Related Domains By DNS
draft-brotman-rdbd-03

Abstract

This document describes a mechanism by which a DNS domain can publicly document the existence or absence of a relationship with a different domain, called "Related Domains By DNS", or "RDBD."

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 March 21, 2020.

Copyright Notice

Copyright (c) 2019 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust’s Legal Provisions Relating to IETF Documents (https://trustee.ietf.org/license-info) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.
1. Introduction

Determining relationships between DNS domains can be one of the more difficult investigations on the Internet. It is typical to see something such as "example.com" and "dept-example.com" and be unsure if there is an actual relationship between those two domains, or if one might be an attacker attempting to impersonate the other. In some cases, anecdotal evidence from the DNS or WHOIS/RDAP may be sufficient. However, service providers of various kinds may err on the side of caution and treat one of the domains as untrustworthy or abusive if it is not clear that the two domains are in fact related. This specification provides a way for one domain to explicitly document, or disavow, relationships with other domains, utilizing DNS records.

It is not a goal of this specification to provide a high-level of assurance as to whether or not two domains are definitely related, nor to provide fine-grained detail about the kinds of relationships
that may exist between domains. However, the mechanism defined here is extensible in a way that should allow use-cases calling for such declarations to be handled later.

1.1. Use-Cases

The use cases for this include:

- where an organisation has names below different ccTLDs, and would like to allow others to correlate their ownership more easily, consider "example.de" and "example.ie" registered by regional offices of the same company;
- following an acquisition, a domain holder might want to indicate that example.net is now related to example.com in order to make a later migration easier;
- when doing Internet surveys, we should be able to provide more accurate results if we have information as to which domains are, or are not, related;
- a domain holder may wish to declare that no relationship exists with some other domain, for example "good.example" may want to declare that it is not associated with "g00d.example" if the latter is currently being used in some cousin-domain style attack in which case, it is more likely that there can be a larger list of names (compared to the "positive" use-cases) for which there is a desire to disavow a relationship.

[[Discussion of this draft is taking place on the dnsop@ietf.org mailing list. Previously, discussion was on the dbound@ietf.org list. There’s a github repo for this draft at <https://github.com/abrotman/related-domains-by-dns> - issues and PRs are welcome there.]]

1.2. 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 [RFC2119].

The following terms are used throughout this document:

- Relating-domain: this refers to the domain that is declarating a relationship exists. (This was called the "parent/primary" in -00).
2. New Resource Record Types

We define a resource record type (RDBD) that can declare, or disavow, a relationship. RDBD also includes an optional digital signature mechanism that can somewhat improve the level of assurance with which an RDBD declaration can be handled. This mechanism is partly modelled on how DKIM [RFC6376] handles public keys and signatures - a public key is hosted at the Relating-domain (e.g., "club.example.com"), using an RDBDKEY resource record, and the RDBD record of the Related-domain (e.g., "member.example.com") can contain a signature (verifiable with the "club.example.com" public key) over the text representation (‘A-label’) of the two names (plus a couple of other inputs).

2.1. RDBDKEY Resource Record Definition

The RDBDKEY record is published at the apex of the Relating-domain zone.

The wire and presentation format of the RDBDKEY resource record is identical to the DNSKEY record. [RFC4034]

[[All going well, at some point we’ll be able to say...]] IANA has allocated RR code TBD for the RDBDKEY resource record via Expert Review.  [[In the meantime we’re experimenting using 0xffa8, which is decimal 65448, from the experimental RR code range, for the RDBDKEY resource record.]]

The RDBDKEY RR uses the same registries as DNSKEY for its fields. (This follows the precedent set for CDNSKEY in [RFC7344].)

No special processing is performed by authoritative servers or by resolvers, when serving or resolving. For all practical purposes, RDBDKEY is a regular RR type.

The flags field of RDBDKEY records MUST be zero.  [[Is that correct/ok?]]

There can be multiple occurrences of the RDBDKEY resource record in the same zone.
2.2. RDBD Resource Record Definition

To declare a relationship exists an RDBD resource record is published at the apex of the Related-domain zone.

To disavow a relationship an RDBD resource record is published at the apex of the Relating-domain zone.

[[All going well, at some point we’ll be able to say...]] IANA has allocated RR code TBD for the RDBD resource record via Expert Review.  
[[In the meantime we’re experimenting using 0xffa3, which is decimal 65443, from the experimental RR code range, for the RDBD resource record.]]

The RDBD RR is class independent.

The RDBD RR has no special Time to Live (TTL) requirements.

There can be multiple occurrences of the RDBD resource record in the same zone.

RDBD relationships are uni-directional. If bi-directional relationships exist, then both domains can publish RDBD RRs and optionally sign those.

The wire format for an RDBD RDATA consists of a two octet rdbd-tag, a domain name or URL, and the optional signature fields which are: a two-octet key-tag, a one-octet signature algorithm, and the digital signature bits.

We define two possible values for the rdbd-tag in this specification, later specifications can define new rdbd-tag values:

- 0: states that no relationship exists between the domains
- 1: states that some relationship exists between the domains
The domain name field contains either a single domain name, or an HTTPS URL. In the latter case, successfully de-referencing that URL is expected to result in a JSON object that contains a list of domain names, such as is shown in the figure below.

```
[  
  "example.com",
  "example.net",
  "foo.example"
]
```

If an optional signature is included, the sig-alg field MUST contain the signature algorithm used, with the same values used as would be used in an RRSIG. The key-tag MUST match the RDBDKEY RR value for the corresponding public key, and is calculated as defined in [RFC4034] appendix B.

If the optional signature is omitted, then the presentation form of the key-tag, sig-alg and signature fields MAY be omitted. If not omitted then the sig-alg and key-tag fields MUST be zero and the signature field MUST be an empty string. [[Is that the right way to have optional fields in presentation syntax for RRs?]]

The input to signing ("to-be-signed" data) is the concatenation of the following linefeed-separated (where linefeed has the value '0x0a') lines:

```
relating=<Relating-domain name>
related=<Related-domain name or URL>
rdbd-tag=<rdbd-tag value>
key-tag=<key-tag>
sig-alg=<sig-alg>
```

The Relating-domain and Related-domain values MUST be the 'A-label' representation of these names. The trailing "." representing the DNS root MUST NOT be included in the to-be-signed data, so a Relating-domain value above might be "example.com" but "example.com." MUST NOT be used as input to signing.

The rdbd-tag and key-tag and sig-alg fields MUST be in decimal with leading zeros omitted.

A linefeed MUST be included after the "sig-alg" value in the last line.
3. RDBD processing

- If multiple RDBD records exist with conflicting "rdbd-tag" values, those RDBD records SHOULD be ignored.

- If an RDBD record has an invalid or undocumented "rdbd-tag", that RDBD record SHOULD be ignored.

- The document being referenced by a URL within an RDBD record MUST be a well-formed JSON [RFC8259] document. If the document does not validate as a JSON document, the contents of the document SHOULD be ignored. There is no defined maximum size for these documents, but a referring site ought be considerate of the retrieving entity’s resources.

- When retrieving the document via HTTPS, the certificate presented MUST properly validate. If the certificate fails to validate, the retrieving entity SHOULD ignore the contents of the file located at that resource.

- Normal HTTP processing rules apply when de-referencing a URL found in an RDBD record, for example, a site may employ HTTP redirection.

- Consumers of RDBD RRs MAY support signature verification. They MUST be able to parse/process unsigned or signed RDBD RRs even if they cannot cryptographically verify signatures.

- Implementations producing RDBD RRs SHOULD support optional signing of those and production of RDBDKEY RRs.

- Implementations of this specification that support signing or verifying signatures MUST support use of RSA with SHA256 (sig-alg==8) with at least 2048 bit RSA keys. [RFC5702]

- RSA keys MUST use a 2048 bit or longer modulus.

- Implementations of this specification that support signing or verifying signatures SHOULD support use of Ed25519 (sig-alg==15). [RFC8080][RFC8032]
A validated signature is solely meant to be additional evidence that the relevant domains are related, or that one disavows such a relationship.

4. Use-cases for Signatures

[[The signature mechanism is pretty complex, relative to anything else here, so it might be considered as an at-risk feature.]]

We see two possibly interesting use-cases for the signature mechanism defined here. They are not mutually exclusive.

4.1. Many-to-one Use-Case

If a bi-directional relationship exists between one Relating-domain and many Related-domains and the signature scheme is not used, then making the many required changes to the Relating-domain zone could be onerous. Instead, the signature mechanism allows one to publish a stable value (the RDBDKEY) once in the Relating-domain. Each Related-domain can then also publish a stable value (the RDBD RR with a signature) where the signature provides confirmation that both domains are involved in declaring the relationship.

This scenario also makes sense if the relationship (represented by the rdbd-tag) between the domains is inherently directional, for example, if the relationship between the Related-domains and Relating-domain is akin to a membership relationship.

4.2. Extending DNSSEC

If the Relating-domain and Related-domain zones are both DNSSEC-signed, then the signature mechanism defined here adds almost no value and so is unlikely to be worth deploying in that it provides no additional cryptographic security (though the many-to-one advantage could still apply). If neither zone is DNSSEC-signed, then again, there may be little value in deploying RDBD signatures.

The minimal value that remains in either such case, is that if a client has acquired and cached RDBDKEY values in some secure manner, then the RDBD signatures do offer some benefit. However, at this point it seems fairly unlikely that RDBDKEY values will be acquired and cached via some secure out-of-band mechanisms, so we do not expect much deployment of RDBD signatures in either the full-DNSSEC or no-DNSSEC cases.

However, where the Relating-domain’s zone is DNSSEC-signed, but the Related-domain’s zone is not DNSSEC signed, then the RDBD signatures
do provide value, in essence by extending DNSSEC "sideways" to the Related-domain. The figure below illustrates this situation.

```
+-----------------+
   | Relating-domain|
   | (DNSSEC-signed) |
   +-----------------+<----+ Related-domain
   | RDBDKEY-1       |
   +-----------------+     | (NOT DNSSEC-signed)
                  +----+ RDBD RR with SIG
```

Extending DNSSEC use-case for RDBD signatures

5. Security Considerations

5.1. Efficacy of signatures

The optional signature mechanism defined here offers no protection against an active attack if both the RDBD and RDBDKEY values are accessed via an untrusted path.

5.2. DNSSEC

RDBD does not require DNSSEC. Without DNSSEC it is possible for an attacker to falsify DNS query responses for someone investigating a relationship. Conversely, an attacker could delete the response that would normally demonstrate the relationship, causing the investigating party to believe there is no link between the two domains. An attacker could also replay an old RDBD value that is actually no longer published in the DNS by the Related-domain.

Deploying signed records with DNSSEC should allow for detection of these kinds of attack.

5.3. Lookup Loops

A bad actor could create a loop of relationships, such as a.example->b.example>c.example->a.example or similar. Automated systems SHOULD protect against such loops. For example, only performing a configured number of lookups from the first domain. Publishers of RDBD records SHOULD attempt to keep links direct and so that only the fewest number of lookups are needed, but it is understood this may not always be possible.
6. IANA Considerations

This document introduces two new DNS RR types, RDBD and RDBDKEY. [[Codepoints for those are not yet allocated by IANA, nor have codepoints been requested so far.]]

[[New rdbd-tag value handling will need to be defined if we keep that field. Maybe something like: 0-255: RFC required; 256-1023: reserved; 1024-2047: Private use; 2048-65535: FCFS. It will also likely be useful to define a string representation for each registered rdbd-tag value, e.g. perhaps "UNRELATED" for rdbd-tag value 0, and "RELATED" for rdbd-tag value 1, so that tools displaying RDBD information can be consistent.]]

7. Acknowledgements

Thanks to all who commented on this on the dbound and other lists, in particular to the following who provided comments that caused us to change the draft: Bob Harold, John Levine, Pete Resnick, Andrew Sullivan, Tim Wisinski, Suzanne Woolf, Joe St. Sauver, and Paul Wouters. (We're not implying any of these fine folks actually like this draft btw, but we did change it because of their comments:-) Apologies to anyone we missed, just let us know and we'll add your name here.

8. References

8.1. Normative References


Appendix A. Implementation (and Toy Deployment:-) Status

[[Note to RFC-editor: according to RFC 7942, sections such as this one ought not be part of the final RFC. We still dislike that idea, but whatever;-)]]

We are not aware of any independent implementations so far. One of the authors has a github repo at <https://github.com/sftcd/rdbd-deebeedeerrr> with scripts that allow one to produce zone file fragments and signatures for a set of domains. There is also a wrapper script for the dig tool that provides a nicer view of RDBD and RDBDKEY records, and that verifies signatures. See the README there for details.

In terms of deployments, we used the above for a "toy" deployment in the tolerantnetworks.ie domain and other related domains that one can determine by following the relevant trail:-)

Appendix B. Examples

These examples have been generated using the proof-of-concept implementation mentioned above. These are intended for interop, not for beauty:-) The dig wrapper script referred to above produces more readable output, shown further below..

The following names and other values are used in these examples.
o Relating domain: my.example

o Related domain: my-way.example

o Unrelated domain: my-bad.example

o URL for other related domains: <https://my.example/related-names>

o URL for other unrelated domains: <https://my.example/unrelateds>

my.example zone file fragments:

```plaintext
my.example.     3600 IN TYPE65448 \# 298 
000003083082012300d06092a864886f70d010101050 
00382010f003082010a0282010100bb3b0979b3c4e61 
0f231dafbd8295d56d9475eba8df1c0f49b0b99a7b5 
15e660c243b8ce7175cc9857be00847c0f865ca81e65a 
f0ec1813a43787902e8b2560b64016c4c8e64262b7b8e 
ae2e6f735e1186237ff49110227b69fbcefa1cfddff7f 
df052f250871bb03be114493a8e29a95d04b50b9e99b5 
8e40e703813b4c159d02d781e6837791c2ead0c547e7f 
fb0aa198b2aef259c42273a69af4f22c7439972d3052d 
4a581995e203115963689044b4cbbdb6cf90ff1866630 
593aad625772e6f540bd93801c5781fd74481fbb6399 
f745b4525c767e3fb44d919e265541f6beee95d0b9e1 
15bd4749a3a9748e2d8745466629fa6682d36e83cbac8 
30203010001
```

```plaintext
my.example.     3600 IN TYPE65443 \# 85 
0001066d792d776179076578616d706c650039820f039 
b08e9d5a8e057a87c67d7b92a680b7a2e69baef46404 
b3bc9fcd93f4fe261bda56c107dbac6d72255a86a771f 
cc3e0ca0f12cdd1b302f20b2234de8610e03
```

```plaintext
my.example.     3600 IN TYPE65443 \# 18 
0000066d792d626164076578616d706c6500
```

```plaintext
my.example.     3600 IN TYPE65443 \# 39 
00012368747470733a2f2f6d792d7761792e6578616d7 
06c652f6d797374756662e6a736f6e00
```

```plaintext
my.example.     3600 IN TYPE65443 \# 42 
0002668747470733a2f2f6d792d7761792e6578616d7 
06c652f6e6f746d797374756662e6a736f6e00
```
my.example private key:

```
-----BEGIN RSA PRIVATE KEY-----
MIIePAIBAAKCAQAEzuRsJ15s8ThEPIx2vvYKVlbZRR166jHF9Jsu2p2gV5mDCQ7
jOc0XMMF6eAI8R8/42ncB5wR769gYpDeHkc6LJrCQ2BbEYOZCRe46ulM9zXhGGI3
/3RMCJ7a0V0+hz933/fBS81CHG7A74RRJ0o4pqV0ETQnebWQOCQaT6MFZ0C14
Hmg3eRwurQxQfu7/7CqGySq7ywQic6aa9P1sdDmXLTBS1KB1V4qMRWNOkE50f7
22z5D/GGZjZQqli3V3m9UC94AcV4H10S8B+7Y5n9RbSxH27P75k2Rni2dVB9r
7PQxQeUEVUDjo610j2HRUZKfpmgtNug8uuw1DAQABoIBAFisJwwkBGGjocb2
4CLijtsVorM/E0pdIdr+F2MSkdhD/BM//3drWvajJGXCmMqWkizpXYptT0iUsGljd
cGIsJzgeW9h6nEIG/XgIH/reil2u barbarAICHQCeGZh9s2szWXb+FSQdEnQacMJJFbmv
pw0d1K5FTi2h9wTdkupF988y9h4OzVkW9qIdqozKAKnxoyY20xiglaUg6NeHrs2
Sv7ow5CerKm4ZDqvctcqxu/wB1m3i5LsPGKexDzFhDQq1e7jHbKKUw+2REF8hzc
bCfa3A5Xyo?nLdgGR2DO2lizQQA+i5Cnbp35gdODV+gipLt1wdrnm81c8U1zflf47T
a0xh2YECgYE4aU/vQV2B4ux4NXX8g3womc/rJZOMWVxd8odRhBy4s0c+atGy3ztp
SoPrBqrrkcF831b596MOE11y1gPmKk7q5nI2CMtMnStnLoj27a95QVznwby6a6a
q3cIAz/10HCexLzi8edjcwTxJv1XNE9518SbkU0Ew20Y5jzShUI4OCQYaEzVt
m3PrU5/J1GqmRhDa7PyfB9ESq5mIXA7s6MPh0XLyMn2uMFMBC3iuxNayjogzI
Gg3XVC1cb4vvrVDrkxY5aTDmizVvVf0MletBiLYjCwWHASqGqI4hxxwhvENYcYVcjs
T0WSHg8uuuHa371+2hBKrRELeLHR1yho02MCegYEA4JCh5PSNnRj19h2Wtrzc
eGBus8lqVPMAqM1f8q81jZsLj0mskI3d4N6Ez0eKryJwAc2jKf2feWFe2PaecOB3
/bNMjDHsulcXTTf2qzq0HdzA8R87FcnJ3iegT1lIR0k/KymRuLGwOna1+85DB
7Xtsy3f/L2AOE8asJCNwvY6kgAyGpuc5BW5sIF5FK/LMVZrH+OuHaf801hI8tg
G1Sm/c37EHD0+aVv3iviYdgpLpow1q94AOoCxb19A12j6KdAbhsgpzyLx5sJmFYhB
1DgnsSyAcfV0yM3aN28fVrCkXJxe0mQeOaTQHyrX9sN1ctQ+dB/biVvRcrL7q
ziaNQKBoqC59EmCOVh/bjYV66oC52YAa6A4CfDhaXw40I90ck5gWr3FeV7gw
b2xq7zLOX2H2Z+6HejJMNc/eFZKVNO0kke4KGOxcDH3pYrrLsLCLNRWx8bSyOlt
e3elKariq1XcrsPSBnn7nakU7g3Z07Hb31c0GMYf9znN+qWda+3g==
-----END RSA PRIVATE KEY-----
```

my-way.example zone file fragments:
Appendix C.  Possible dig output...

Below we show the output that a modified dig tool might display for the my.example assertions above.

my-way.example private key:

```
0000000 5f24 3132 daa0 4cc4 0a77 4cb6 e834 16db
0000020 05b0 faf7 ca27 16b6 0ae7 e177 d3f9 db5f
0000040
```
$ dig RDBD my.example

;; DiG 9.11.5-P1-1ubuntu2.5-Ubuntu <<< RDBD my.example
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 4289
;; flags: qr rd ra ad; QUERY: 1, ANSWER: 5, AUTHORITY: 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
;; EDNS: version: 0, flags:; udp: 4096
;; COOKIE: e69085d4b9a18cca63ae96035d7bc0aa96580e0d6255c122 (good)
;; QUESTION SECTION:
;my.example. IN RDBD

my.example. 3600 IN RDBD RELATED may-way.example Sig: good
    KeyId: 50885 Alg: 15 Sig: UIi04agb...
my.example. 3600 IN RDBD UNRELATED my-bad.example
my.example. 3600 IN RDBD RELATED https://my-way.example/mystuff.json
my.example. 3600 IN RDBD UNRELATED https://my-way.example/notmine.json

;; Query time: 721 msec
;; SERVER: 127.0.0.1#53(127.0.0.1)
;; WHEN: Fri Sep 13 17:15:38 IST 2019
;; MSG SIZE rcvd: 600

Appendix D. Changes and Open Issues

[[RFC editor: please delete this appendix ]]

D.1. Changes from -02 to -03

- Incorporated feedback/comments from IETF-105
- Suggest list discussion move to dnsop@ietf.org
- Adopted some experimental RRCODE values
- Fixed normative vs. informative refs
- Changed the examples to use the PoC implementation.
- Restructured text a lot
D.2. Changes from -01 to -02

- Added negative assertions based on IETF104 feedback
- Added URL option based on IETF104 feedback
- Made sample generation script
- Typo fixes etc.

D.3. Changes from -00 to -01

- Changed from primary/secondary to relating/related (better suggestions are still welcome)
- Moved away from abuse of TXT RRs
- We now specify optional DNSSEC-like signatures (we’d be fine with moving back to a more DKIM-like mechanism, but wanted to see how this looked)
- Added Ed25519 option
- Re-worked and extended examples

Authors’ Addresses

Alex Brotman
Comcast, Inc

Email: alex_brotman@comcast.com

Stephen Farrell
Trinity College Dublin

Email: stephen.farrell@cs.tcd.ie