmful content. What metrics and inputs such filtering services use to make such determinations is well out of scope for this particular study.

These open resolvers operate on well-known IP addresses and a third party with access to the network close to the client can eavesdrop on these unencrypted UDP port 53 queries, and even intercept the queries and generate an answer in place of that which would've been generated by the addressed open DNS resolver.

The DNS is changing and with the advent of DNS over TLS (DoT) it is not possible for a man-in-the-middle to perform this inspection or interception of traffic destined to a DNS resolver, as the TLS session requires knowledge of a private key that is not normally known to the interceptor. This is taken a step further with DNS over HTTPS (DoH) where the browser is able to select a DNS resolver in a manner that is invisible to the underlying host platform and potentially invisible to the user as well.

Moving the DNS from the access ISP to the browser may not necessarily enhance open competition in the DNS world. In today's Internet just two browsers, Chrome and Safari dominate the browser world with an estimated 80% share of all users (https://gs.statcounter.com/browser-market-share). If the DNS becomes a browser-specific setting, then what would that mean for the DNS resolver market? And why should we care?

One way to phrase these concerns is to imagine an extreme Internet where all DNS resolvers are handled by only one DNS resolver service provider. In such a hypothetical Internet this provider would have considerable power. It could refuse to resolve a name, and through this action that name would effectively disappear from the Internet. Given that all Internet transactions start with a DNS resolution call it would also place the operator of this single DNS service in a remarkable position of privilege, having real time knowledge of what we are all doing on the Internet all of the time. So, taken to an extreme position, DNS centrality, when that centrality results in just a handful of DNS resolver providers for the Internet's entire user base, appears to be not in the interests of anyone bar this handful of providers that would be operating this highly centralised service environment.

It is not just the concentration of capability that is a by-product of centralization in the DNS. Part of the Internet's inherent strength lies in its diversity and the ability for a dispersed set of servers to operate in a loosely coupled manner. This diversity implies that there is a smaller level of common exposure to known exploits and potential vulnerabilities.

While the DNS name resolution protocol operates in the clear over UDP port 53, it's relatively easy to see what is happening in the DNS, which appears to have both good and bad consequences. But what happens when it is not so readily observable? Can we still understand the extent to which the DNS is becoming more or less centralised when the individual DNS transactions from the edge to the resolver are hidden behind the encryption provided by DoT or DoH?

It would be useful to understand what is going on in the DNS today, before there has been any major shift to adopt DoH or DoT by high-use applications such as browsers. Can we measure the level of DNS centrality in the Internet today?

In this article I'd like to describe how we are approaching this question at APNIC Labs.

Measuring DNS Centrality

Firstly, let's look at some models of the DNS.

The generic model of the DNS infrastructure is that of *stubs*, *recursives* and *authoritatives*. A *stub resolver* is a client-side DNS function. It passes queries to a recursive resolver. The *recursive resolver* performs the DNS name resolution function by querying *authoritative name servers*. While recursive resolvers were initially constructed as a single host system it appears that a far more common configuration these days is through the use of *resolver farms*, where a set of recursive resolvers is positioned behind a *front-end* load balancer. The user configures the IP address of the load balancer as the IP address of the recursive resolver, while the *back-end* resolver engines query authoritative servers. So while the standard model of the DNS is as shown in Figure 1, a more likely guess of the structure of the interior of the DNS so shown in Figure 2.

Figure 1 – The Architecture of the DNS

The first part of this exercise is to assemble a list of open resolvers that are used broadly across the internet. We are not going to look at accidentally configured open resolvers, and for the moment we've avoided ISP-based resolvers that appear to be open without being promoted as such (either through inattention or deliberately). We had to start somewhere with a list of open resolvers, and the list we chose is that provided by https://www.publicdns.xyz.

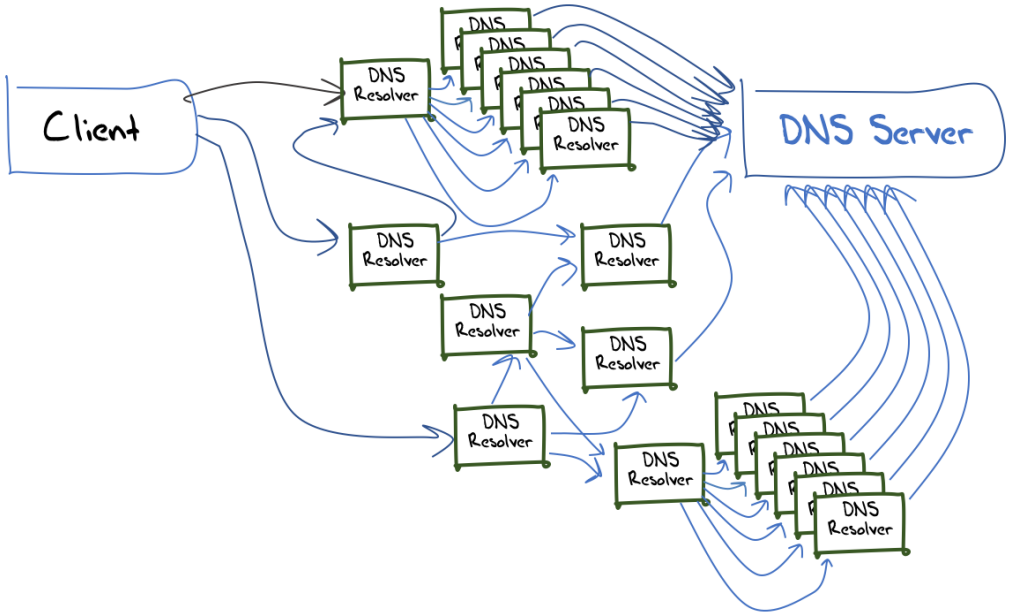

Figure 2 – DNS resolver infrastructure

This leads to the first issue in identifying which resolvers are actually used by these open DNS resolvers. We need to map the client side facing address (such as the address set 8.8.8.8, 8.8.4.4, 2001:4860:4860::8888 and 2001:4860:4860::8844 for Google's public DNS resolver service) to a set of resolver engine IP addresses that perform the queries seen by the authoritative server. Simple directed queries will not be all that helpful in this context, as the open resolver service address is commonly configured as an anycast address with multiple points of service presence. In some cases the open resolver operator provides a set of IP addresses that are used by the resolver system to query authoritative servers. Not all open DNS resolvers are as helpful as Google, whose resolver IP engine addresses are themselves loaded into the DNS at `locations.publicdns.goog`. To identify the larger set or worker addresses we need to query the open resolver with a unique DNS name, to prevent the resolver generating a response from its cache and do so from a diverse set of locations.

Years ago, this would've been a somewhat forbidding task involving assembling a large team of volunteers co-opted to assist in making these DNS queries, but these days tools are at hand to achieve this quickly and efficiently. We've used RIPE NCC's Atlas (https://atlas.ripe.net) to conduct this data gathering exercise. The interface to the Atlas measurement network allows the DNS query name to be unique and include the atlas probe identity as well as the time. After setting off a measurement via Atlas we collect the matching queries from our authoritative servers. The IP source addresses of the matching queries generally correspond to the IP addresses of the query engines associated with the open resolver service.

> The word "generally" is used here deliberately. It is clear that some probes are located behind DNS interceptors that take the probe's query and re-direct it to a different DNS resolver. In some cases, the re-direction is to an ISP-operated local DNS resolver, but in many more cases the query is mapped from one open resolver to another. This form of DNS interception is worthy of an article in its own right, so we'll note that we observed this behaviour here and move on how we measure resolver centrality.
>
> We also observed some forms of query stalking and replay, either by direct observation or with delayed re-queries that appear to be indicative of DNS query log replay. This is a long-standing issue that we looked at some years ago (https://www.potaroo.net/ispcol/2013-07/overlooking.html).

The reasons for including an encoding into the DNS query of the time that the query was generated allows us to strip off the most obvious log replay resolvers that lag the original query by a half a minute or longer.

This now gives us a set of open resolvers and the IP addresses (and AS numbers) used by the resolver engines associated with each of these open resolvers.

It is common to configure a host to query two or more recursive resolvers, either by name or by IP address. The theory behind this practice is that if one resolver fails to respond or responds with a SERVFAIL DNS response code, then there is a backup resolver that can be used. This implies that when we look at the resolvers that are used by end hosts there are in fact two measures to be made. One is the collection of resolvers that are invoked on the *first call*. What are the resolvers we use first off? The second measure is to discover the entirety of resolvers that are configured into end hosts. What are the resolvers we will turn to if the first resolver fails to respond? What is the user's *full resolver set*?

The next part of the measurement is the test setup. We've been using an online ad network for many years to perform measurement tests that span the breadth of the Internet to a level that exposes most public Internet service providers to the measurement. The ad campaign is currently generating between 10M and 15M impressions per day, spread across most of the Internet.

As always, at APNIC Labs we are indebted to Google Research for their generous assistance in this advertisement-based measurement program, and here we would like to once again explicitly acknowledge Google's unstinting and long-standing support to this APNIC Labs measurement program with our profound gratitude. Thanks indeed!

In this case, we use two URL components in the test.

The first is a URL with a unique DNS name that is readily resolvable. The DNS authoritative name servers respond on both IPv4 and IPv6 and responds in both UDP port 53 and TCP port 53. The DNS name is not DNSSEC-signed, so the resolution should be fast and efficient. The DNS name itself contains a number of fields, including the time of the measurement, allowing us to distinguish between primary queries and subsequent query replay events.

The second test is a DNS name where the servers always respond with a SERVFAIL response code, irrespective of the flags used in the DNS query. The intention of this test is to expose the complete set of resolvers that the stub resolver may use to resolve a name.

We perform query logging at the authoritative servers to map the end client, identified in the ad impression phase of the measurement, with the subsequent queries that are received by the authoritative servers.

We now have described the objective, the methodology, and the nature of the measurement vehicle. What results have been gathered so far?

Centrality Measurement Results

It should be noted that this is not intended to be a single one-off measurement, but a long baseline measurement. A similar exercise has been underway where we have been measuring the uptake of IPv6 in the Internet for the past eight years (https://stats.labs.apnic.net/ipv6/XA), and the uptake of DNSSEC validation by resolvers for the past six years (https://stats.labs.apnic.net/dnssec/XA). However, here we will present some initial results, based on recently collected data.

The first set of results concerns the overall concentration of resolvers in today's Internet. There are many ways to phrase this question to yield a single metric from the distribution of individual results. The question I will be using here is: What percentage of visible recursive resolvers are used by 90% of users? Figure 3 shows the concentration of resolvers when looking at the 'first use' resolver from a single day of collected data (across some 12M individual tests).

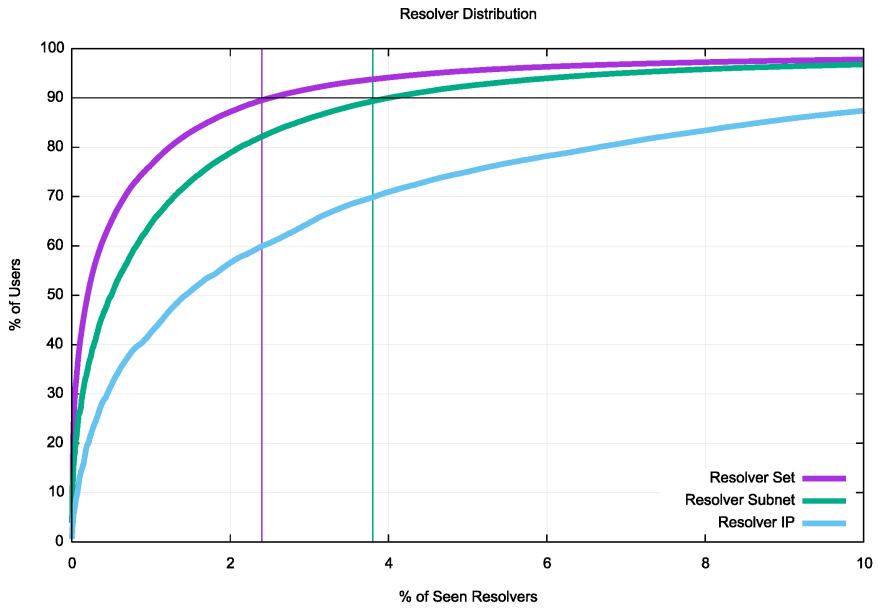

Figure 3 – Cumulative Distribution of Users and the First Resolver

There are three types of resolver sets shown in Figure 3. The first is taking the IP address used by the resolver. This IP address categorisation does not take into account resolver farms. Detecting resolver farms is challenging as an observer at the authoritative server. One way of grouping resolvers is through a common IP subnet of the resolver IP addresses. The subnet size we've used is a /24 in IPv4 and a /48 in IPv6. Some 75% of visible resolvers are members of a common subnet with at least one other resolver. The third grouping is one that uses the sets of resolver addresses that are used by the listed open resolvers, while the remainder of resolver IP addresses are grouped by their origin AS (or network).

Some 14% of resolvers are used by 90% of users when using resolver IP addresses when looking at the 'first use' resolvers by IP address. This number drops to slightly less than 4% when using a resolver subnet grouping and drops further to slightly over 2% when using open resolver lists and common AS groupings of resolvers.

This data is based on the resolution of a unique domain name that is not DNSSEC-signed and where the authoritative server always responds.

The distribution changes somewhat when we use a domain name which is not DNSSEC-signed, but where the authoritative server always responds with SERVFAIL irrespective of the flag settings in the query. The anticipated behaviour from stub resolvers is that this SERVFAIL response code will prompt the stub resolver to re-query using other locally configured resolvers until the local resolver set is exhausted and/or a local time limit for name resolution is reached. This experiment is intended to expose all resolvers that are used by the end-hosts' stub resolvers.

The results of this experiment using the same single day experiment results is shown in Figure 4.

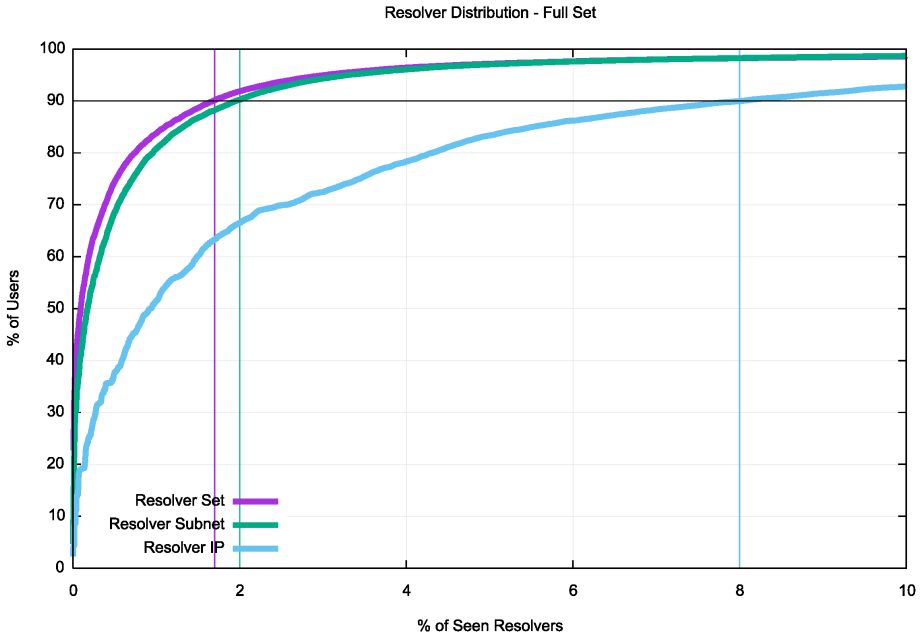

Figure 4 – Cumulative Distribution of Users and the Full Resolver Set

When we include all of the resolvers that are configured for use by end hosts the degree of commonality in resolvers increases significantly. By resolver IP address some 8% of resolvers are used by 90% of users. When considering resolver subnets the number drops to 2% and when using the open resolver sets and resolver AS's the number drops further to 1.8%.

There is an alternate way of interpreting the experiment data.

What proportion of resolvers fully encompass the resolver sets configured in end hosts? We will only count a user as included in this set if all of the user's configured resolvers are within the resolver set. While 90% of users have a common set of 1.8% of open resolvers and AS resolver sets configured (Figure 4), 90% of users have the entirety of their DNS queries directed to some 2.6% of grouped resolvers. In this case out or some 15M experiments on unique end points, some 592 grouped resolvers out of a total pool of 23,092 such resolver sets completely serve 90% of these 15M end points, and these users direct all their queries to resolvers in these 592 resolver sets.

Is this too *centralised*? Or is it a number of no real concern? *Centrality* is not a binary condition, and there is no threshold value where a service can be categorised as *centralised* or *distributed.*

It should be noted that the entire population of Internet endpoints could also be argued to be *centralised* in some fashion. Out of a total of an estimated 3.6 billion Internet users, 90% of these users appear to be located within 1.2% of networks, or within 780 out of a total number of 65,815 of ASNs advertised in the IPv4 BGP routing system. In that respect, the level of centralisation seen in the DNS is not unique to the DNS. If there were no open resolvers and every end point only used recursive resolvers that were operating from within the same network as the end point, the outcomes of the use of in-network recursive resolvers would be little different from the measurement results we have shown here. That observation would tend to suggest that if DNS centrality is a concern, then the evidence that creates the appearance of centrality in the DNS is actually the existence of a small number of networks who have extremely large user populations, which reflects the underlying centrality of Internet access providers in many markets.

Nevertheless, there is some evidence that open DNS resolvers enjoy a considerable level of use, either through direct configuration by end host users, or by referral from the network-provisioned DNS, or, perhaps as a future option, by the action of applications such as the intended operation of the Firefox browser and DNS over HTTPS.

Measuring Use of Open Resolvers

As already noted, we are using a list of open resolvers maintained at https://www.publicdns.xyz.

To what extent have these open resolvers been adopted by users?

Table 1 shows the percent of users over the period 21 August 2019 to 20 September 2019 who use open DNS resolvers, The data set encompasses a set of 303 M individual tests spanning the entire visible Internet (the results include a processing step that attempts to compensate for ad placement bias, which I won't describe in this article).

	First (%)	All (%)
Same AS	51.19	55.97
Same CC	36.50	39.65
Different CC	0.89	2.09
Any Open Resolver	15.28	28.27
Google Public DNS	9.12	22.36
Open DNS	1.44	2.41
114 DNS	1.36	2.80
DNSPAI	1.10	1.26
OneDNS	0.93	1.07
Level 3	0.81	1.99
Cloudflare	0.74	1.15
Yandex	0.17	0.33
Quad 9	0.08	0.16
Green Team DNS	0.06	0.06
Neustar	0.05	0.08
Verisign	0.04	0.09
Hurricane Electric	0.03	0.07
DNS Watch	0.03	0.10
Comodo	0.02	0.06
FreeDNS	0.02	0.05
CNNIC	0.02	0.03
BAIDU	0.01	0.04
Safe DNS	0.01	0.01
OracleDyn	0.01	0.02
Clean Browsing	0.00	0.01
DNS POD	0.00	0.00
Freenom	0.00	0.00
Uncensored DNS	0.00	0.01
All DNS	0.00	0.01
Quad 101	0.00	0.00
Open NIC	0.00	0.00
Alternate DNS	0.00	0.00
Puntcat	0.00	0.00

Table 1 – Relative use of Open DNS resolvers

I should note that the columns do not add up to 100%. Where a user has directed queries to two or more resolver types then they are counted against each of the resolvers that they use.

The first row is reporting that slightly over half of the users tested use DNS resolvers that are located in the same AS that they are themselves are located as their first resolvers. A further 4% of users have a local resolver listed in their resolver set, but not as the first-choice resolver.

A little over 1 in 6 users use one of these listed DNS Open resolvers as their primary resolver, and the number rises towards 1 in 3 when that is broadened to include all resolvers. It is highly unlikely that this

is solely the result of users changing their default DNS settings on their device, and points to a very widespread practice of network service providers using open DNS resolvers as either their primary resolver, or as a backup resolver.

Clearly, Google's DNS service is very widely used, both as a primary resolution service and as a secondary DNS service. More than 1 in 5 users have Google as some form of secondary backup resolver and just under 1 in 10 users send all their queries to Google's Public DNS service as their first-call resolver.

Open DNS is a long-standing open DNS resolver and its longevity may well be a factor in its relative use. The next two resolvers appear to be based in China, 114DNS is described at https://www.114dns.com/ and DNS PAI at http://www.dnspai.com/public.html.

While the global average of use of Open DNS resolvers is some 15%, the average for China is 20% of first use resolvers and 30% when we look at full resolver sets. Three Chinese DNS providers: 114DNS, DNS PAI and OneDNS are the most widely used in China, while Google's DNS service is used by some 4% of Chinese users.

Centrality?

There is no clear binary *centrality* state and there is no particular threshold value where one could pronounce that a market is excessively centralised or not.

In the Open DNS provider environment Google is clearly the dominant Open DNS provider across the entire Internet, while other providers appear to be used on a regional basis, such as the already noted Chinese DNS providers, or the use of Yandex in the Russian Federation. One might've expected Quad 101 service to be popular in Taiwan, but it appears that Google, Cloudflare and Open DNS enjoy a greater level of use there.

Is the DNS excessively *centralised* through the adoption of the use of open DNS resolvers? Some 87% of users use first call DNS resolvers provided by their network service provider or operated in the same economy as they are located in. A far smaller number use the services of open DNS resolvers, and it is challenging to make a case that this level of use represents some form of centralisation of the DNS.

Next Steps

These are just the first steps in a larger study of *centrality* in the DNS, particularly in relation to the possible uptake of the use of DoH in the Internet. Part of the current narrative about the use of DoH is that the browser can invisibly redirect a user's DNS queries to a DoH server and that this largely invisible redirection might cause a significant shift in the profile of use of DNS resolvers and could possibly generate conditions that would amount to *centrality* in the DNS.

As usual, data will assist in this area and our next step in the DNS measurement program is to measure the extent to which the DNS query patterns from the Firefox and Chrome browsers change as and when they deploy DoH within their browser.

The results we have gathered so far represent a baseline prior to the widespread uptake of DOH, assuming that it does indeed occur, and we will be looking for changes to the overall distribution of use of DNS resolvers over the coming months.

DNS Resolver Dashboard

The results of these resolver measurements are loaded into a system which is updated on a daily basis. The resolver measurement report is available at https://stats.labs.apnic.net/rvrs.

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