Workgroup: add
Internet-Draft:
draft-schwartz-add-ddr-forwarders-02
Published: 8 July 2022
Intended Status: Standards Track
Expires: 9 January 2023
Authors: B. Schwartz C. Box
Google LLC BT

Reputation Verified Selection of Upstream Encrypted Resolvers

Abstract

This draft describes an extension to the Discovery of Designated Resolvers (DDR) standard, enabling use of encrypted DNS in the presence of legacy DNS forwarders.

Discussion Venues

This note is to be removed before publishing as an RFC.

Discussion of this document takes place on the mailing list (add@ietf.org), which is archived at https://mailarchive.ietf.org/arch/browse/add/.

Source for this draft and an issue tracker can be found at https://github.com/bemasc/ddr-forwarders.

Status of This Memo

This Internet-Draft is submitted in full conformance with the provisions of BCP 78 and BCP 79.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at https://datatracker.ietf.org/drafts/current/.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

This Internet-Draft will expire on 9 January 2023.

Copyright Notice

Table of Contents

1. Introduction

The Discovery of Designated Resolvers specification [DDR] describes a mechanism for clients to learn about the encrypted protocols supported by a DNS server. It also describes a client validation policy that has strong security properties.

Recent estimates suggest that a large fraction, perhaps a majority, of residential internet users in the United States and Europe rely on local DNS forwarders that are not compatible with DDR. This is because they are accessed via a private IP address, which TLS certificates cannot normally prove ownership of. Many such devices also face significant hurdles in being upgraded to support encrypted DNS, so it is likely that a large installed base of legacy DNS forwarders, providing Do53 on a private IP address, will remain for some years.

A client in such a network that wants to use the network's DNS resolver is forced to use Do53. It is therefore vulnerable to passive surveillance both on the local network, and between this network and the upstream provider, even if the upstream DNS resolver supports encrypted DNS.

Many of these attacks can be mitigated by using the method described in this document. In a nutshell the process is as follows.

1. The client begins DDR discovery, querying for _dns.resolver.arpa.

2. The legacy DNS forwarder, since it does not understand DDR, forwards this query upstream.

3. The upstream recursive resolver, which supports DDR, replies with details of how to access its encrypted DNS service.

4. The client receives this response and performs Reputation Verified Selection (see Section 3).

5. On successful completion, the client may commence using encrypted DNS towards the upstream resolver. This is known as Cross-Forwarder Upgrade.

By this process, Do53 is replaced with encrypted DNS for most queries. The client may wish to continue to send locally-relevant queries (e.g. .local) towards the legacy DNS forwarder.

1.1. Scope

This document describes the interaction between DDR and legacy DNS forwarders.

DNS forwarders and resolvers that are implemented with awareness of DDR are out of scope, as they are not affected by this discussion (although see Security Considerations, Section 7).

IPv6-only networks whose default DNS server has a Global Unicast Address are out of scope, even if this server is actually a simple forwarder. If the DNS server does not use a private IP address, it is not a "legacy DNS forwarder" under this draft's definition.

2. Conventions and Definitions

The key words "**MUST**", "**MUST NOT**", "**REQUIRED**", "**SHALL**", "**SHALL NOT**", "**SHOULD**", "**SHOULD NOT**", "**RECOMMENDED**", "**NOT RECOMMENDED**", "**MAY**", and "**OPTIONAL**" in this document are to be interpreted as described in BCP 14 [RFC2119] [RFC8174] when, and only when, they appear in all capitals, as shown here.

Private IP Address - Any IP address reserved for loopback [RFC1122], link-local [RFC3927], private [RFC1918], local [RFC4193], or Carrier-Grade NAT [RFC6598] use.

Legacy DNS Forwarder - An apparent DNS resolver, known to the client only by a private IP address, that forwards the client's queries to an upstream resolver, and has not been updated with any knowledge of DDR.

Cross-Forwarder Upgrade - Establishment and use of a direct, encrypted connection between the client and the upstream resolver.

3. Reputation Verified Selection (RVS)

Reputation Verified Selection (RVS) is a method for validating whether connection using DDR is allowed. Clients **MAY** use RVS when (a) the local DNS server is identified by a Private IP address and (b) the DDR SVCB resolution process does not produce any Encrypted DNS endpoints that have this IP address in their A or AAAA records. RVS then proceeds as follows:

1. The client connects to one of the indicated Encrypted DNS endpoints.

2. The client receives a certificate, which it verifies to a suitable root of trust.

3. For each identity (e.g. SubjectAltName) in the certificate, the client constructs a Resolver Identity:

    * For DNS over TLS and DNS over QUIC, the Resolver Identity is an IP address or hostname and the port number used for the connection.

    *For DNS over HTTPS, the Resolver Identity is a URI Template in absolute form, containing the port number used for the connection and path indicated by dohpath.

4. The client determines the reputation of each Resolver Identity derived from the certificate.

5. The maximum (i.e. most favorable) reputation is the reputation of this connection.

Successful validation then permits cross-forwarder upgrade.

> OPEN QUESTION: Would it be better to use the SVCB TargetName to select a single Resolver Identity? This would avoid the need to enumerate the certificate's names, but it would require the use of SNI (unlike standard DDR), and would not be compatible with all upstream encrypted resolvers.
>
> OPEN QUESTION: Can we simplify the resolver identity to just a domain name? This would make reputation systems easier, but it would not allow distinct reputation for different colocated resolution services, so reputation providers would have to be sure that no approved resolver has other interesting colocated services.

This process **MUST** be repeated whenever a new TLS session is established, but reputation scores for each resolver endpoint **MAY** be cached.

For DNS over HTTPS, the :authority pseudo-header **MUST** reflect the Resolver Identity with the most favorable reputation, to ensure that the HTTP requests are well-formed and are directed to the intended service. If the Resolver Identity is a wildcard, the reputation system **MUST** replace it with a valid hostname that matches the wildcard.

Assessing reputation limits the ability of a DDR forgery attack to cause harm, as it will only allow an attacker to direct clients to a resolver they consider trustworthy. Major DoH client implementations already include lists of known or trusted resolvers [CHROME-DOH] [MICROSOFT-DOH][MOZILLA-TRR].

Clients **SHOULD** start by checking the resolver endpoint with the numerically lowest SVCB SvcPriority. Clients **MAY** wait until a DNS query triggers an Encrypted DNS connection attempt before performing this verification.

If RVS encounters an error or rejects the server, the client **MUST NOT** send encrypted DNS queries to that server. If RVS rejects all

compatible ServiceMode records, the client **MUST** fall back to the unencrypted resolver (i.e. plaintext DNS on port 53).

3.1. Reputation systems

Embedding a list of known trusted resolvers in a client is only one possible model for assessing the reputation of a resolver. In future a range of online reputation services might be available to be queried, each returning an answer according to their own specific criteria. These might involve answers on other properties such as jurisdiction, or certification by a particular body. It is out of scope for this document to define these query methods, other than to note that designers should be aware of bootstrapping problems. It is the client's decision as to how to combine these answers, possibly using additional metadata (e.g. location), to make a determination of reputation.

3.2. Using resolvers of intermediate reputation

If the determined reputation is a binary "definitely trustworthy" or "definitely malicious", the client's recommended action is clear. However, intermediate trust levels are also possible (e.g. "probably safe", "newly launched"). In these cases there are some options clients can consider:

* The client can simply decline to the use the encrypted service. In this case, unless there is another option, the client will fall back to Do53.

* The client can ask the user about a specific domain names that appear in the certificate. These names might be recognizable to the user, e.g. as that of an ISP. It's also possible to present more details about why a Resolver Identity lacks some element of reputation.

* The client can use the encrypted service for a limited time, as a means of mitigating interception attacks. For example, if the client limits the DDR response TTL to 5 minutes, this ensures that any attacker can continue to redirect queries for at most 5 minutes after they have left the local network.

4. Management of local blocking functionality

Certain local DNS forwarders block access to domains associated with malware and other threats. Others block based on the category of service provided by those domains, e.g. domains hosting services that are not appropriate for a work or school environment. In the short term to ensure this service is not lost due to a cross-forwarder upgrade, the maintainers can simply add "resolver.arpa" to their actively curated list of domains to block. This pattern has

been deployed by Mozilla, with the domain "use-application-dns.net" [MOZILLA-CANARY].

In the long term, it is best for filtering DNS forwarders to implement support for encrypted DNS. The following subsections describe some ways to implement this.

4.1. Local implementation with DNR

The local forwarder can be upgraded to one that implements an encrypted DNS service discoverable through DNR. This requires a TLS certificate on the local device, proving ownership of the chosen Authentication Domain Name (ADN). Onward queries to the internet **SHOULD** also be protected with encryption.

4.2. Local implementation with DDR

If the local forwarder can be upgraded to offer an encrypted DNS service, this can then be made discoverable through classic DDR. If the device has a private IP (as presumed for RVS), a self-signed certificate is sufficient as long as the client supports the Opportunistic Discovery mode of DDR. Onward queries to the internet **SHOULD** also be protected with encryption.

4.3. Move upstream

The blocking functionality can be moved to the upstream resolver. Cross-forwarder upgrade then enables the service to continue, as long as the upstream resolver has sufficient reputation.

5. Compatibility issues that can arise from cross-forwarder upgrade

Legacy DNS forwarders sometimes provide various additional services that would be lost in the event of a cross-forwarder upgrade. For all of these, a possible general mitigation is to provide users or administrators with the ability to control whether DDR is used with legacy forwarders. For example, this control could be provided via a preference, or via a notification upon discovering a new upstream resolver. Specific mitigations are also described below.

5.1. Split-horizon namespaces

Some local network resolvers contain additional names that are not resolvable in the global DNS. A simple cross-forwarder upgrade might lose access to these local names. Clients **SHOULD** be aware of well-known suffixes (e.g. .local, .home.arpa.) that require local resolution. Dynamic discovery of local prefixes would help this issue. To address any remaining ones, the following mitigation can be used.

5.1.1. Mitigation: NXDOMAIN Fallback

In "NXDOMAIN Fallback", the client repeats a query to the unencrypted resolver if the encrypted resolver returns NXDOMAIN. This allows the resolution of local names, provided they do not collide with globally resolvable names (as required by [RFC2826]).

This is similar to the fallback behavior currently deployed in Mozilla Firefox [FIREFOX-FALLBACK].

NXDOMAIN Fallback results in slight changes to the security and privacy properties of encrypted DNS. Queries for nonexistent names no longer have protection against a local passive adversary, and local names are revealed to the upstream resolver.

NXDOMAIN Fallback is only applicable when a legacy DNS forwarder might be present, i.e. the unencrypted resolver has a private IP address, and the encrypted resolver has a different IP address. In other DDR configurations, any local names are expected to resolve similarly on both resolvers.

5.2. Interposable domains

An "interposable domain" is a domain whose owner deliberately allows resolvers to forge certain responses. This arrangement is most common for search engines, which often support a configuration where resolvers forge a CNAME record to direct all clients to a child-appropriate instance of the search engine [DUCK-CNAME][BING-CNAME] [GOOGLE-CNAME].

Future deployments of interposable domains can instruct administrators to enable or disable DDR when adding the forged record, but forged records in legacy DNS forwarders could be lost due to a cross-forwarder upgrade.

5.2.1. Mitigation: Exemption list

There are a small number of pre-existing interposable domains, largely of interest only to web browsers. Clients can maintain a list of relevant interposable domains and resolve them only via the network's resolver.

5.3. Caching

Many legacy DNS forwarders also provide a shared cache for all network users. Cross-forwarder upgrades will bypass this cache, resulting in slower DNS resolution for some queries.

5.3.1. Mitigation: Stub caches

Clients can compensate partially for any loss of shared caching by implementing local DNS caches. This mitigation is already widely deployed in browsers and operating systems.

6. Privacy Considerations

6.1. Privacy gains

The conservative validation policy results in no encryption when a legacy DNS forwarder is present. This leaves the user's query activity vulnerable to passive monitoring [RFC7258], either on the local network or between the user and the upstream resolver.

Reputation Verified Selection enables the use of encrypted transport in these configurations, reducing exposure to a passive surveillance adversary.

6.2. Privacy losses

In some legacy DNS forwarder implementations, the upstream resolver is not able to determine whether two queries were issued by the same client inside the network. It can only see aggregated queries being made by the forwarder. [DDR] to a non-local resolver requires individual encrypted DNS connections from each device, revealing which queries were made by the same client. RVS shares this property.

6.2.1. Mitigation: Open multiple connections

If the above issue is a concern, clients **MAY** open multiple connections to the designated encrypted resolver with separate local state (e.g. TLS session tickets), and distribute queries among them. This may reduce the upstream resolver's ability to link queries that came from a single client.

7. Security Considerations

When the client uses the conservative validation policy described in [DDR], the client can establish a secure DDR connection only in the absence of an active attacker. An on-path attacker can impersonate the resolver and intercept all queries, by preventing the DDR upgrade.

This basic security analysis also applies if the client uses Reputation Verified Selection. However, the detailed security properties differ, as discussed in this section.

7.1. Redirection

An on-path attacker might be located on the local network, or between the local network and the upstream resolver. In either case, the attacker can redirect the client to a resolver of the attacker's choice, *as long as that resolver meets the client's requirements for reputation*. Hence the reputation system is essential to the security of the user.

Weaknesses in the reputation system could reopen this class of vulnerabilities.

7.1.1. Possible weakness: Stale reputation

If a previously-reputable resolver is compromised, users can be redirected to it while this reputation remains high. Once an attack has been detected, it should be reported to relevant reputation services so that they can revise their assessment of this resolver.

7.1.2. Possible weakness: Inappropriate reputation

The reputation of a resolver might depend on aspects of the client's connection context, e.g. their geographic location. For example, a local ISP's resolver could be reputable for clients in its service area, but suspicious for clients on distant continent. Accordingly, very large reputation systems may need to customize their results based on the context.

7.2. Forensic logging

7.2.1. Network-layer logging

With the conservative validation policy, a random sample of IP packets is likely sufficient for manual retrospective detection of a DNS redirection attack.

With Reputation Verified Selection, local forensic logs must capture a specific packet (the attacker's DDR designation response) to enable retrospective detection of a redirection attack.

7.2.1.1. Additional Mitigation: Log all DDR responses

Redirection attacks are largely mitigated by RVS, but the loss of network-layer logging for such attacks can be mitigated by logging all DDR responses, or more generally all DNS responses. This makes retrospective attack detection straightforward, as the attacker's DDR response will indicate an unexpected server.

7.2.2. DNS-layer logging

DNS-layer forensic logging conducted by a legacy DNS forwarder would be lost in a cross-forwarder upgrade.

7.2.2.1. Solution: Plan to upgrade

Forwarders that want to observe all queries from RVS clients should plan to implement DDR or DNR. In the short term it is possible for the forwarder to disable DDR by responding negatively to _dns.resolver.arpa, but this is not recommended long-term as it prevents confidentiality protection.

8. References

8.1. Normative References

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, DOI 10.17487/RFC2119, March 1997, <https://www.rfc-editor.org/rfc/rfc2119>.

[RFC8174] Leiba, B., "Ambiguity of Uppercase vs Lowercase in RFC 2119 Key Words", BCP 14, RFC 8174, DOI 10.17487/RFC8174, May 2017, <https://www.rfc-editor.org/rfc/rfc8174>.

8.2. Informative References

[BING-CNAME] "Block adult content with SafeSearch - Map at a network level", n.d., <https://help.bing.microsoft.com/#apex/bing/en-us/10003/0>.

[CHROME-DOH] "DoH providers: criteria, process for Chrome", n.d., <https://docs.google.com/document/d/128i2YTV2C7T6Gr3I-81zlQ-_Lprnsp24qzy_20Z1Psw/edit>.

[DDR] Pauly, T., Kinnear, E., Wood, C. A., McManus, P., and T. Jensen, "Discovery of Designated Resolvers", Work in Progress, Internet-Draft, draft-ietf-add-ddr-08, 5 July 2022, <https://datatracker.ietf.org/doc/html/draft-ietf-add-ddr-08>.

[DUCK-CNAME] "Force Safe Search at a Network Level", n.d., <https://help.duckduckgo.com/duckduckgo-help-pages/features/safe-search/>.

[FIREFOX-FALLBACK] "About our rollout of DNS over HTTPS", n.d., <https://support.mozilla.org/en-US/kb/firefox-dns-over-https#w_about-our-rollout-of-dns-over-https>.

[GOOGLE-CNAME] "Keep SafeSearch turned on for your school, workplace, or home network", n.d., <https://support.google.com/websearch/answer/186669?hl=en>.

[MICROSOFT-DOH] "Determine which DoH servers are on the known server list", n.d., <https://docs.microsoft.com/en-us/windows-server/networking/dns/doh-client-support#determine-which-doh-servers-are-on-the-known-server-list>.

[MOZILLA-CANARY] "Canary domain - use-application-dns.net", n.d., <https://support.mozilla.org/en-US/kb/canary-domain-use-application-dnsnet>.

[MOZILLA-TRR] "Mozilla Policy Requirements for DNS over HTTPs Partners", n.d., <https://wiki.mozilla.org/Security/DOH-resolver-policy#Mozilla_Policy_Requirements_for_DNS_over_HTTPs_Partners>.

[RFC1122] Braden, R., Ed., "Requirements for Internet Hosts - Communication Layers", STD 3, RFC 1122, DOI 10.17487/RFC1122, October 1989, <https://www.rfc-editor.org/rfc/rfc1122>.

[RFC1918] Rekhter, Y., Moskowitz, B., Karrenberg, D., de Groot, G. J., and E. Lear, "Address Allocation for Private Internets", BCP 5, RFC 1918, DOI 10.17487/RFC1918, February 1996, <https://www.rfc-editor.org/rfc/rfc1918>.

[RFC2826] Internet Architecture Board, "IAB Technical Comment on the Unique DNS Root", RFC 2826, DOI 10.17487/RFC2826, May 2000, <https://www.rfc-editor.org/rfc/rfc2826>.

[RFC3927] Cheshire, S., Aboba, B., and E. Guttman, "Dynamic Configuration of IPv4 Link-Local Addresses", RFC 3927, DOI 10.17487/RFC3927, May 2005, <https://www.rfc-editor.org/rfc/rfc3927>.

[RFC4193] Hinden, R. and B. Haberman, "Unique Local IPv6 Unicast Addresses", RFC 4193, DOI 10.17487/RFC4193, October 2005, <https://www.rfc-editor.org/rfc/rfc4193>.

[RFC6598] Weil, J., Kuarsingh, V., Donley, C., Liljenstolpe, C., and M. Azinger, "IANA-Reserved IPv4 Prefix for Shared

Address Space", BCP 153, RFC 6598, DOI 10.17487/RFC6598, April 2012, <https://www.rfc-editor.org/rfc/rfc6598>.

[RFC7258] Farrell, S. and H. Tschofenig, "Pervasive Monitoring Is an Attack", BCP 188, RFC 7258, DOI 10.17487/RFC7258, May 2014, <https://www.rfc-editor.org/rfc/rfc7258>.

Acknowledgments

Thanks to Anthony Lieuallen and Eric Orth for early reviews of a previous draft.

Authors' Addresses

Benjamin Schwartz
Google LLC

Email: bemasc@google.com

Chris Box
BT

Email: chris.box@bt.com