Abstract

This document outlines a mechanism by which a registered domain can publicly document a relationship with a different registered 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 http://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 September 7, 2019.

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.
Determining relationships between registered 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 because it is not clear that the two domains are in fact related. This specification provides a way for one domain to explicitly document a relationship with another, utilizing DNS records.

Possible use cases include:
where a company has websites in different languages, and would like 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 related.

It is not a goal of this specification to provide a high-level of assurance that two domains are definitely related, nor to provide fine-grained detail about the kind of relationship that may exist between domains.

Using "Related Domains By DNS", or "RDBD", it is possible to declare that two domains are related.

We include 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., "example.com") and a reference from the related-domain (e.g., "dept-example.com") contains a signature (verifiable with the "example.com" public key) over the text representation ('A-label') of the two domain names (plus a couple of other inputs).

RDBD is intended to demonstrate a relationship between registered domains, not individual hostnames. That is to say that the relationship should exist between "example.com" and "dept-example.com", not "foo.example.com" and "bar.dept-example.com" (where those latter two are hosts).

There already exists Vouch By Reference (VBR) \[RFC5518\], however this only applies to email. RDBD could be a more general purpose solution that could be applied to other use cases, as well as for SMTP transactions.

This document describes the various options, how to create records, and the method of validation, if the option to use digital signatures is chosen.
1.1. 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).

- Related-domain: This refers to the domain that is referenced by the relating-domain, such as "dept-example.com". (This was called the "secondary" in -00.)

2. New Resource Record Types

We define two new RRTYPES, an optional one for the relating-domain (RDBDKEY) to store a public key for when signatures are in use and one for use in related-domains (RDBD).

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.

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? I’ve no idea really:-)]}
2.2. RDBD Resource Record Definition

The RDBD resource record is published at the apex of the related-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.

The RDBD RR is class independent.

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

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

```
+-----------------------------+-----------------------------+-----------------------------+
|           rdbd-tag           |                               |
+-----------------------------+-----------------------------+
|                               | relating-domain name        |
+-----------------------------+-----------------------------+
|                               | key-tag                    |
|                               | sig-alg                    |
|                               | signature                  |
+-----------------------------+-----------------------------+
```

The rdbd-tag field MUST contain the value zero. Later specifications can define new rdbd-tag values.

If an optional signiture 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.

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 a an empty string. [[Is that the right way to have optional fields in RRs? Not sure.]]

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>
related=<related-domain>
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.

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

[[Presentation syntax and to-be-signed details are very liable to change.]]

See the examples in the Appendix for further details.

3. Directionality and Cardinality

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

If one domain has relationships with many others, then the relevant RDBD RRs (and RDBDKEY RRs) can be published to represent those.

4. Required Signature Algorithms

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 SHOULD 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]

5. Validation

A validated signature is solely meant to be additional evidence that the two domains are related. The existence of this relationship is not meant to state that the data from either domain should be considered as more trustworthy.

6. Security Considerations

6.1. Efficiency 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.

If the RDBDKEY value has been cached, or is otherwise known via some sufficiently secure mechanism, then the RDBD signature does confirm that the holder of the private key (presumably the relating-domain) considered that the relationship with the related-domain was real at some point in time.

6.2. 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.

If the relating-domain has DNSSEC deployed, but the related-domain does not, then the optional signature can (in a sense) extend the DNSSEC chain to cover the RDBD RR in the related-domain’s zone.

If both domains have DNSSEC deployed, and if the relating-domain public key has been cached, then the signature mechanism provides additional protection against active attacks involving a parent of one of the domains. Such attacks may in any case be less likely and detectable in many scenarios as they would be generic attacks against DNSSEC-signing (e.g. if a registry injected a bogus DS for a
relating-domain into the registry’s signed zone). If the public key from the relevant RDNDKEY RRs is read from the DNS at the same time as a related RDBD RR, then the signature mechanism provided here may provide little additional value over and above DNSSEC.

6.3. Lookup Loops

It’s conceivable that an attacker could create a loop of relationships, such as a.com->b.com->c.com->a.com or similar. This could cause a resource issue for any automated system. A system SHOULD only perform three lookups from the first domain (a.com->b.com->c.com->d.com). The related and relating-domains 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.

7. 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.]]

8. 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, Andrew Sullivan, Suzanne Woolf, 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.

9. Informative References


Appendix A. Examples

[[TODO: script up generation of all samples - it’s not unlikely we mucked up somewhere below when generating ‘em partly-manually;-)]

A.1. Sample Unsigned RDBD RR

When example.com is the relating-domain and dept-example.com is the related-domain, an unsigned RDBD RR would look like this in a zone file:

department-example.com. IN 3600 RDBD 0 example.com.

The following is equivalent to the above:

department-example.com. IN 3600 RDBD 0 example.com. 0 0 ""
A.2. Sample RSA Signature

Appendix C of [RFC6376] has some reference material on how to create a set of keys for use in this type of use case. The RSA key length is recommended to be at least 2048 bits instead of the 1024 recommended in that appendix.

Creation of keys:

$ openssl genrsa -out rsa.private 2048
$ openssl rsa -in rsa.private -out rsa.public -pubout -outform PEM

Sample Key:

rsa.private:

----BEGIN RSA PRIVATE KEY-----
MIIEowIBAAKCAQEA2LNjBAdNATZOMdd3hlem2F8a0onOce0g1KwhKzryDCfH4LZ
kXOPzAvz4ykMHW5ykOz90zGL01GM8nsL9yztBxC4o5bYnWqPl4nbvOfdf6vyLy
7Qggpp+dj6RrscyjDlitiYapHwRyuKmER1QL6MDWLl9Z5S1qskzLVPgwq780xch
U65HipKkr21uASySZyYNEf58pRea3DpBlkLy5hCDhr2+6GF2gq1jI9gMopd2P/2Xx
Hkvz13TFTx66jS5LTsb2dy3stEd7vbf/EyQfWvrs4495a80Uk0BY7V4kgKbY5SSk
GPMwOpbV7hCQjEAIURMLM9J7Eou3U1WIQ7tj01DIQABOoIBAH/eAgwrfw6/0X
Bkg4iQ096vnWpvCvW5Z40rjRq+MnsnshKPrVL+krIGU/fvt7vaIzIFTGfrf7VWhl3
+oZg/lsRFPYUItjaluqjaxEhWvHH1saYCB21AV1x9QtkjBv4F6G2qfl1MJfro32
QP36s/hiAvJddHHNsb7BKDqrz6VEVIR5j2PmW4aLjHClqsyDIUM4zRcl4exzv+rstl
z2seOhhJrnYdc+VnKeG5GKE1ndZ3tZ0y3je/0sFNJktJPnrPirPq1ve3h3u3d+PK
obZ7BM+xokz29Fxf6AgC996r9BatTa/a8q7NYmkVRls/JdOF1UuDDNd3r93Ae4n
54qqcEGcYEb/Xuct8ALG2/6kD4iImm91055ILVxdwB+1wGlJNE11B+0I+6B8W9po
vk/fVvHMEV2BoRrREB58xxa+o1CB1m1zTUQXYQnMbDzL2N+X3FrkDSGPZPQy7GzD
wFdpY3cJShou1brt4/hPWL135ZMX3yBjeGhboTUMyVdWkXtn0WeucYEA2teP
+bg91IYUJAg/CEpdWn+8ZxzhRnBdi988Grl+iarsWAWE809GyPaeip1bywewnXRb
vllskE43CgshtnRKY8dWB2AQnRESvJOKo8wr/ONSx1WTPc78xSmmxNSy0B4s4r
quMc6HTmaetCM5/10ddCY3/r1s9FTEs536RxeCEgYEA5Af6mHYwB4AT3/ERMCsa
ZAuw7Sf5f8+9vZUItvrTV1H7DRXKE7M0pib2796rKXWxXxq2G7zrnnZa/5JXl1dW
FwC4Ofsd/A7Yv1lpWx6w3Ctce1GWRAsjFCJUZJhYBKW7Egn8xBoFmFTYxxzrWOUP
bksHrId9XJXvJU8hYekXgYBhf9i24THTVnUtyTn1b+o1s1jxMxAL7Cc674uO
qxC2u6wC8leiXNzBr2bMt4k1OJCQpwtDCNzsSyMyBoWas8ngYrmeA6sAdz
xDOGx00HwPSasNh6wDdVvMPyQ1bNGFv+78qUAQ4Aw+zgoGzjDmpJSAJrunJi2yzQ
G7MNQQKBCz KurdistanECUg2x8vgVTR5b8sB7P1iHp2j8Wabw+dMIUNEHB7q4HG zinc88XA
JXnJn2VQWBL0s10yP9o9I7TWVBCZ3MqQOqsNIQam9NkIzA46ILtXPwz2q3Nw2Tk1
m7RBP999gM9mn19/azK7YsujIj/03cNJLEIWhraKkyDFvXNyEtP
----END RSA PRIVATE KEY-----
rsa.public:
-----BEGIN PUBLIC KEY-----
MIIBIjANBgkqhkiG9w0BAQEFAAOCAQ8AMIIBCgKCAQEA2LNjBAdNATZOMdd3hIem
ZF8aOnOcEo5gLKnzryDCFH4LZkXOPzAJvz4yKMHW5ykOz90zGL01GL8ns8ly
9ztBXc4obY5wnQp14nbvOdf6vyLy7Gqgp+dj6RrycSYJdLit1YapHwRuYKmERlQL
6MDWLU9ZSW1qsKzLVFgwqtT80xchU65HipelKkr2luSAy5ZyyNef58pRea3D3p8
ShCDhr2+6GF2q91j9qMopd2P/XXKhvz13TFx56GjP5LTsb2dy3tED7vbf/EyQFV
wrs4495a80UkOBY7V4YkgRbFYSSkGFmhWoPbV7hCQjEAURWLM9J7EUou3U1WIqTj
1QIDAQAB
-----END PUBLIC KEY-----

To calculate the key-tag as specified in Appendix B of [RFC4034] we
used python code from: <https://www.v13.gr/blog/?p=239>

File containing to-be-signed data:

```bash
$ cat to-be-signed-8.txt
relating=example.com
related=foo-example.com
rdbd-tag=0
key-tag=65498
sig-alg=8
$
```

```bash
$ od -x to-be-signed-8.txt
0000000  6572 616c 6974 676e 653d 6178 706d 656c
0000020  632e 6d6f 720a 6c65 7461 6465 643d 7065
0000040  2d78 656d 616c 702e 656f 630a 6d 6472
0000060  642d 6761 303d 6b0a 7965 742d 6761
0000080  363d 3435 3839 730a 6769 612d 676c 383d
00000a0  000120 00a
0000121
```

To sign that file:
The presentation form of a signed RDBD record (with a 3600 TTL) would be:

```
department-example.com. 3600 RDBD 0 example.com. 65498 8 (hfnhVSlTZMF1t2q0U+4vyCPbfSMutxuV8zEyBv7GshOcKMOW
VLFBK116wRUb7wVgG9TSunXyIuCjdQtidEjWftwV28SsXBoz
tJPMq9Hbv2aAnfmx4HxxAMHCpx9QJ2cOK/5vobd2m2e2mX14
jd9JslGuez2wVeiCkwk0x6z/tA6I5SHmDIFJb5zeKbuvbN14y
ABoNE88pxoj7EMvQd/voag2MqtsiaM3k6vkYX3g5vhM
mzH+kRXGlePeYly/a180Sn3X2bksrffqPuiQsVC03UEm9Vn
YjgzzsJnsvNXvwWJpJ9zyWmVbgdR3/vvUsz2pVvkyJsLT9Kn1
fPNpvg== )```

The base64 encoded value for the signature can be produced using:

```
$ base64 -w48 rsa.sig
hfnhVSlTZMF1t2q0U+4vyCPbfSMutxuV8zEyBv7GshOcKMOW
VLFBK116wRUb7wVgG9TSunXyIuCjdQtidEjWftwV28SsXBoz
tJPMq9Hbv2aAnfmx4HxxAMHCpx9QJ2cOK/5vobd2m2e2mX14
jd9JslGuez2wVeiCkwk0x6z/tA6I5SHmDIFJb5zeKbuvbN14y
ABoNE88pxoj7EMvQd/voag2MqtsiaM3k6vkYX3g5vhM
mzH+kRXGlePeYly/a180Sn3X2bksrffqPuiQsVC03UEm9Vn
YjgzzsJnsvNXvwWJpJ9zyWmVbgdR3/vvUsz2pVvkyJsLT9Kn1
fPNpvg==
```
To verify, with "rsa.sig" containing the above signature:

```
$ openssl dgst -sha256 -verify rsa.public \
   -signature rsa.sig to-be-signed.txt
Verified OK
```

The RDBDKEY RR for this example would be:

```
example.com. 3600 RDBDKEY 0 3 8 {
    LS0tLS1CRUdJTiBQVUJMSUMgS0VZLS0tLS0KTUlJQklqQU5C
    Z2txaGtpRzl3MEJBUUVGQUFPQ0FRQEFNSU1CQ2dLQ0FRRUEy
    TE5qQkFkTkF0gk9NZQQzagaGxlBQpaRhjhMG9uT2NFbznMVUtX
    bkt6cn1EQ2ZINEa1hPUHpBsnZ6NH1LTuhXNX1rT3o5T3pH
    TDAxR01sOG5oEx5cij1dEJYyzRvYlk1d25RcGw0bmJ2T2Rm
    NnZ5THk3R3FncCtkajZScn1juU1KZExpdG1ZYXBId1J5dUtt
    RVJsUuwKnk1EV0xVOvTv2xxc2t6TFQZ23dxdFQ4MHhjaFU2
    NUhpcEtrcjJsdVNBvNeaX1ORWY1OHBSZWEzRDNwQmtMeQo1
    aEhNeaH1iyKzZHrjX0wWKOXFMN3BkM1AvV1h4SVt3emwzVEZ0
    WDZHa1A1TFRzYjJkeTNORUQ3dmmL0V5UWZWCndyczQ0QTvh
    OE9va09CeTWNF1rZ0tiR11Tu2tHUG1o2V9QY1y3aENRakVB
    VVtXTE05SjdpFVV91M1uxV0lxVGogKMVFJREFRQUIKLS0tL1F
    TkQqUFvCFE1DIEtEFWS0tLS0tCgo= }
```

A.3. Sample Ed25519 Signature

Since OpenSSL does not yet support Ed25519 signing via its command line tool, we generate our example using the python script below. This uses the python library from Appendix A of [RFC8032].
#!/usr/bin/env python3
# CODE_BEGINS
import sys, binascii
from eddsa2 import Ed25519

# secret chosen to be 32 octets funnily enuugh:-)
secret="rdbd-example0001rdbd-example0002".encode('utf-8')
privkey,pubkey = Ed25519.keygen(secret)
msg=open('to-be-signed-15.txt','r').read().encode('utf-8')
signature = Ed25519.sign(privkey, pubkey, msg)

print("private:" + str(binascii.hexlify(privkey)))
print("public:" + str(binascii.hexlify(pubkey)))
print("sig:" + str(binascii.hexlify(signature)))
print("to-be-signed:" + str(msg))

with open("ed25519.sig", "wb") as sigf:
sigf.write(signature)
with open("ed25519.pub","wb") as pubf:
    pubf.write(pubkey)

# CODE_ENDS

The to-be-signed-15.txt file contains:

$ cat to-be-signed-15.txt
relating=example.com
related=dept-example.com
rdbd-tag=0
key-tag=35988
sig-alg=15
$

The output when the above code is run (with some spacing added) is:
$ ./ed25519-signer.py
private:
b'726462642d6578616d706c6530303031726462642d6578616d706c6530303032'
public:
b'353fc31e1168c91f0af65d6c26fd441f7df9671a23a746bb3ec86be8d35b648'
sig:
b'466a80ce6377b1e4bec563d85b8d55bd4a51a5b91c1c1e46a9c4e22a16557c38
e85ccc8ac05e6046d0066c2c52b3a17420b59af9627840ac312f5ab55e11be07'
to-be-signed:
b'relating=example.com
related=dept-example.com
rdbd-tag=0
key-tag=35988
sig-alg=15'

The presentation form for an RDBD RR would then be:

depth-example.com. 3600 RDBD 0 example.com. 35988 15 (RmqAzmN3seS+xWPYW41VvUpRpbkCHB5GqCTiKhZVfDjoXMyKwF5gRtAGbCxSs6F0IIWEa+WJ4QKwxL1q1XhG+Bw== )

The RDBDKEY for this example would be:

eexample.com. 3600 RDBDKEY 0 3 15 (NT/DHhFoyR8K91lsJv1EH7fflnGi0nRrs+yGvo0ltkg= )

Appendix B. Changes and Open Issues

[[RFC editor: please delete this appendix ]]

B.1. 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)
o Added Ed25519 option

o Re-worked and extended examples

B.2. Open Issues

Current open github issues include:

o #5: specify input for signing more precisely - e.g. is there a CR or NULL or not

o #6: what, if anything, does rdbd for example.com mean for foo.example.com?

These can be seen at: <https://github.com/abrotman/related-domains-by-dns/issues>

Authors’ Addresses

Alex Brotman
Comcast, Inc

Email: alex_brotman@comcast.com

Stephen Farrell
Trinity College Dublin

Email: stephen.farrell@cs.tcd.ie