February 2026
Geoff Huston

Measuring DNS over IPv6

In theory, the design of IPv6 was such that the upper-level end-to-end transport protocols, namely TCP and UDP need not be aware of the IP layer protocol being used. We should be able to use IPv4 and IPv6 interchangeably in the DNS right?

This topic was the subject of an Internet Best Common Practice published in 2004, RFC 3901, "DNS IPv6 Transport Operational Guidelines". This document had two major recommendations:

- Firstly, every recursive name server should be either IPv4-only or dual stack.
- Secondly, every DNS zone should be served by at least one reachable name server using IPv4.

It's really saying "Don't stop using IPv4 yet!" which at the time was pragmatic advice.

There is a proposed revision to this document that is nearing publication in early 2026, RFC 3901bis, that's proposing a somewhat different set of recommendations. This document proposes that at least two nameservers for a zone are dual-stacked name servers, which implies that every nameserver would be reachable using IPv6. This document is saying as an operational guideline for IPv6 support in the DNS that: "It's time to take IPv6 seriously!" The assumption behind this RFC 3901bis document is that IPv6 is now a mature and well-understood technology and using IPv6 as a transport for the DNS is as efficient and fast as using IPv4.

But is this reasonable advice? The main operational problem for the DNS terms of service resilience has been in the area of adaptation of large payloads to travel through size-constrained networks. IPv6 removed the ability for routers to perform "fragmentation on the fly," and this has impacted the DNS when using an IPv6 UDP transport with large payloads.

> I have looked at this scenario a number of times in recent years in recent years, including "Adding IPv6-only to DNS and Truncation in UDP", "IPv6, the DNS and Happy Eyeballs", "IPv6 and the DNS", and there was a measured failure rate of some 40% when the DNS attempts to pass large responses over IPv6 that require packet fragmentation.

What if we restrict our view to the common DNS case, where queries and responses can sit comfortably within the 1,280-byte maximum unfragmented IPv6 packet size limit? How much of the Internet user base can reliably access a DNS server where the only form of access is via IPv6?

To answer such a question, we would need to undertake a broad measurement of Internet users and test their ability to resolve a DNS name where the only means of access is by using IPv6. This topic of the extent of end-user readiness to use IPv6 to access all forms of Internet services has been a topic that has had some considerable interest over many years, within in the larger scheme of transitioning the entire Internet to IPv6.

The question I would like to examine here is one of the extent of support for DNS over IPv6 in users' DNS resolver environments. In other words, if we placed an authoritative server on an IPv6-only

platform, what extent of the Internet's user population would be able to resolve a name that was served in such a manner?

Measuring the DNS

Firstly, a few words of caution. In measuring the DNS nothing is as straightforward as it sounds. When we talk about the DNS in this context, we are actually talking about the function of name resolution. And central to that is the concept of a "DNS Resolver".

Resolvers

So, let's ask the question: "What is a DNS Resolver?"

Oddly enough, there isn't a straightforward answer. We really don't understand what a "DNS resolver" is. It could be a single platform running an instance of DNS resolver code or it could be a collection of back-end DNS systems with some kind of front-end load distributor, or it could be a hybrid collection of servers with a set of semi-synchronized caches so they emulate a common cache and leverage that common cache in the operation of each server engine.

We've crudely classified DNS resolvers into *stub resolvers* that exist close to the end user in the end user's device, *recursive resolvers* that are meant to navigate through the DNS' distributed database to resolve queried names, and *forwarding resolvers* that pass on their queries to other resolvers rather than answering it themselves. There are more convoluted forms of hybrid behaviour that may be distributed across many multiple resolution systems. All of these are called *resolvers*.

Not all resolvers are equal, or are as significant as other resolvers. A resolver might have just one client, as is the DNS resolver in my local network, or it may have millions if not hundreds of millions of clients as is the case with some of the larger open DNS resolvers or of course anything in between. So, when we talk about DNS resolvers, it's actually quite challenging to understand exactly what we're talking about. Deriving a measurement statistic of the form of some percentage of DNS resolvers that support queries over IPv6 is a meaningless measurement in terms of assessing the utility of IPv6 in the DNS. This is why when we talk about the level of support for IPv6 in DNS resolution, the metric we'll be using is not a count of DNS resolvers per se, but a count of users, as a proportion of the total estimated user population.

Queries

Of course, if you thought that was a tough DNS question, there's another one too: "What is a query?"

It sounds like a silly question, but there is a point here. The resolution process used by the DNS to navigate through the DNS data structures causes a fan-out of queries where a single initial query to a recursive resolver causes that resolver to launch a sequence of discovery queries, as the resolver tries to understand where that requested information might lie within the framework of the distributed database that is the DNS.

The related issue is that by default the DNS uses UDP as its transport. UDP does not provide any end-to-end assurance, so the sender must wait for some period of time for a response, and if none if forthcoming in that time it has no choice but to retry the query. Resolvers do not behave uniformly, and implementations use their own timer selections to figure out when to re-query. The timer choice is a compromise between attempting to respond as rapidly as possible to the original DNS query and avoiding flooding the DNS with superfluous queries.

DNS queries have no hop count, so if a query is forwarded on from one resolver to another then there's no attached history of where the query comes from, and no reason code as to why the query has been forwarded. There's no context in the DNS that would allow us to determine the reason for any particular query, no hop count to detect looping conditions and no time-to-live to detect and remove "aged" queries. The result is that the DNS resolution tends to err on the side of profligacy in query generation.

Our measurements, taken at the authoritative server for a DNS name, show that some 25% to 30% of individual queries get repeated within the DNS (Figure 1).

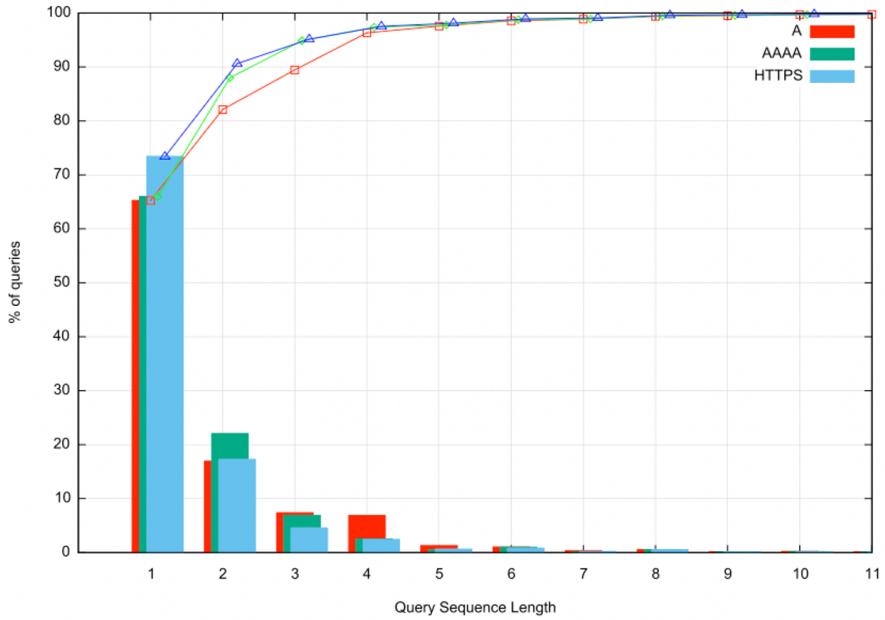

Figure 1 – DNS Query Repeat Profile

I suspect that part of the reason for this profligate query repetition within the DNS lies with the choice to use UDP as the default transport protocol. A resolver does not know when a query has been dropped, and the best strategy for a resolver is to fan out the query, to other authoritative servers, and in the case of a stub resolver to other recursive resolvers to compensate for unpredictable response times over a UDP transport.

APNIC's Measurement

Finally, let's have a look at APNIC's own measurement setup that is used to perform these kinds of measurements.

We use Google's advertising network to seed DNS queries inside online advertising campaigns, and use parameters in the ad campaign to get as broad a distribution of the ad as we can. Each ad contains a list of URLs, and a control script that tasks the browser to fetch each URL and then report back on the success (or failure) of each attempted fetch. DNS names that are part of each URL contain a name component that is unique to each and every individual user that has received an ad. That way the DNS recursive resolvers have no cached data and they are forced to query inward towards the authoritative name server to resolve the presented name. We observe these recursive-to-authoritative queries by instrument our authoritative name server and match those queries against the experiment platform's ad placement records.

Measuring DNS over IPv6 using Web Fetch

To perform this measurement, we scripted an ad that includes a URL that uses a unique DNS name and configured the authoritative server for that DNS name that can only be reached using IPv6.

The user can only resolve the DNS name. and thereby fetch the Web object, only if their name resolver setup supports DNS queries over IPv6. While the resolution of the DNS name relies on IPv6, the name itself is resolvable to both IPv4 (A) and IPv6 (AAAA) records.

The measurement is therefore quite straightforward. We measure the ad placement records, and the proportion of those records where the user performs a fetch of the IPv6 over DNS web object. The

proportion of successful fetches is therefore the measurement of the internet-wide capability of users to perform a DNS resolution using IPv6 This data is shown in Figure 2.

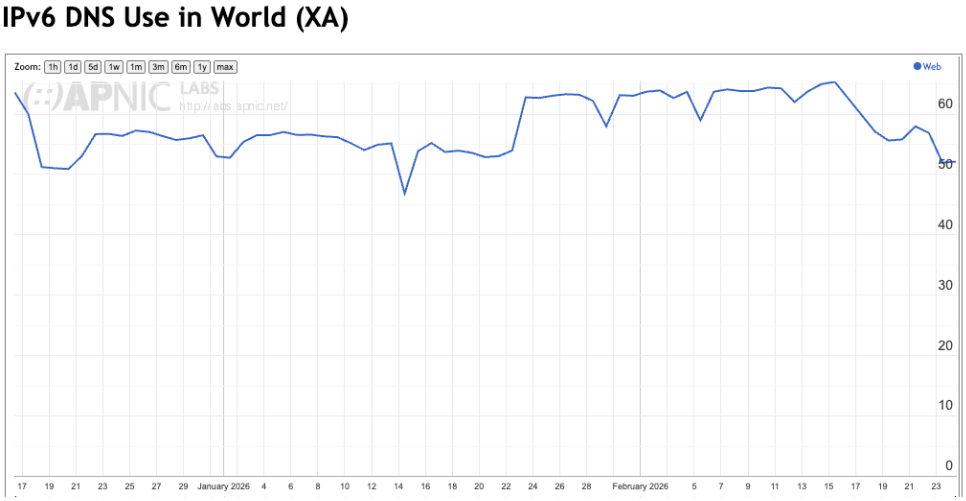

Figure 2 – Web-based Measurement of the capability of DNS resolution using IPv6

This appears to be a reasonable measurement. The result, namely that some 50% to 65% of users demonstrate their capability to resolve a DNS name that requires the use of DNS over IPv6.

We're done, right?

Well, maybe not. You see, there's couple of actions going on inside the user's browser that we need to be aware of. The browser is received a task to fetch a web object. It's given its name, a DNS name, that it firstly has to resolve. Only when that name has been resolved by the DNS is it then capable of performing the fetch. But what if the execution of the ad script is terminated before the script completes? The DNS task, which may have already been passed to a resolver will continue until completion, but when the result is passed back to the browser there will be no subsequent web fetch. Perhaps this result is undercounting.

Is there another way to perform this measurement of IPv6 capability in the DNS using only the DNS? If this was feasible, then as soon as the task was passed into the DNS the effort to resolve the name would continue until completion and would not be terminated early if the ad's control script was terminated.

Measuring DNS over IPv6 using the DNS

The answer to this question is that it is possible to perform this form of measurement completely within the DNS. The technique we use is one of "glueless delegation" in the DNS.

When a recursive resolver asks an authoritative server for a name that does not exist within the served zone, but is contained in a delegated zone, the nameserver will return a *referral response*.

Delegation in the DNS lists the name of the delegated zone, and the names of the nameservers that are authoritative for that zone. When an authoritative nameserver is queried for a name that is in a delegated zone, the nameserver will generate a referral response that contains the nameserver records in the Authority Section of the response. It will also contain the IP addresses of these nameservers in the Additional Section of the response ("glue records"). If these nameserver names are contained with the delegated zone then these glue records are essential, as there is no other way for the recursive resolver to resolve these names to their IP addresses. However, when the names are defined in a different zone that is unrelated to the queried zone then these glue records are not essential to the resolution. If the glue records are not present in a referral response, then the recursive resolver will need to suspend the primary

task of resolving the original name, and work on a sub-task of resolving the nameserver name. Once the nameserver name is resolved, if can use these IP addresses to resume the resolution of the original name.

This then provides the basis for an IPv6 capability test within the DNS, as shown in Figure 3

```
Zone: example.com

…
a       IN      NS      ns1.some.other.zone.
…


Zone: some.other.zone
.       IN      NS      ns0.some.other.zone.
ns0     IN      AAAA    2001:db8::1
…
ns1     IN      AAAA    2001:db8::2
…


Zone: a.example.com

.       IN      NS      ns1.some.other.zone.
.       IN      A       192.0.2.3
        IN      AAAA    2001:db8::3
```

Figure 3 – Example of a "Glueless Delegation"

In this case a recursive resolver attempting to resolve the name **`a.example.com`** would receive a referral response with the nameserver name of **`ns1.some.other.zone`** with no glue records. It would then need to suspend the task of resolving **`a.example.com`** and commence resolution of the nameserver name **`ns1.some.other.zone`**. Once this resolution is completed (with the IPv6 address **`2001:db8::2`**) it can then resume its original task and query this IPv6 address for the IP addresses of **`a.example.com`**. However, as the nameserver for **`some.other.zone`** is an IPv6-only nameserver, then the recursive nameserver can only pose this query if it supports DNS queries over IPv6, and can receive the IPv6-only DNS response.

We can perform both DNS-over-IPv6 measurements within the same ad script, testing the user's DNS environment using both methods. We expect the measurements to be reasonably well correlated, with the DNS measurement giving a slightly higher result. This is what we have observed (Figure 4). We can attribute the approximate 10% lower Web-based measurement to "lossy" transition from DNS resolution to a completed Web fetch.

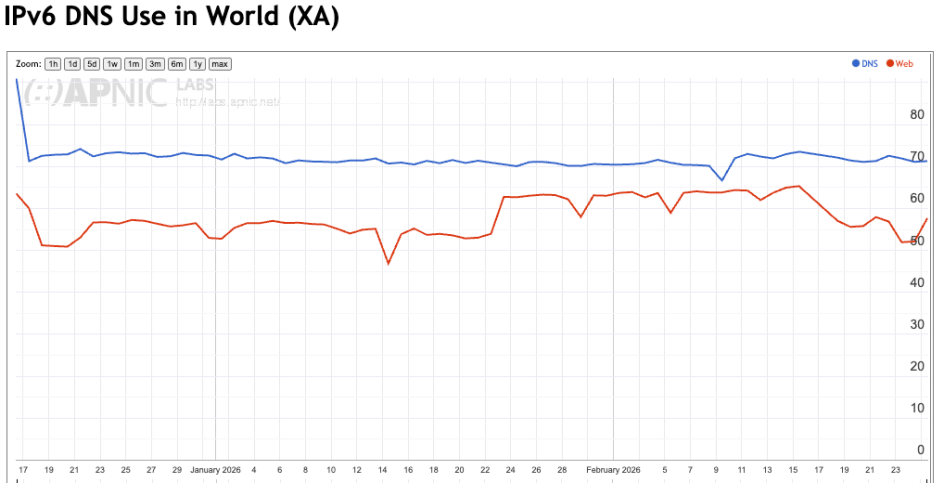

Figure 4 – Measurement of the capability of DNS resolution using IPv6 using DNS and Web measurements

Are we justified in making this assumption?

Measurement Anomalies

There are a number of countries in Africa that report a consistent, but anomalous behaviour in this measurement where the Web-based measurement is consiustently higher than the DNS-based measurement. A good example is Algeria (Figure 5), although a similar behaviour is apparent in Libya, Egypt, Mauritania, Liberia and a number of other mostly African countries (See https://stats.labs.apnic.net/v6odns).

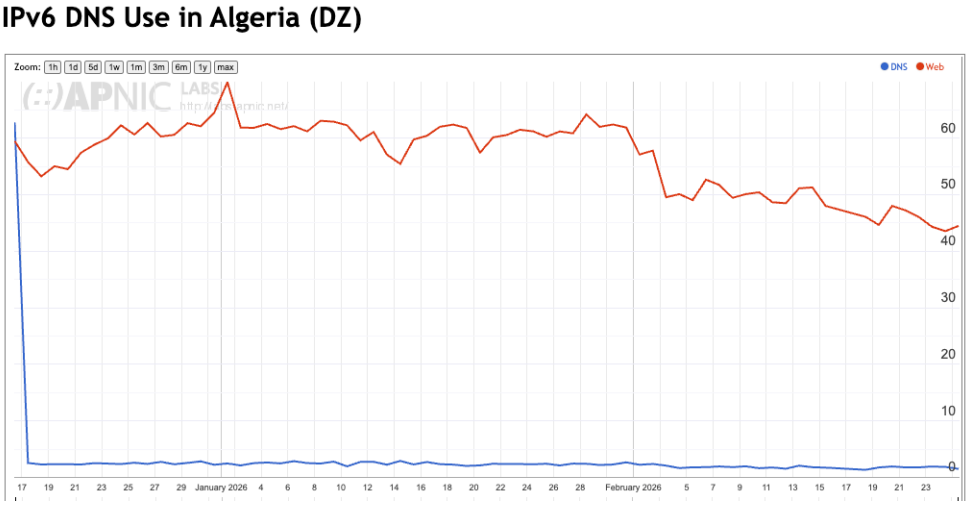

Figure 5 – Measurement of the capability of DNS resolution using IPv6 using DNS and Web measurements in Algeria

There are also countries which have the opposite anomalous measurement result, where the DNS measurement is far higher than just 10% more than the Web measurement. The result for Ethopia is a good example of this opposite anomalous result (Figure 6).

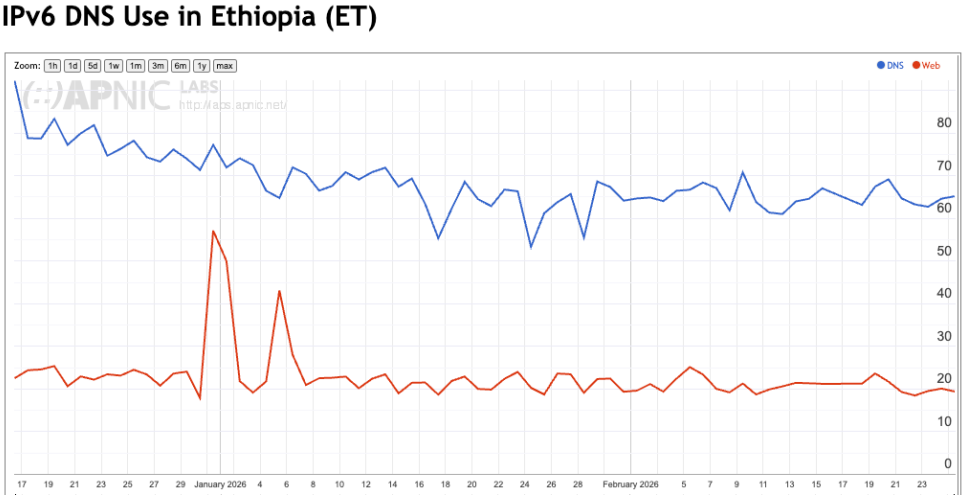

Figure 6 – Measurement of the capability of DNS resolution using IPv6 using DNS and Web measurements in Ethopia

Another result is also worth mentioning can be seen in the country measurements for Germany, where the daily WEB-based and DNS-based measurements correlate very well. The level of "web-loss" where the DNS result is not converted into a visible web fetch is extremely low across the entire sampled population in that country (Figure 7).

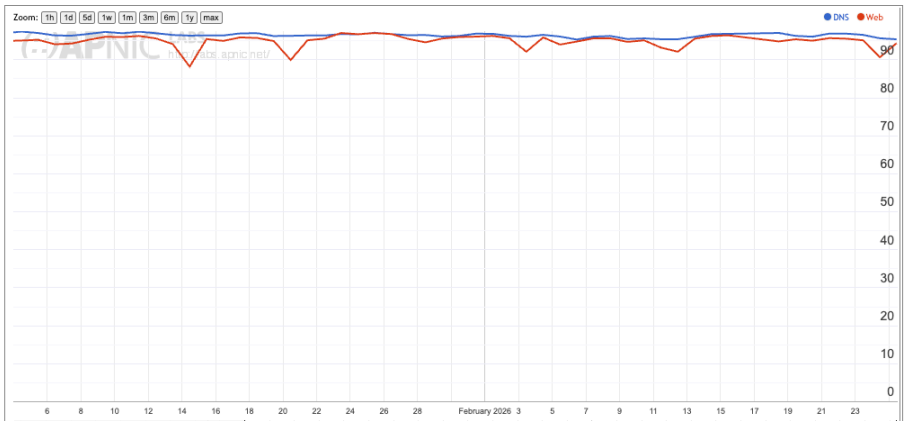

Figure 7 – Measurement of the capability of DNS resolution using IPv6 using DNS and Web measurements in Germany

These results are challenging to explain within the framework of a conventional model of the DNS infrastructure since they are attempting to measure precisely the same behaviour using the same users, testing the same DNS resolvers.

One way to compare these two methodologies is to look at each network and compare the DNS and Web measurements. This comparison is shown in Figure 8.

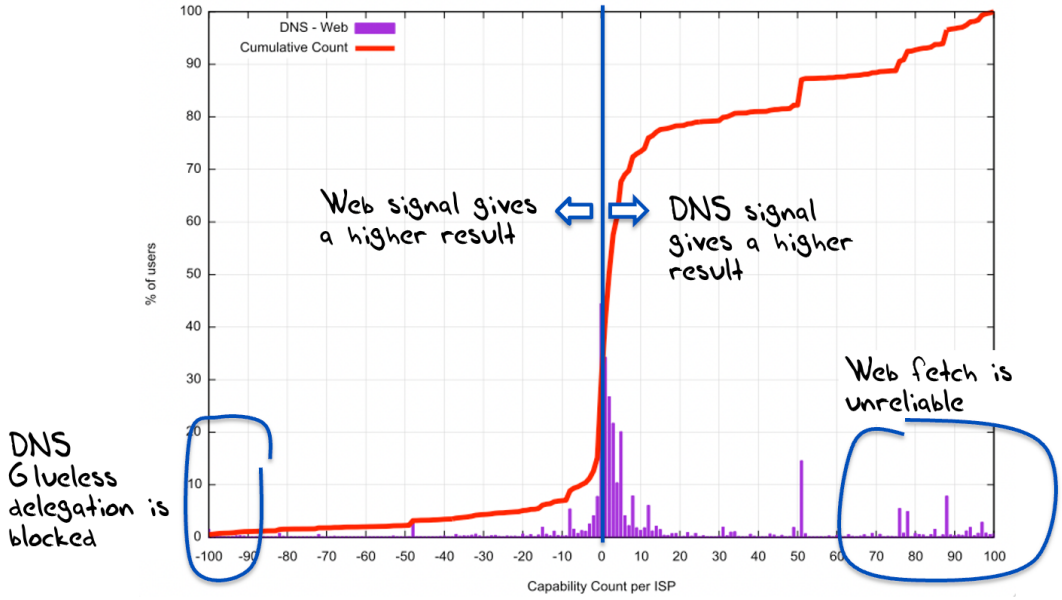

Figure 8 – Comparison of Measurement of the capability of DNS resolution using IPv6 using DNS and Web

For less than 8% of measured endpoints the DNS measurement was more than 10% smaller than the Web measurement, while the opposite was the case for 30% of the measured endpoints. For the remaining 62% of measurement samples, the two measurements were within 10% of each other.

Figure 9 shows a breakdown of the Web and DNS results on a daily basis. In 4% of cases the Web is giving a positive result showing support for DNS over IPv6, while the DNS does not provide a positive result. In 26% of cases the DNS measurement provides a positive result, while the Web result does not. In 45% of cases both the Web and the DNS provide positive measurements.

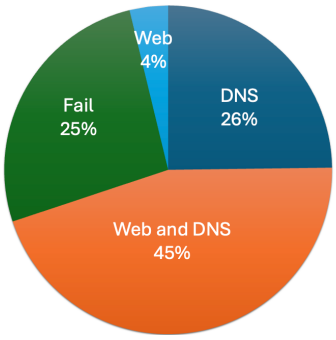

Figure 9 – Breakdown of Web and DNS measurement results

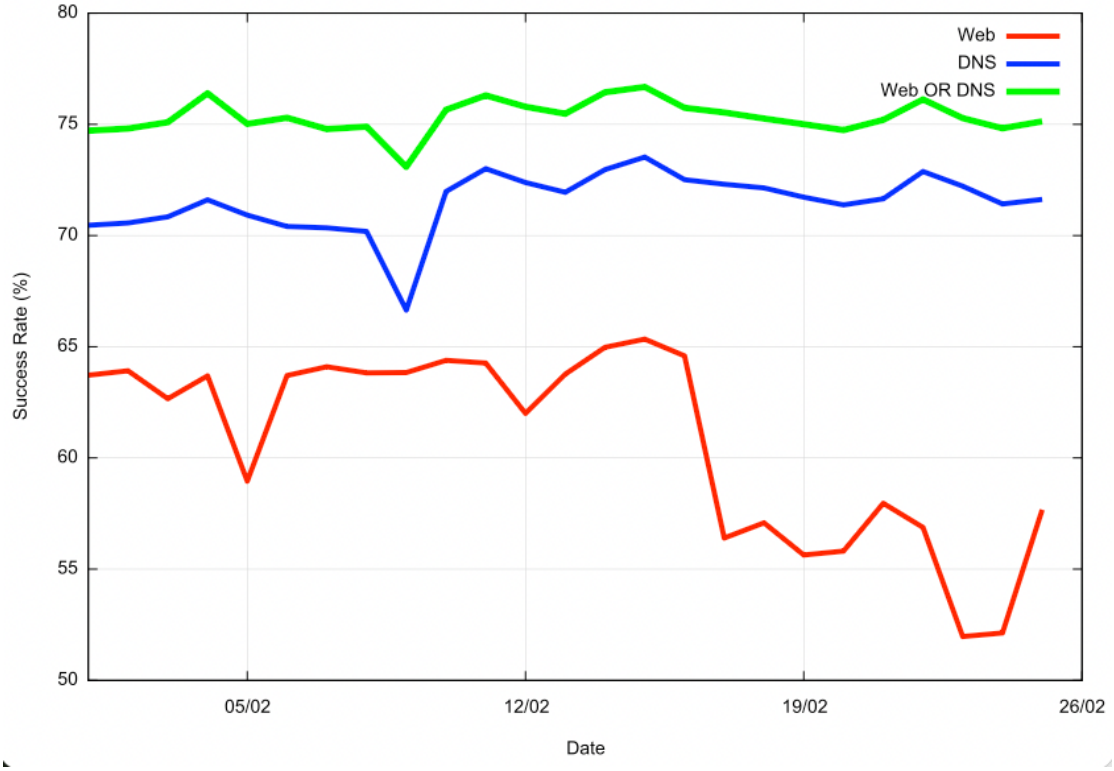

Figure 10 – Daily Series of Results

The results are certainly interesting. The DNS-only measurement is more robust in that we see the DNS operate through the glueless delegation in a more reliable manner. There are, however, networks where there appears to be blocking of resolving domain names where there are no glue records (predominately in North Africa), and this blocking behaviour appears to affect some 3% of the total user count.

The overall result is that we see an overall capability of some 75% of users who are able to use DNS over IPv6.

Disclaimer

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

Author

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

www.potaroo.net