Workgroup: ADD
Internet-Draft:
draft-reddy-add-enterprise-split-dns-09
Published: 2 March 2022
Intended Status: Standards Track
Expires: 3 September 2022
Authors: T. Reddy (Akamai) D. Wing (Citrix) K. Smith (Vodafone) B. Schwartz (Google)

Split-Horizon DNS Configuration

Abstract

When split-horizon DNS is deployed by a network, certain domains can be resolved authoritatively by the network-provided DNS resolver. DNS clients that don't always use this resolver might wish to do so for these domains. This specification enables networks to inform DNS clients about domains that are inside the split-horizon DNS, and describes how clients can confirm the local resolver's authority over these domains.

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 3 September 2022.

Table of Contents

1. Introduction

To resolve a DNS query, there are three essential behaviors that an implementation can apply: (1) answer from a local database, (2) query the relevant authorities and their parents, or (3) ask a server to query those authorities and return the final answer. Implementations that use these behaviors are called "authoritative nameservers", "full resolvers", and "forwarders" (or "stub resolvers"). However, an implementation can also implement a mixture of these behaviors, depending on a local policy, for each query. We term such an implementation a "hybrid resolver".

Most DNS resolvers are hybrids of some kind. For example, stub resolvers frequently support a local "hosts file" that preempts query forwarding, and most DNS forwarders and full resolvers can also serve responses from a local zone file. Other standardized hybrid resolution behaviors include Local Root [RFC8806], mDNS [RFC6762], and NXDOMAIN synthesis for .onion [RFC7686].

In many network environments, the network offers clients a DNS server (e.g. DHCP OFFER, IPv6 Router Advertisement). Although this server is formally specified as a recursive resolver (e.g. Section 5.1 of [RFC6106]), some networks provide a hybrid resolver instead. If this resolver acts as an authoritative server for some names, we say that

the network has "split-horizon DNS", because those names resolve in this way only from inside the network.

Network clients that use pure stub resolution, sending all queries to the network-provided resolver, will always receive the split-horizon results. Conversely, clients that send all queries to a different resolver or implement pure full resolution locally will never receive them. Clients with either pure resolution behavior are out of scope for this specification. Instead, this specification enables hybrid clients to access split-horizon results from a network-provided hybrid resolver, while using a different resolution method for some or all other names.

To achieve the required security properties, clients must be able to authenticate the DNS servers provided by the network, for example using the techniques proposed in [I-D.ietf-add-dnr] and [I-D.ietf-add-ddr], and prove that they are authorized to serve the offered split-horizon DNS names. As a result, use of this specification is limited to servers that support authenticated encryption and split-horizon DNS names that are properly rooted in the global DNS.

2. Terminology

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.

This document makes use of the terms defined in [RFC8499]. The terms "Private DNS", "Global DNS" and "Split DNS" are defined in [RFC8499].

'Encrypted DNS' refers to a DNS protocol that provides an encrypted channel between a DNS client and server (e.g., DoT, DoH, or DoQ).

The terms 'Authorized Split Horizon' and 'Domain Camping' are also defined.

2.1. Authorized Split Horizon

A split horizon configuration for some name is considered "authorized" if any parent of that name has given the local network permission to serve its own responses for that name. Such authorizations generally extend to the entire subtree of names below the authorization point.

2.2. Domain Camping

Domain Camping refers to operating a nameserver which claims to be authoritative for a zone, but actually isn't. For example, a domain

called example.com on the Internet and an internal DNS server also claims to be authoritative for example.com, but has no delegation from example.com on the Internet. Someone might domain camp on a popular domain name providing the ability to monitor queries and modify answers for that domain.

A common variation on domain camping is "NXDOMAIN camping", in which a nameserver claims a zone that does not exist in the global DNS. This is a form of domain camping because it seizes a portion of the parent zone without permission. The use of nonexistent TLDs for local services is a form of NXDOMAIN camping on the root zone.

Any form of domain camping likely violates the IAB's guidance regarding "the Unique DNS Root" [RFC2826].

3. Scope

The protocol in this document allows the domain owner to create a split-horizon DNS. Other entities which do not own the domain are detected by the client. Thus, DNS filtering is not enabled by this protocol.

4. Provisioning Domains dnsZones

Provisioning Domains (PvDs) are defined in [RFC7556] as sets of network configuration information that clients can use to access networks, including rules for DNS resolution and proxy configuration. The PvD Key dnsZones is defined in [RFC8801]. The PvD Key dnsZones notifies clients of names for which one of the network-provided resolvers is authoritative. Attempting to resolve these names via another resolver might fail or return results that are not correct for this network.

Each dnsZones entry indicates a claim of authority over a domain and its subdomains. For example, if the dnsZones entry is "example.test", this covers "example.test", "www.example.test", and "mail.eng.example.test", but not "otherexample.test" or "example.test.net".

[RFC8801] defines a mechanism for discovering multiple Explicit PvDs on a single network and their Additional Information by means of an HTTP-over-TLS query using a URI derived from the PvD ID. This set of additional configuration information is referred to as a Web Provisioning Domain (Web PvD). The PvD RA option defined in [RFC8801] SHOULD set the H-flag to indicate that Additional Information is available. This Additional Information JSON object SHOULD include the "dnsZones" key to define the DNS domains for which the network claims authority.

4.1. Confirming Authority over the Domains

To comply with [RFC2826], each dnsZones entry must be authorized in the global DNS hierarchy. To prevent domain camping, clients must confirm this authorization before making use of the entry.

To enable confirmation, the client must discover and validate the Authentication Domain Names (ADNs) of the network-designated resolvers using a method such as DNR [I-D.ietf-add-dnr]. The client must also perform an NS query for each dnsZones entry and confirm that at least one of the ADNs appears in each NS RRSet. This NS query must be conducted in a manner that is not vulnerable to tampering by the local network. Suitable tamperproof resolution strategies are described in Section 4.1.1 and Section 4.1.2.

Note that each dnsZones entry is authorized only for the specific resolvers whose ADNs appear in its NS RRSet. If a network offers multiple encrypted resolvers via DNR, each dnsZones entry may be authorized for a distinct subset of the network-provided resolvers.

4.1.1. Confirmation using a pre-configured public resolver

The client sends an NS query for the domain in dnsZones to a pre-configured resolver that is external to the network, over a secure transport. Clients SHOULD apply whatever acceptance rules they would otherwise apply when using this resolver (e.g. checking the AD bit, validating RRSIGs).

4.1.2. Confirmation using DNSSEC

The client resolves the NS record using any resolution method of its choice (e.g. querying one of the network-provided resolvers, performing iterative resolution locally), and performs full DNSSEC validation locally [RFC6698]. The result is processed based on its DNSSEC validation state (Section 4.3 of [RFC4035]):

* "Secure": the NS record is used for confirmation.

* "Bogus" or "Indeterminate": the record is rejected and confirmation is considered to have failed.

* "Insecure": the client SHOULD retry the confirmation process using a different method, such as the one in Section 4.1.1, to ensure compatibility with unsigned names.

5. An example of Split-Horizon DNS Configuration

Consider an organization that operates "example.com", and runs a different version of its global domain on its internal network.

Today, on the Internet it publishes two NS records, "ns1.example.com" and "ns2.example.com".

To add support for the mechanism described in this document, the network and endpoints first need to support [I-D.ietf-add-dnr] and [RFC8801]. Then, for each site, the administrator would add DNS servers named "ns1.example.com" or "ns2.example.com" (the names published on the Internet). Those names would be advertised to the endpoints as described in [I-D.ietf-add-dnr].

The endpoints compliant with this specification can then determine the network's internal nameservers are owned and managed by the same entity that has published the NS records on the Internet as shown in Figure 1:

Steps 1-2: The client joins the network, obtains an IP address, and discovers the resolvers "ns1.example.com" and "ns2.example.com" and their IP addresses using DNR [I-D.ietf-add-dnr]. Using [RFC8801], the client also discovers the PvD FQDN is "pvd.example.com".

Steps 3-7: The client establishes an encrypted DNS connection with "ns1.example.com", validates its TLS certificate, and queries it for "pvd.example.com" to retrieve the PvD JSON object. Note that [RFC8801] in Section 4.1 mandates the PvD FQDN MUST be resolved using the DNS servers indicated by the associated PvD. The PvD contains:

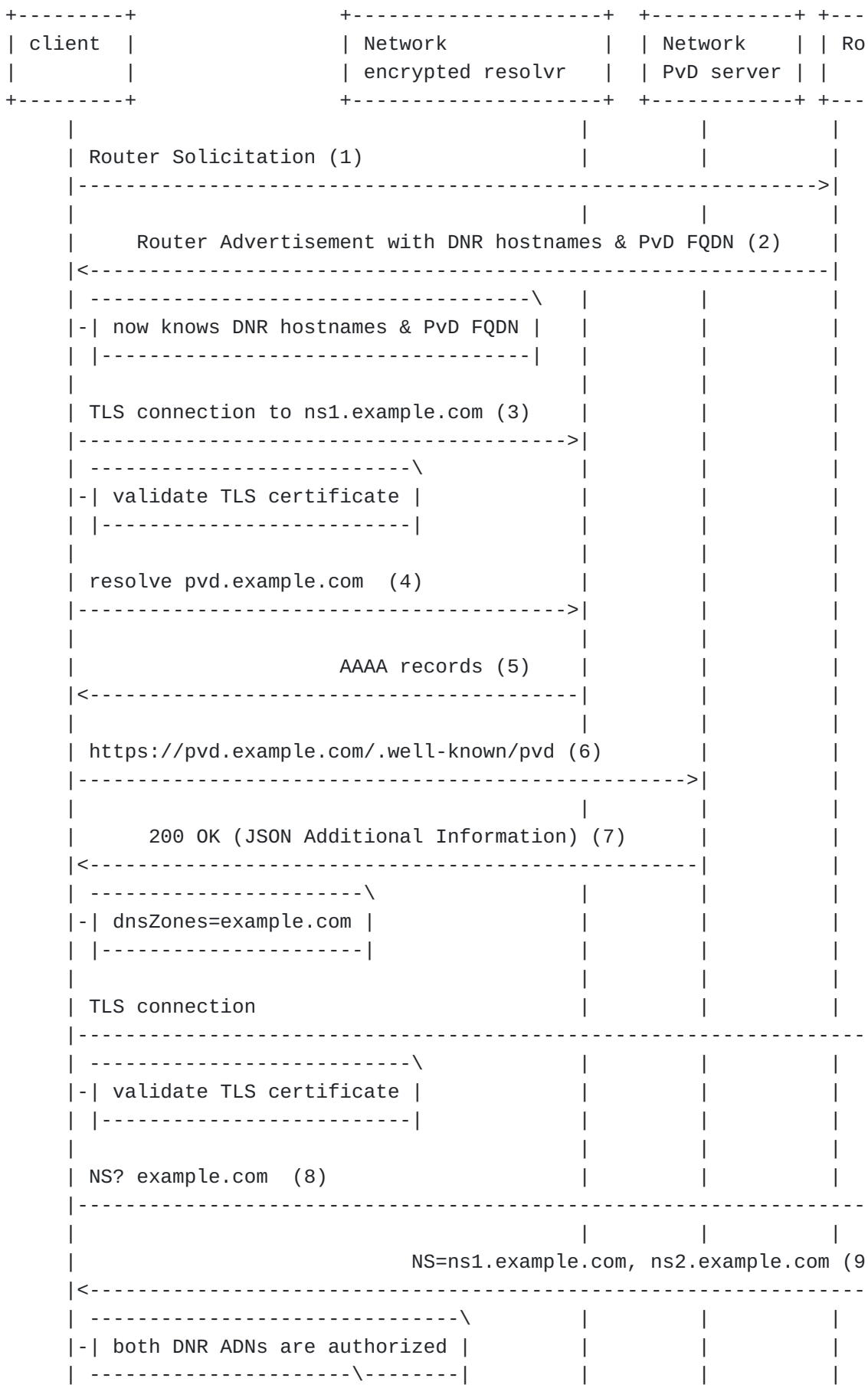
```
 {
    "identifier": "pvd.example.com",
    "expires": "2020-05-23T06:00:00Z",
    "prefixes": ["2001:db8:1::/48", "2001:db8:4::/48"],
    "dnsZones:": ["example.com"]
  }
```

The JSON keys "identifier", "expires", and "prefixes" are defined in [RFC8801].

Steps 8-9: The client then uses an encrypted DNS connection to a public resolver (e.g., 1.1.1.1) to issue NS queries for the domains in dnsZones. The NS lookup for "example.com" will return "ns1.example.com" and "ns2.example.com".

Step 10: As the network-provided nameservers are the same as the names retrieved from the public resolver and the network-designated resolver's certificate includes at least one of the names retrieved from the public resolver, the client has finished validation that the nameservers signaled in [I-D.ietf-add-dnr] and [RFC8801] are owned and managed by the same entity that published the NS records on the Internet. The endpoint will then use that

information from [I-D.ietf-add-dnr] and [RFC8801] to resolve names within dnsZones.


client
Network encrypted resolvr
Network PvD server
Ro
Router Solicitation (1)
Router Advertisement with DNR hostnames & PvD FQDN (2)
now knows DNR hostnames & PvD FQDN
TLS connection to ns1.example.com (3)
validate TLS certificate
resolve pvd.example.com (4)
AAAA records (5)
https://pvd.example.com/.well-known/pvd (6)
200 OK (JSON Additional Information) (7)
dnsZones=example.com
TLS connection
validate TLS certificate
NS? example.com (8)
NS=ns1.example.com, ns2.example.com (9
both DNR ADNs are authorized

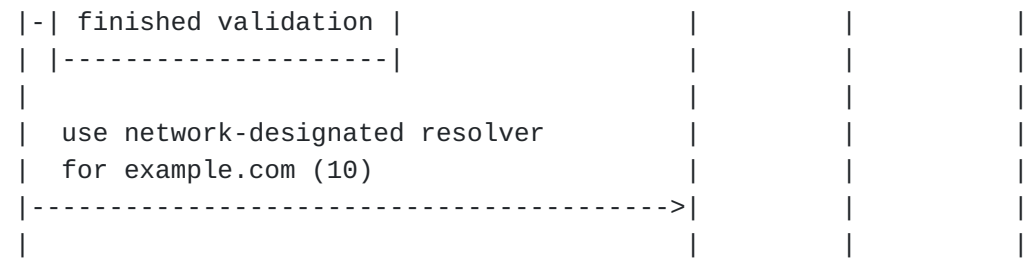


Figure 1: An Example of Split-Horizon DNS Configuration

6. Split DNS Configuration for IKEv2

The split-tunnel Virtual Private Network (VPN) configuration allows the endpoint to access resources that reside in the VPN [RFC8598] via the tunnel; other traffic not destined to the VPN does not traverse the tunnel. In contrast, a non-split-tunnel VPN configuration causes all traffic to traverse the tunnel into the VPN.

When the VPN tunnel is IPsec, the encrypted DNS resolver hosted by the VPN service provider can be securely discovered by the endpoint using the ENCDNS_IP*_* IKEv2 Configuration Payload Attribute Types defined in [I-D.ietf-ipsecme-add-ike]. For split-tunnel VPN configurations, the endpoint uses the discovered encrypted DNS server to resolve domain names for which the VPN provider claims authority. For non-split-tunnel VPN configurations, the endpoint uses the discovered encrypted DNS server to resolve both global and private domain names. For split-tunnel VPN configurations, the IKE client can use any one of the mechanisms discussed in Section 4.1 to determine if the VPN service provider is authoritative over the Split Horizon DNS domains.

Other VPN tunnel types have similar configuration capabilities, not detailed here.

7. Security Considerations

The content of dnsZones may be passed to another (DNS) program for processing. As with any network input, the content SHOULD be considered untrusted and handled accordingly. The client must perform the mechanisms discussed in Section 4.1 to determine if the network-designated encrypted resolvers are authoritative over the domains in dnsZones. If they are not, the client must ignore those dnsZones.

This specification does not alter DNSSEC validation behaviour. To ensure compatibility with validating clients, network operators MUST ensure that names under the split horizon are correctly signed or place them in an unsigned zone.

8. IANA Considerations

This document has no IANA actions.

9. Acknowledgements

Thanks to Mohamed Boucadair, Jim Reid, Tommy Pauly, Paul Vixie, Paul Wouters and Vinny Parla for the discussion and comments.

10. References

10.1. 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/info/rfc2119>.

[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/info/rfc2826>.

[RFC4035] Arends, R., Austein, R., Larson, M., Massey, D., and S. Rose, "Protocol Modifications for the DNS Security Extensions", RFC 4035, DOI 10.17487/RFC4035, March 2005, <https://www.rfc-editor.org/info/rfc4035>.

[RFC6698] Hoffman, P. and J. Schlyter, "The DNS-Based Authentication of Named Entities (DANE) Transport Layer Security (TLS) Protocol: TLSA", RFC 6698, DOI 10.17487/RFC6698, August 2012, <https://www.rfc-editor.org/info/rfc6698>.

[RFC6762] Cheshire, S. and M. Krochmal, "Multicast DNS", RFC 6762, DOI 10.17487/RFC6762, February 2013, <https://www.rfc-editor.org/info/rfc6762>.

[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/info/rfc8174>.

[RFC8801] Pfister, P., Vyncke, É., Pauly, T., Schinazi, D., and W. Shao, "Discovering Provisioning Domain Names and Data", RFC 8801, DOI 10.17487/RFC8801, July 2020, <https://www.rfc-editor.org/info/rfc8801>.

10.2. Informative References

[I-D.ietf-add-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-05, 31

January 2022, <https://www.ietf.org/archive/id/draft-ietf-add-ddr-05.txt>.

[I-D.ietf-add-dnr] Boucadair, M., Reddy, T., Wing, D., Cook, N., and T. Jensen, "DHCP and Router Advertisement Options for the Discovery of Network-designated Resolvers (DNR)", Work in Progress, Internet-Draft, draft-ietf-add-dnr-05, 13 December 2021, <https://www.ietf.org/archive/id/draft-ietf-add-dnr-05.txt>.

[I-D.ietf-ipsecme-add-ike] Boucadair, M., Reddy, T., Wing, D., and V. Smyslov, "Internet Key Exchange Protocol Version 2 (IKEv2) Configuration for Encrypted DNS", Work in Progress, Internet-Draft, draft-ietf-ipsecme-add-ike-00, 17 December 2021, <https://www.ietf.org/archive/id/draft-ietf-ipsecme-add-ike-00.txt>.

[RFC6106] Jeong, J., Park, S., Beloeil, L., and S. Madanapalli, "IPv6 Router Advertisement Options for DNS Configuration", RFC 6106, DOI 10.17487/RFC6106, November 2010, <https://www.rfc-editor.org/info/rfc6106>.

[RFC7556] Anipko, D., Ed., "Multiple Provisioning Domain Architecture", RFC 7556, DOI 10.17487/RFC7556, June 2015, <https://www.rfc-editor.org/info/rfc7556>.

[RFC7686] Appelbaum, J. and A. Muffett, "The ".onion" Special-Use Domain Name", RFC 7686, DOI 10.17487/RFC7686, October 2015, <https://www.rfc-editor.org/info/rfc7686>.

[RFC8499] Hoffman, P., Sullivan, A., and K. Fujiwara, "DNS Terminology", BCP 219, RFC 8499, DOI 10.17487/RFC8499, January 2019, <https://www.rfc-editor.org/info/rfc8499>.

[RFC8598] Pauly, T. and P. Wouters, "Split DNS Configuration for the Internet Key Exchange Protocol Version 2 (IKEv2)", RFC 8598, DOI 10.17487/RFC8598, May 2019, <https://www.rfc-editor.org/info/rfc8598>.

[RFC8806] Kumari, W. and P. Hoffman, "Running a Root Server Local to a Resolver", RFC 8806, DOI 10.17487/RFC8806, June 2020, <https://www.rfc-editor.org/info/rfc8806>.

Authors' Addresses

Tirumaleswar Reddy
Akamai
Embassy Golf Link Business Park
Bangalore 560071
Karnataka

India

Email: kondtir@gmail.com

Dan Wing
Citrix Systems, Inc.
4988 Great America Pkwy
Santa Clara, CA 95054
United States of America

Email: danwing@gmail.com

Kevin Smith
Vodafone Group
One Kingdom Street
London
United Kingdom

Email: kevin.smith@vodafone.com

Benjamin Schwartz
Google LLC

Email: bemasc@google.com