A Systematic Study on the Identification of DNS Query Forwarding by Recursive Resolvers

About the power to resolve DNS queries

Mathijs van de Zande
University of Twente
P.O. Box 217, 7500 AE Enschede
The Netherlands
m.j.vandezande@student.utwente.nl

ABSTRACT

The domain name system (DNS) is one of the most frequently used protocols and systems that lays the foundation of the Internet. Because of the minimal definition of the protocol and many different applications and extensions, a wide range of resolvers exists leading to parties exploiting the many possibilities and as a possible result - security flaws. Most of the research in the area of the DNS is done on security, performance and latency, and not on the identification of the actual resolvers and their organisations. The research most closely related to this research is focused on the identification of resolvers and, more specifically, the resolver software and operating system. This research designed an identification and classification strategy to identify the organisation responsible for the actual resolver. This was extended by the designing, testing and implementation of a classification framework - a framework that uses the strategy to test whether the targeted resolver organisation is also the actual resolving organisation. Consequently, this research resulted in a classification strategy and framework giving an impetus for future research that will be conducted in the area of DNS resolver identification.

1. INTRODUCTION

The Domain Name System (DNS) is one of the most fundamental building blocks of the contemporary version of the Internet. By translating domain names to Internet Protocol(IP) addresses in a pattern that is understandable for humans, the DNS protocol is being used in every TCP/IP flow and at least once for every Web page view. Regardless of its simple rules and minimal specified protocol, the DNS grows to be more complex every day[14]. This complexity has its roots in multiple applications and extensions of the protocol, but nevertheless, the implementation does not always go in line with the fundamental ideas of the protocol. Some of these adaptations can be used for, inter alia, advertisement pages instead of a NXDOMAIN response, the response that according to the protocol should be send back when a domain name does not exist. Another example is the research on the identification and characterisation of users, using the behaviour tracking to resolve a users IP address[15, 4, 13]. The use of these applications and extensions bear the potential to result in privacy concerns. The moment that a third party DNS service like GoogleDNS[3] or openDNS[9] are chosen as a low-cost alternative for Internet Service Providers(ISPs) and disclosure of users browse behaviour is possible, it becomes interesting to know who is actually resolving the requests. These actual resolvers are and will be in the position to conduct analyses on the behaviour and IP addresses of users for tracking as well as classification.

The resolver plays a key role in the implementation of the system and therefore in all issues that were just addressed. Despite this fact, there has not been much research on who is actually resolving our queries. Therefore, the research at hand focuses on the identification of resolvers, more specifically on the organisations that are in control of the resolvers. This strategy is used for the implementation of a classification framework. A framework that uses this strategy for the classification of resolvers, in order to offer an insight in the amount of organisations that actually forward queries and the accompanying security risks being caused by the fact that third party resolvers resolve the queries. Because the knowledge on who is answering DNS queries and whether queries are forwarded is key for estimating privacy risks in future research.

A research on who is resolving our DNS queries is getting more and more compelling, being a consequence of the potential changes and risks in the functioning of the system. Despite of creating awareness on the critical and important role of the system and the future risks of misuse and maintaining users privacy, there has not been conducted much research on the identification of the full recursive resolvers. Most of the research conducted on DNS has been done on the performance of the network and the system, and is often focused on the system’s latency and the high level authority servers like the root and generic Top Level Domain(gTLD) servers. The research most closely related is the research on the characterisation of resolvers and their clients [11]. Shue characterised resolvers in several classes based on their response behaviour. This was extended by a research that connects clients to the resolver identification classes mentioned earlier.

The results gathered in this research are dividable in three aspects known as the identification and classification strategy to retrieve information about the expected and actual resolver. A framework as proof of concept using this strategy to identify resolvers. And the results of the measure-
ments that should give an insight in the status quo on the resolvers question.

Starting with an elaboration on the literature study on how typical DNS requests are resolved. This paper will follow up on this study by the design and test of strategy to identify actual resolvers and classify them based on the measurement topology. The next aspect is the building of a framework as a proof of concept for this classification strategy. And a large scale measurement to test the performance of the framework and give an insight on who is actually resolving our queries.

2. THE FUNCTIONING OF DNS
The Domain Name System is a hierarchical and distributed database that maps domain names that are static and easily to remember for people to IP addresses, that may be dynamic and are used for routing in networks. The domain names themselves are structured as a hierarchical series of zone names separated by dots. Each zone name represents an authority level for which at least one authoritative name server is responsible, possibly in a higher authority level. By traversing through the zone names in order of descending hierarchy the resolvers find the IP address to which the domain name referred.

2.1 The protocol
When a requestor, user or application queries a resolver to resolve a domain name, this first resolver is in most cases a built-in resolver in the operating system or browser. This first stub resolver, from now on called the local resolver, will query the resolver of the ISP to retrieve the correct IP address. The second resolver is the recursive resolver, which means it will try to resolve the IP address by asking a next resolver. This third resolver might be a recursive resolver as well. In the simplest case the third resolver is an authoritative name server, one of the servers that is responsible for a zone name and authority level. An authoritative name server replies with an IP address of the first resolver of the next authority level or the IP address of the domain requested for. The recursive resolver that is doing the look ups in between the local resolver and the authoritative name servers will be called the full recursive resolver from now on.

In this research and the use of the framework both the local resolver and the lowest authority authoritative name server will be controlled by the measurements. Both nodes are controlled data points in this research.

In the resolving process of a typical query, every authoritative name server that is queried reveals a piece of information about the domain that is being queried for. This information consists of an IP address to the domain queried for or a redirection to another name server on a lower authority level with a higher number that is responsible for that zone name. For example, a typical resolver and a query for subdomain.example.tld. would first query the root server. This root server will respond by sending the IP address of the authoritative name server responsible for .tld. That server will reply with an IP address that links to the authoritative name server responsible for example.tld. When available, this authoritative name server will send the address of the next subdomain or the domain requested. When the domain is not available or there is no recursive server to link to, the server will send a NXDOMAIN response which stands for the non existence of the domain. This error message will tell other servers, operating systems and browsers that the page does not exist, so the user or host can take appropriate measures.

2.2 The system
The actual Domain Name System is often more complex. This is because of caching, load balancing or multiple recursive resolvers that are in between two of any of the controlled data points shown in figure 1(a). To give a basic idea of each of these complexity factors a short explanation is given here.

Caching is the process of storing recently or frequently used domain to ip mappings at a resolver. This caching can be done at all resolver levels like the local resolver and recursive resolvers. Doing so at all resolver levels increases the performance of the system significantly because less queries need to be resolved.

Load balancing is a process used for load distribution over for example a group of servers. By mapping one domain name to multiple IP addresses, the hosts querying this IP address are spread over the instances. This way it is possible for sites with a lot of visitors to balance the visitors over multiple servers enabling them to scale by adding more machines instead of a more powerful machine. The term load balancing in DNS either refers to the load balancing settings in a resolver, the mapping of one domain to multiple IP addresses or it refers to the load balancing of the resolvers themselves. Many resolvers have a backup server in case one of the servers stops working, which in many cases is used to take 50% of the requests so both resolvers share the load.

Recursiveness of resolvers is one of the consequences of a loosely specified protocol. The protocol defines a tree structure, a hierarchy and the algorithm used to resolve queries. Because this system should be highly scalable, the protocol allows recursive resolvers to be added at almost any place in the resolving chain. This allows companies to host their own DNS service resolve local DNS addresses faster and forward external queries, or just for a faster and more reliable performance.
3. RELATED WORK

There has been done a wide range of research on the topic of the DNS. From caching to security, and from security to server behaviour, each research focuses on another aspect of the system or its protocol. The research most similar to this research is the research done by C. A. Shue, that will be discussed in this section, just like the research that has been conducted on the security risks of the system and other related research.

Shue conducted a research on the characterisation and classification of resolver classes based on the way these resolvers handled their requests[11]. This way, Shue was able to identify a resolver’s operating system and resolver software, like for example BIND by testing on small differences in responses. It is notable that Shue’s research focuses on the internal software of the resolver instead of the organisations responsible for them. This research at hand focuses on an identification problem on a higher organisational level, by classification of resolvers and their organisations.

D. Hermann conducted research on three techniques that allow resolvers to passively track dynamic IP addresses that are based on characteristic behavioural patterns. This way he could identify about 85% of the users despite a dynamic IP address[4]. This research concerns the privacy of the users of the DNS and confirms the importance on the knowledge of what organisation is resolving the requests. However, not only research shows that the resolver is and might get a better position to analyse users. For example the proposal of van der Gaast about extending the DNS request by adding the client IP address to the query[13]. Besides the research and proposals, also companies and organisations exploit the freedom of the system and its protocol by for example, making money on pages filled with advertisements instead of responding with the protocol defined NXDOMAIN[15]. Further related works are primarily based on the performance of the system and the privacy risks or best counter measures of DDOS and DNS amplification attacks[8, 1].

4. METHODOLOGY

This research focuses on the classification of resolver organisations and their policies by the identification of the query routes and organisations in control. In doing so, a framework has been designed and implemented that finds changes in the queries sent to and received from resolvers. These changes in combination with extra information from IP to ASN look-ups give an insight in the route a query takes in order to get resolved. This section extends the introductory explanation of the DNS by an elaboration on the resolver classification strategy to identify resolver organisations by ASN and classify a possible forwarding policies by comparing data from start- and end node. Followed by a description of the design of the framework and the measurement techniques, this section offers a total overview of the methodology of this research.

4.1 What nodes are in control

As described earlier, the DNS protocol leaves room for a wide variety of implementations and topologies. This variety leads to highly scalable, dynamic but complex system. To make sure the framework works for all of these complex topologies and implementations, this research stated requirements for the selection of controlled data points. The requirements used in this research are:

**Controllable**: The node should be controllable by the framework in terms of the specification of the requests, data collection and timing.

**Flexible**: The nodes should be part of all topologies, query routes, in such a way that the nodes are not influenced by the complexity of the routes.

**Closed**: The nodes and request sent by those nodes should influence the usual traffic within the system as little as possible.

Based on these requirements, the following nodes were selected and used in the implementation and the measurement:

**The local resolver**, where every query has a requestor and the local resolver gives a lot of opportunities to tweak the requests for better performance or extra measurements. For example, the opportunity to send a NXDOMAIN response back or an IP address is.

Secondly **the authoritative name server** being responsible for the domain name. Since the authoritative name server will be the end node of the resolving process and therefore the receiving node of the actual query. An authoritative name server with a higher authority level, like the root or gTLD servers, would make the analysis of the data much more difficult coming along with privacy concerns and noise because of the large amounts of data. Another problem with an authoritative name server with a higher authority level is caching; the lower the authority level of a zone, the less likely it is that the domain name is cached in transit.

By controlling both the start- and end node, the research focuses on the route highlighted in figure 2(a). A request following this route would start at the local resolver of a probe and will arrive via zero or more recursive resolvers at the full recursive resolver that is querying the authoritative name server irrespective of the route’s complexity. Because start- and endpoint are part of this route the creation and receiving of requests is fully controlled by the measurements. The choice for these nodes make this route the most adaptable for probe measurements and less likely to become subject of any form of complexity, for example in the sense of caching and load balancing.

The two endpoints are also the nodes that contain, receive or create the most data. These two data points include data of all incoming and outgoing queries and responses to and from the recursive resolvers used in the resolving process. By extending this data with the information of the local resolvers, or probes in case of the framework, it is possible to use for example the local resolvers ASN and geolocation as well.

4.2 The classification of resolver organisations

Of the available data in the start- and end node, the IP addresses are most interesting for identifying resolver organisations. Although IP addresses do not necessarily remain unique or the same over time, they are distributed
by an AS as an identifier for a certain device. This AS is especially of interest for this research because each ASN assigned to an AS has a large organisation or ISP that is responsible for the routing and distribution of IP addresses. More important is that ISPs are responsible for the Internet connection and therefore the resolving of DNS requests.

This makes the ASN a fit measure to identify the organisations that are expected to resolve DNS requests for the local resolvers. And by doing a look-up on the source IP address of the incoming query at the end node, the ASN of the resolver that actually resolved the query can be determined. By this approach it should be possible to distinguish ISPs from large companies and third party resolvers. And classify the forwarding policy of the targeted resolver by comparing the targeted resolver and the actual resolver. Notable is that local resolvers can query multiple resolvers at about the same time, which may result in queries from multiple ASNs at the authoritative name server. Therefore, measurement results are given labels for the different route properties that are identified, to classify the cases of the expected resolver resolves, a third party resolves or a combination of both resolve the query.

4.3 Framework
The implementation of the framework is key to this research as being the proof of the concept and tool to create the insight in the status quo. This research is aiming at an initial solution that is capable of working for larger data sets and network tests as well. This measurement framework consist of the following parts including the nodes described earlier.

The start node is a local resolver that sends unique domain name request to the framework’s authoritative name server. In this framework, the start nodes are implemented as probes of the RIPE ATLAS framework[7] that send DNS requests using their own ISP to a subdomain, domain name combination, uniquely specified per probe. By doing this, the receiving name server can identify the probe where the request originates from, and enable the data analysis part of the framework to match both start- and end node. Beside the matching the unique and low authority level domain name ensures that the query can not be resolved from cache because it has never been queried for.

Besides the start node, a server is needed to manage the incoming data and to control the probes. This part of the design is the center of the framework because it controls the data, probes and measurements during a measurement run. As it is the controller, it could run on any server, as for example with the end node.

The end node is an authoritative name server that is listening for DNS requests. When it receives a request, it stores the important data and sends a NXDOMAIN response back to the resolver. The important data at the authoritative name server is the source IP address of the resolver which is needed for the identification of the full recursive resolver, by this the domain name queried for and based on the subdomain of this domain name, the probe it originates from.

At last, the design uses a program to analyse the data. This is done by look-ups for the IP addresses and comparing and labeling the queries that were stored earlier in the process. The comparing and labeling is done based on the ASN’s routes and uses the labels that are described further in this section. The exact implementation details and design choices are explicited in the results section 5.

4.4 Measurements
The measurements used in this research are semi-automated test for proving the correctness, performance and working of the framework, the identification and classification strategy and in the end for creating the insight in the status quo of actual resolvers.

The first tests of the framework were small scale tests, using 3 to 50 probes or emulated requests to test the implementation and data. Followed by a test of the same scale to test the feasibility of the identification and classification strategy, designed for this research, and its implementation in the framework.

Important aspects of these tests for testing the performance of the framework were the data collection and parsing, the look-ups from IP to ASN and most importantly the plausibility of the results. For the tests to check the strategy, known hosts and local resolvers were used to emulate realistic requests and test the correctness of the labels.

At last the framework did one large scale test by requesting approximately 7500 probes to query a uniquely specified domain name. This domain name was registered under the authoritative name server of the framework. All data that was stored during this measurement of about a minute was merged and combined for analysis. This
analysis consists of the look-ups and labeling described in section 4.3. From these results and by using all the probes available within the RIPE ATLAS network, this measurement should create an insight in the division of targeted and actual resolvers.

5. RESULTS
The results of this research are divided in two aspects. The implementation details and design choices that were made during the implementation of the design described in section 4.3, followed by the results of the measurements described in section 4.4. These measurements are split in the small scale measurements that tested the correctness and performance of the implementation and the classification strategy and its limitations. And the large scale measurements primarily used for the creation of an insight in the usage of recursive resolvers that forward queries to a third party based on the measurements performed by the framework is given.

5.1 Framework implementations details and design choices
The framework is one of the key results of the research, being the proof of concept for the identification and classification technique that is subject of this research. The design, as described in the methodology, served as the basis for the implementation of the framework and needed several design choices on the go. Those design choice originated from the wide variety of implementations of the system and protocol, as well as the limitations caused by the use of libraries and frameworks.

5.1.1 Controller
One of the most important parts of the framework is the controller, which is responsible for the data collection and probe management. For these responsibilities, the controller used probes of the RIPE ATLAS [7] network to generate realistic domain name requests from all over the world using many different ASs. For controlling the probes the python library Cousteau[2] was used. For data collection and parsing the Sagan library [10] was used.

When a measurement was run, the controller loaded the probes from the most up-to-date .json file of the RIPE platform[6]. From this data set the controller selected the probes that were recently active, publicly available and were IPv4 compatible. This ensured that all the probes are working or did work recently, are publicly accessible and are able to use the IPv4 protocol. This resulted in a higher response rate and a more realistic result in measurements because IPv4 is still better and more extensively used than IPv6.

After loading the probe information, the controller created measurements requesting 500 probes at a time to query a unique domain name. This domain name consisted of the probe ID followed by a timestamp, a measurement ID and the domain under our control, in this case and for descriptive purposes: domainname.tld. Taking 13.146589798.1.bachref.domainname.tld as an example. Besides the request to query a specific domain name, the probes were requested to append both the q- and abuf to the request for later analysis.

Using the Sagan library, the controller checked the measurements whether they are running, whether they are finished and for any error messages. When the measurements finished, the controller collected all the necessary data from RIPE ATLAS and stored this raw data in a measurements data set.

5.1.2 Listener and authoritative name server
The implementation of the listener became more complex than the original design. Listening for DNS requests on port 53 and storing them appeared to be insufficient in combination with the measurements of the RIPE ATLAS platform. Those measurements need a reply of the targeted resolver to demand credits and provide measurement data. This resulted in the implementation of an authoritative name server that handles incoming request by storing them and by sending a response message. To reduce the complexity of the implementation the twisted library[12] was used to send and receive requests. Now the server stores the important data of incoming requests and sends a response. The authoritative name server could respond with actual IP addresses or send an error message.

Because this is the defined response in the protocol, if a domain name does not exist, the server replies by sending a NXDOMAIN response back to the requestor. Besides the definition of the protocol, a wide range of incorrect routed request resulted in a second motivation to use the NXDOMAIN response. This is done to reduce the amplification factor in possible DNS amplification attacks, a NXDOMAIN response does not have any retries, and therefore no amplification factor.

5.1.3 Data collection and analyses
The merging and analysis of the collected and generated data is done by a third program. This program collects data from the three data sets, probes, measurements and listener and extends this data by doing IP to ASN lookups. For this measurements the ipwhois library[5] is used which uses several servers to get the information that belongs to this IP address. In case no IP to ASN matching exists, the measurement is omitted because of an exceptional case of no interest for this research.

For every probe in the measurement, the destination IP addresses are matched to their ASNs and compared to the ASN of the probe itself to determine whether the probe uses its ISP to do the look-ups, or an third party resolver by settings. Then the source IP addresses of the queries received by the listener are matched to their ASNs and compared to ASNs that are expected to resolve the query. Based on this comparison, the program assigns labels to the three different cases.

**EXT:** means the resolver used is of another AS then the probe itself.

**OWN:** means this probe used a resolver of its own AS.

**OWNAO:** means this probe used a resolver of its own AS among resolvers of other ASs.

**NA:** means not applicable due to incomplete data, the use of a private network or the use of an AS that the framework was unable to identify.

Based on the combination of labels given to the target and source resolver, a resolver was classified as a forwarding resolver. The following paragraph will discuss the labeling details in more dept.
5.2 Performance measurement
The performance measurements focused on different aspects of the implementation of the framework. Aspects like data collection and message parsing were tested during those measurements. But also the IP address to ASN look-up is an important part of these measurements. From the moment all units of the framework worked properly and produced the expected results, the framework was tested and stored. The probes that were requested for this measurement were registered at 805 different ASs, used both the KPN and Google look-up services where a phone using Tele2 used only its own DNS service. A selection from the results of one of the small scale test that verifies that some organisations use a third party resolver to support their own services. Besides the classification, this test verifies the correctness of the look-ups for the ASNs and the data collection and labeling in the data merging and analysis part. This is an example of three independent probes or devices that use a uniquely specified domain name and are identified per AS, or multiple ASs. These ASNs were checked using the information about the providers.

5.3 Classification measurement
The classification strategy uses the route, described in section 4.1, and the ASNs to identify organisations responsible for the resolvers that, are the target resolvers for the local resolvers and are queried by the authoritative name server. By comparing these ASNs, target and source resolvers were given a label as described in section 5.1.3. Each tuple of results in a classification of the source resolver. For example, the labels EXT and OWN mean that the probe uses a resolver from another AS to send its requests to and this targeted resolver queries the authoritative name server controlled by the framework without using another AS. Or the labels OWN and OWNNO mean that the probe queries a resolver in its own AS for resolving but the queries are resolved by resolver in its own and other ASs. The small scale tests described in the previous paragraph were used to verify the strategy after they were proven to work correctly. The example from the same paragraph shows the correct working of both the framework and the classification method.

Although this strategy worked in the small scale test and for the larger companies and ISPs, a lot of the probes in the final measurement used an IP address from a private network to resolve their domains. Those probe measurements were omitted because of possible user settings that may distort the general outcome. Using the these measurements would result in one dimensional measurement of this labeling step on the measurements resulted in the outcome shown in table 2. It can be seen from this table, 100% of the measurements performed by the framework resulted in the label OWN, which means that all of the resolvers targeted by the probes used resolved the queries themselves or used an other resolver within the AS. In any case is only one organisation responsible for the services and data analysis performed by the resolvers in this AS.

5.4 Large scale measurement
After the small scale measurements, the framework was ready to perform the large scale tests for creating the insight on the status quo of actual resolvers. This test resulted in outcomes different from expectations. First of all, the 7167 probes that where used to query the frameworks authoritative name server only 1507 probes were identified and stored. The probes that were requested for this measurement were registered at 805 different ASs, used personal settings, the AS’s DNS service, possibly among other resolvers, or an other external resolver to resolve their queries. These categories are given the labels described earlier. In the table 1, the exact numbers and percentages of the target resolvers are presented with their associated label.

<table>
<thead>
<tr>
<th>Label</th>
<th>Number</th>
<th>Percentage</th>
</tr>
</thead>
<tbody>
<tr>
<td>EXT</td>
<td>379</td>
<td>25%</td>
</tr>
<tr>
<td>OWN</td>
<td>990</td>
<td>66%</td>
</tr>
<tr>
<td>OWNNO</td>
<td>71</td>
<td>5%</td>
</tr>
<tr>
<td>NA</td>
<td>67</td>
<td>4%</td>
</tr>
</tbody>
</table>

Table 1. Target resolvers used by probes

For the classification of the resolvers that send their request to the authoritative name server comparable labels were used except the NA label because this check was performed together with the destination resolver. The results of this labeling step on the measurements resulted in the outcome shown in table 2. It can be seen from this table, 100% of the measurements performed by the framework resulted in the label OWN, which means that all of the resolvers targeted by the probes used resolved the queries themselves or used an other resolver within the AS. In any case is only one organisation responsible for the services and data analysis performed by the resolvers in this AS.

<table>
<thead>
<tr>
<th>Label</th>
<th>Number</th>
<th>Percentage</th>
</tr>
</thead>
<tbody>
<tr>
<td>EXT</td>
<td>0</td>
<td>0%</td>
</tr>
<tr>
<td>OWN</td>
<td>1507</td>
<td>100%</td>
</tr>
<tr>
<td>OWNNO</td>
<td>0</td>
<td>0%</td>
</tr>
</tbody>
</table>

Table 2. Distribution of source and actual resolvers

5.5 Evaluation of results and performance
The results of this research can be split into two separate parts. One part, the working of the framework and its identification and classification strategy and secondly, the creation of an insight in the actual resolver question key to this research.

The identification of resolving organisations based on their AS is a strategy that works in almost all cases and, when this is not the case, these measurements are easily identified and omitted. But also the classification of actual resolvers based on the ASNs and the labels is an easy and effective way of classifying what organisation is resolving the requests. Different from the expectations, the large scale measurements resulted in a 100% score which implies no resolver organisation forwarded its queries. What causes this is, is probably to be found in the implementation of the framework or the selection of probes.

The small scale test performed in many aspects exactly as expected. The tests during the implementation of the design ensured that all aspects of the implementation work as expected. This applies for the correctness of data for the labeling and analysis, as well as the implementation of the external libraries and platforms, like for example RIPE ATLAS, used for the implementation of the framework.

When analysing the results of the large scale test two interesting results emerged. The first result: only 1507 of the 7167 probes were measured. What exactly caused this is impossible to define, and might be caused by a variety of reasons. That only 1507 of the probe measurement were
qualified for this research might be caused by a discrepancy in the amount of results gathered by the probes and the request stored by the authoritative name server. A look on both sides of the results showed that all the measurements from the RIPE ATLAS platform took place as expected and generated results for all the probes. On the contrary, the authoritative name server performed different from its expectations. As to be seen from the limited queries it stored and the many SERVFAIL errors in the probe responses in the RIPE ATLAS results. Apparently the implementation of the authoritative name server was not able to receive, store and answer more than 1500+ queries within a couple of minutes.

But also a second result of the large scale measurement can be perceived as against expectations, where the small scale test, using between the 3 and 50 devices with known providers, showed cases where a third party was used as a supplement or as the primary DNS service for a provider. This is in contrast to the large scale measurement with a 100% score on the OWN label. Which means that there exists a one-to-one mapping between the destination and source AS’s, and that all probes that are measured use a specific resolver, user specified or ISP provided. It is likely that the requests that got in the measurements were the first 1500 requests that got through because of a low latency. This is one of the possible explanations, but one that is supported by the top 10 ASs that were responsible for the probes, see table 3 that shows this top 10 and their ASs. Those probes were served by ASs that were geographically close or known for their quality networks.

<table>
<thead>
<tr>
<th>Rank</th>
<th>ASN</th>
<th>Description</th>
<th>Amount</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>7922</td>
<td>WASHINGTON-9</td>
<td>50</td>
</tr>
<tr>
<td>2</td>
<td>12322</td>
<td>FR-PROXAD</td>
<td>39</td>
</tr>
<tr>
<td>3</td>
<td>6830</td>
<td>UPC-INFRA</td>
<td>19</td>
</tr>
<tr>
<td>4</td>
<td>3215</td>
<td>FR-OLEANEBLOCK-9</td>
<td>17</td>
</tr>
<tr>
<td>5</td>
<td>3320</td>
<td>DTAG-DIAL15</td>
<td>16</td>
</tr>
<tr>
<td>6</td>
<td>3265</td>
<td>NL-XS4ALL-970509</td>
<td>13</td>
</tr>
<tr>
<td>7</td>
<td>9143</td>
<td>ZIGGO-INFRA-2-ZL-DNS</td>
<td>12</td>
</tr>
<tr>
<td>8</td>
<td>577</td>
<td>NETBLK-WORLDLINX</td>
<td>11</td>
</tr>
<tr>
<td>9</td>
<td>31334</td>
<td>KABEL-DEUTSCHLAND-11</td>
<td>10</td>
</tr>
<tr>
<td>10</td>
<td>701</td>
<td>VIS-BLOCK</td>
<td>9</td>
</tr>
</tbody>
</table>

Table 3. Top 10 probe ASs

Another explanation that might be the cause of this result can be found in the use of the RIPE ATLAS probes. These probes are mainly installed at universities, server centres and such, which means that the selection of probes is by no means a realistic representation of the average Internet connection. Although the probes are ideal for a measurement like the actual resolver test but focused on a specific, well represented AS.

A last point of concern are the IP to ASN look-ups. In the current implementation, the ipwhois library is used for doing those look-ups. In case this library does incorrect look-ups or only the high-level ASs, the classification strategy used in this framework will be more likely to give the label OWN. Even though the library was also used in the successful small scale tests the results of the framework are highly dependable on correct results of the look-ups.

6. CONCLUSION

The working of DNS the full path of a query is described from local resolver to the authoritative name server, that is responding with the awaited answer, of a typical DNS query. Then the identification and classification strategy by using the ASNs of resolvers was proved to be successful, although dependent on the quality of the IP to ASN look-ups. By implementing the strategy in the data merging and analysis part of a framework, that request probes to query an authoritative name server and combines the data from both probe and server.

The classification strategy used in this framework gives a clear picture of the query topology, to the extend of the first and the last node in between the local resolver and the authoritative name server. Often only one or two nodes represent the path from local resolver to authoritative name server and in most of the cases the last node operates as the full recursive resolver. Based on this element the classification of the organisations responsible for the target resolver and source resolver should be sufficient to identify the actual resolving organisation. Designing and implementing the identification and classification methods in a framework that uses these methods to identify the resolving organisations, was key to this research as a proof of concept of the classification technique. This framework and strategy were tested extensively and were proven to work in small scale tests of a maximum of 50 probes or devices. Although, the small scale tests showed that some ISPs use a third party resolver as a supplement or substitute for their DNS services, the large scale measurements did not have this expected outcome. What exactly causes this difference in outcome remains unclear from the measurements, apart from the suggestions and discussion in the previous section, and is thereby a topic for further research.

Although, not all aspects of this research performed as expected, this research is a step in the direction of the identification of the actual resolving organisation, an organisation that we as users of the Domain Name System give a lot of power. The power in the sense of determining our information access but also in the sense of the large amounts of personal information we share unwittingly with our resolvers. In the research field of DNS, this work introduces a new view on resolvers, resolvers that were always seen as the attendants of the system might have a different, less supportive, role in the future.

6.1 Future work

As this research is an initial research by designing and implementing a proof of concept, a lot of aspects require a closer look in future works. For example, the implementation might need a performance update more specifically the authoritative name server that has to receive, store and reply to large amounts of requests to be able to handle large scale measurements. A more detailed classification of the servers should be possible as well, in combination with the behaviour tracking algorithms described in Shue’s research [11]. By using query IDs to identify different queries from the same local resolver, as was the original design of this research, it should be possible to address more detailed behaviour patterns of resolvers. Patterns as for example, a resolving organisation first tries to resolve a domain name itself and secondary tries a third party resolver to support, substitute or backup its initial result.
Besides the implementation the measurements as well might need a further look. For example, the coming of IPv6 might have a big influence on the implementation of resolvers within ASs. In research on the implications of this change on the query policies and resolver implementations. Besides IPv6 a research with a better performing authoritative name server and a more realistic representation by for example more selectively chosen probes, may result in different and more complete outcomes.

7. REFERENCES


