

Domain Boundaries  
Internet-Draft  
Intended status: Informational  
Expires: 11 March 2026

J. Yao  
CNNIC  
H. Luo  
Beihang University  
September 2025

A Method for Identifying the Administrative Boundaries of DNS Names  
draft-yao-dbound-identify-dns-boundary-01

## Abstract

Two or more DNS names belonging to the same registrants may share the same DNS administrative boundaries. Currently, there are no good methods to identify such shared administrative boundaries in DNS. This document adds the function of lookup of domain name administrative boundary to domain name system, which describes a new method for using dbound resource record for determining domain name administrative boundaries.

## 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 5 March 2026.

## Copyright Notice

Copyright (c) 2025 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 Revised BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Revised BSD License.

This document may contain material from IETF Documents or IETF Contributions published or made publicly available before November 10, 2008. The person(s) controlling the copyright in some of this material may not have granted the IETF Trust the right to allow modifications of such material outside the IETF Standards Process. Without obtaining an adequate license from the person(s) controlling the copyright in such materials, this document may not be modified outside the IETF Standards Process, and derivative works of it may not be created outside the IETF Standards Process, except to format it for publication as an RFC or to translate it into languages other than English.

## Table of Contents

1. Introduction . . . . .	2
2. Terminology . . . . .	3
3. Framework . . . . .	3
4. Application Algorithm for Dbound Query . . . . .	5
5. Wildcard issue . . . . .	8
6. IANA Considerations . . . . .	8
7. Security Considerations . . . . .	9
8. Acknowledgements . . . . .	9
9. Change History . . . . .	9
9.1. draft-yao-dbound-dns-solution . . . . .	9
9.2. draft-yao-dbound-identify-DNS-boundary: Version 00 . . . . .	9
10. References . . . . .	9
10.1. Normative References . . . . .	9
10.2. Informative References . . . . .	10
Authors' Addresses . . . . .	10

## 1. Introduction

Two or more DNS [RFC1034] [RFC1035] names may have the same administrative boundaries. If they share the same DNS administrative boundaries, we regard that they have a relationship. Otherwise they have not a relationship. This document describes a method for using dbound resource record for judging domain name administrative boundaries. The Domain Name System (DNS) currently lacks a standardized method to identify administrative boundaries between

domain names owned by the same entity. While multiple domains may belong to the same registrant, DNS does not natively express these relationships.

The drafts [Boundaries-Problem] [Boundaries-Concepts] list many use cases where some applications may use domain name administrative boundaries. With the growth of Internet, there should have more Internet applications which will use domain name administrative boundaries technology.

As the ICANN new gTLD program has expanded, registrants frequently obtain multiple domain names serving a common purpose. Consequently, there is a need to develop a mechanism for determining when two or more domain names share administrative boundaries. This requirement is equally critical for second-level domains (SLDs) within the same top-level domain (TLD). Numerous organizations manage multiple SLDs (e.g., variant1.tld, variant2.tld, variant3.tld) that operate under unified administrative control, yet lack a DNS-native means to express this relationship.

## 2. Terminology

The basic DNS terms used in this specification are defined in the documents [RFC1034] and [RFC1035]. The shared administrative boundaries of DNS names imply, at minimum, that such names are under the control of either the same registrant or cooperating registrants.

## 3. Framework

This section presents a mechanism to lookup of the administrative boundary between two domains. The mechanism defines a new resource record type (RRTYPE) called as Dbound to satisfy the requirements specified in the previous section. The RDATA for an Dbound RR consists of a 1 octet Flag field, a 1 octet Reserved 1 field, a 1 octet Reserved 2 field, and a Anchor Name / Name Collection field.

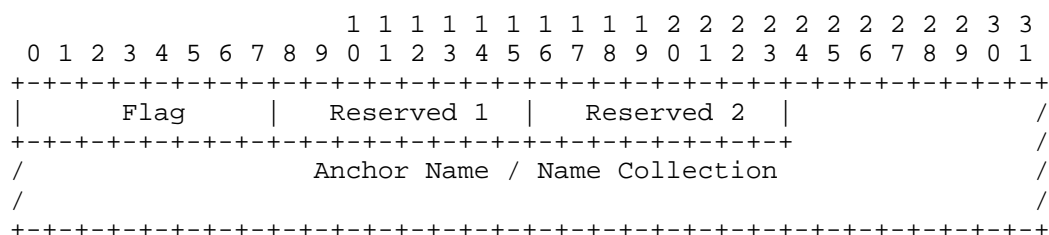


Figure 1: The structure of RDATA of Dbound resource record

Flag: The Flag field identifies the usage of Anchor Name / Name Collection field.

If flag=0, the Anchor Name / Name Collection is the anchor name, and the anchor name will be the string of PSL. Through it, the DNS administrators can configure the relationship between the owner name and PSL. Those which point to the PSL will share the same DNS administrative boundaries;

If flag=1, the Anchor Name / Name Collection is the anchor name, and it means that dbound record is to try to build a connection between the owner name and the anchor name which is a FQDN. Through it, the DNS administrators can configure the relationship between the owner name and the anchor name. Those which share the same anchor name will share the same DNS administrative boundaries;

If flag=2, the Anchor Name / Name Collection is the name collection, and the Name Collection will be a collection of names which are supposed to share the same DNS boundaries under the same anchor name and will be separated by comma(,). The owner name is some names' anchor name in other dbound RR. Through it, the application can learn how many names share the same DNS boundaries under the owner name (some names' anchor name in other dbound RRs)

Reserved 1 field and Reserved 2 field: These two fields will be kept for the future use.

Anchor Name / Name Collection: There are two kinds of relationship mechanism, one is controlled by PSL; the other is specified by building the connection among names.

If Flag is 0, the Anchor Name / Name Collection field will have the value of PSL;

If Flag is 1, the Anchor Name / Name Collection field will have the value of the anchor name. The anchor name acts like a middleman. All names sharing the same administrative boundaries will point to the same anchor name;

If Flag is 2, the Anchor Name / Name Collection field will have the value of name collection with names separated by comma (,).

#### 4. Application Algorithm for Dbound Query

Given two domain names A and B There are two cases where application can determin whether domain names A and B share the same administrative boundaries.

Case 1: If A and B's flag value in the dbound record are 0, application should confirm that the Anchor Name / Name Collection fields of both names have the value of PSL.

Case 2: If A and B's flag value in the dbound record are 1, application should check whether the names point to the same anchor.

## Algorithm :

1)When the application needs to know whether two names A and B share the same administrative boundary, it needs to do the following steps to confirm it.

Step 1, the application sends the query of A for dbound record to the DNS servers, and analyzes the response. If the application gets the dbound RR for A, it checks whether there is a dbound record with the flag value of 0 or 1. If the value of flag of A's dbound records is 0, go to step 2; If the value of flag of A's dbound records is 1, go to step 3; Otherwise, go to step 4;

Step 2, the application sends the query of B for dbound record to the DNS servers, and analyzes the response. If the application gets the dbound RR, it checks this RR. If the value of flag of B's dbound records is 0, check whether the value of anchor name of A and B's dbound records are PSL. If yes, it means that A and B tend to enjoy the same administrative boundaries under the PSL. Then check the PSL to confirm whether A and B enjoy the same administrative boundaries. If yes, return the response and exit. Otherwise go to step 4

Step 3, the application sends the query of B for dbound record to the DNS servers, and analyzes the response. If the application gets the dbound RR, it checks this RR. If the value of flag of B's dbound records is 1, check whether the value of anchor name of A and B's dbound records are same. If yes, it means that A and B tend to enjoy the same administrative boundaries under the same anchor name. Then check the anchor name to confirm whether A and B enjoy the same administrative boundaries. That means If yes, the application should send the query of the anchor name for dbound record to the DNS servers, and analyzes the response. If the application gets the dbound RR, it checks this RR. If the value of flag of anchor's dbound records is 2, check whether the dbound record's value include A and B in its name collection. If yes, then return the response and exit. Otherwise go to step 4

step 4, Exit and display some error information

2)Given name A, the application queries the DNS servers for the dbound record of name A and processes the response. If a dbound RR for A is retrieved, and if the flag value of A's dbound record is 1, the application checks for the presence of a dbound record with flag value 2 associated with A's anchor name. Upon such a record being found, the application extracts the name collection from this dbound record to obtain the list of names. For each name in this list, the application uses the procedure specified in Algorithm 1) (for the case of two domain names A and B) to verify whether it shares the same administrative boundary as A.

3) Given more than two names, check whether they share the same administrative boundaries. The application selects one of the names as A and check whether every other name shares the same administrative boundaries with A via the the method 1) which describes the case of two domain names A and B.

For examples:

EXAMPLE 1,  
Configuration: if a.example and b.exmaple want to share the same DNS administrative boundaries, it can configure the following RRs:

```
a.example dbound 1 c.example
b.example dbound 1 c.example
c.example dbound 2 a.example,b.example
```

or the anchor name can also be one of the names who share the same DNS administrative boundaries:

```
a.example dbound 1 b.exmaple
b.example dbound 1 b.example
b.example dbound 2 a.example,b.example
```

USAGE: if the application wants to check whether a.example and b.example share the same DNS boundaries, it finds a.example and b.example share the same anchor under the flag's value of 1 under the RRs above, and verify that a.example and b.example share the same DNS boundaries.

if the application wants to check which domain names share the same DNS boundaries with a.example, it finds a.example and b.example are supposed to have the same DNS boundaries under the flag's value of 2, and verify that a.example and b.example share the same DNS boundaries through checking a.example and b.example sharing the same anchor under the flag's value of 1 .

EXAMPLE 2,  
Configuration: if a.example and b.exmaple want to share the same DNS administrative boundaries under PSL, it can configure the following RRs:

```
a.example dbound 0 http://mxr.mozilla.org/mozilla-
central/source/netwerk/dns/ effective_tld_names.dat?raw=1
b.example dbound 0 http://mxr.mozilla.org/mozilla-
central/source/netwerk/dns/ effective_tld_names.dat?raw=1
```

USAGE: if the application wants to check whether a.example and b.example share the same dns boundaries, it finds a.example and b.example share the same anchor under the flag's value of 0, and verify that a.example and b.example share the same dns boundaries via the PSL link.

This new mechanism builds a relationship through the anchor name (middleman) to avoid to construct too many pairwise relationship. It will help to reduce the RRs configuration and checking when there are many domain names which are supposed to share the administrative boundaries.

## 5. Wildcard issue

The parent name may announce that all names under it to share the same administrative boundaries with itself, but it needs two-way assertion here. Parents can say all its children are under its control and share the same boundaries. In the other hand, the children should confirm that they share the same boundary with its parents too.

For example:

```
example.com dbound 1 example.com
```

```
*.example.com dbound 1 example.com
```

```
example.com dbound 2 example.com, *.example.com
```

It means that example.com and its children share the same administrative boundaries.

In some cases, the children may lose its parent's control by configure some DNS records for themselves. The dbound record has similar same limitation with the wildcard. Wildcards work for the non-configured sub-domain names only. Those names which can not queried through wildcard will not work for dbound too. Those names should configure their own dbound record separately instead of wildcard dbound configuration. For example:

If there is an A record at a.b.example.com, the wildcard will not match a.b.example.com or b.example.com. In this example, querying c.example.com will work if c.example.com is not configured in some ways. If there is a A record for a.b.example.com, it indicates that the a.b.example.com or b.example.com might exist. so under this situation, a.b.example.com or b.example.com should configure their own dbound record since a.b.example.com or b.example.com may be out of control of the parents.

## 6. IANA Considerations

The IANA should allocate the new DNS type for DBOUND.



The template is as follows:

RRTYPE: DBOUND

Number: TBD

Contact: IETF Chair

Description: Identifying the Administrative Boundaries of DNS Names.

Reference: This document

## 7. Security Considerations

The introduction of the DBOUND (Domain Boundary) Resource Record (RR) to identify administrative boundaries in the DNS raises several security considerations that should be addressed. Registrars and DNS operators should ensure that DBOUND records are correctly provisioned to avoid misconfigurations that could break domain validation logic. Since DBOUND records define administrative ownership boundaries, they must be protected by DNSSEC to ensure data origin authentication and integrity. Without DNSSEC, an attacker could forge or modify DBOUND records, leading to incorrect assumptions about domain ownership.

## 8. Acknowledgements

Thanks a lot for WG discussion in dbound WG. Especially thanks for Andrew Sullivan and John R Levine's kind comments and helpful debate.

## 9. Change History

RFC Editor: Please remove this section.

### 9.1. draft-yao-dbound-dns-solution

1. A Resource Record for DNS Administrative Boundaries
2. This draft was created during the old DBOUND WG

### 9.2. draft-yao-dbound-identify-DNS-boundary: Version 00

- \* This draft was originally discussed in IETF dbound working group. It is one potential solution for DBOUND problem. The dbound WG closed without publishing any RFC. The WG suggested the drafts in this WG to pursue the publication through independent stream. Since the ICANN's new gTLD program is coming in 2026 and many proposed IDN TLDs are name variants, the topic about how to identify DNS Administrative Boundaries becomes very important. So the author decides to update the draft.

## 10. References

### 10.1. Normative References

- [RFC1034] Mockapetris, P., "Domain names - concepts and facilities", STD 13, RFC 1034, DOI 10.17487/RFC1034, November 1987, <<https://www.rfc-editor.org/info/rfc1034>>.
- [RFC1035] Mockapetris, P., "Domain names - implementation and specification", STD 13, RFC 1035, DOI 10.17487/RFC1035, November 1987, <<https://www.rfc-editor.org/info/rfc1035>>.

## 10.2. Informative References

### [Boundaries-Concepts]

Deccio, C. and J. Levine, "Concepts for Domain Name Relationships", draft: dbound concepts, July 2015, <<https://tools.ietf.org/html/draft-deccio-dbound-name-relationships-00>>. <https://tools.ietf.org/html/draft-deccio-dbound-name-relationships-00>

### [Boundaries-Problem]

Sullivan, A., Hodges, J., and J. Levine, "DBOUND: DNS Administrative Boundaries Problem Statement", draft: dbound problem, July 2015, <<https://tools.ietf.org/html/draft-sullivan-dbound-problem-statement-01>>. <https://tools.ietf.org/html/draft-sullivan-dbound-problem-statement-01>

### [publicsuffix.org]

Mozilla Foundation, "Public Suffix List", also known as: Effective TLD (eTLD) List, <[http://mxr.mozilla.org/mozilla-central/source/netwerk/dns/effective\\_tld\\_names.dat?raw=1](http://mxr.mozilla.org/mozilla-central/source/netwerk/dns/effective_tld_names.dat?raw=1)>. <https://publicsuffix.org/>

## Authors' Addresses

Jiankang Yao  
CNNIC  
Building 4, No.9 Beijing Auto Museum West Road, Fengtai District  
Beijing  
Beijing, 100070  
China  
Phone: +86 10 59116505  
Email: [yaojk@cnnic.cn](mailto:yaojk@cnnic.cn)

Hongbin Luo  
Beihang University  
Beijing, 100190  
China  
Email: [luohb@buaa.edu.cn](mailto:luohb@buaa.edu.cn)