

A DNS RR for Pointers to RRs outside class IN
<[draft-hardie-out-rr-00.txt](#)>

Status of this Memo

This document is an Internet-Draft and is in full conformance with all provisions of [Section 10 of RFC2026](#).

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF), its areas, and its working groups. Note that other groups may also distribute working documents as Internet-Drafts.

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."

The list of current Internet-Drafts can be accessed at <http://www.ietf.org/ietf/lid-abstracts.txt>.

The list of Internet-Draft Shadow Directories can be accessed at <http://www.ietf.org/shadow.html>.

This Internet-Draft will expire in December 2002.

Copyright Notice

Copyright (C) The Internet Society (2002). All Rights Reserved.

Abstract

The Domain Name System is a global distributed lookup system with delegation. In the original specification of the DNS [[RFC 1035](#)], CLASSES were described as parallel data structures within a single namespace but with potentially different delegations of authority. [BCP 42] defines a different vision, in which different CLASSES represent fundamentally different namespaces. Though [BCP 42] includes procedures for assignment of CLASSES, there has been little use of this axis of extensibility; in practice, CLASS IN is the only widely deployed CLASS in the DNS. The ubiquity of CLASS IN for name to IP address mapping has caused a vicious cycle in which extensions are placed within that CLASS to take advantage of its global deployment, with each addition further increasing its gravitational attraction. This document describes a Resource

Record for use within CLASS IN that contains a pointer to a CLASS outside of IN. This mechanism is intended to allow administrators to indicate that a named resource identified within CLASS IN is also present in a different namespace, potentially under a different name. This cross-class pointer will allow the DNS to handle new namespaces with mechanisms appropriate to those namespaces while providing a connection to the globally deployed CLASS IN namespace.

Table of Contents

1. Terminology
2. Introduction
3. The OUT RR
 - 3.1 OUT Presentation format
 - 3.2 OUT RDATA Format
 - 3.3 Examples
 - 3.3.1 Example 1
 - 3.3.2 Example 2
4. Use of the OUT RR
5. Alternatives
6. Security Considerations
7. IANA Considerations
8. IAB Considerations
9. References
10. Acknowledgments
11. Author's Address
12. Full Copyright Statement

[1. Terminology](#)

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in [[RFC 2119](#)].

[2. Introduction](#)

The Domain Name System is a distributed lookup system with delegation. Though the DNS was initially described as related to a single namespace[RFC 1035], its inclusion of CLASSes which may derive from different roots creates a de facto mechanism for using the DNS with multiple namespaces. The DNS uses common resource record and packet formats for all CLASSes, but the namespaces thus created are fully independent. This independence means that a name

within one class has no necessary relationship to the same name in a different CLASS.

This independence is architecturally sound in principle, but the deployment history of the DNS shows that resource records have been highly concentrated in the CLASS IN namespace. Because developers of new RRs wish to take immediate advantage of the deployed DNS infrastructure, they use CLASS IN even in situations where the use of a new namespace would produce a better long term result. This document proposes a mechanism by which a Resource Record within CLASS IN can be used as a pointer to a NAME in a CLASS outside IN. It presumes that a cross-class pointer mechanism will allow the development of namespaces outside CLASS IN without requiring CLASS IN to be replaced or its resource records redeveloped in the new CLASSES.

[3.](#) The OUT RR

The OUT RR is a CLASS IN RR used to store a pointer to related data stored in a non-CLASS IN DNS namespace. The type number for the OUT RR is to be assigned by IANA. The OUT RR occurs in CLASS IN only. The format for the OUT RR is:

NAME TTL IN OUT IN-EXTERNAL-CLASS IN-EXTERNAL-NAME

Each OUT RR stores a single pointer, as a pair composed of an IN-EXTERNAL-CLASS and an IN-EXTERNAL-NAME. Even if a NAME in CLASS IN maps to multiple IN-EXTERNAL-NAMEs in the same IN-EXTERNAL-CLASS, each pair should be stored in a separate OUT RR.

[3.1](#) OUT Presentation format

The format for the OUT RR is:

NAME TTL IN OUT IN-EXTERNAL-CLASS IN-EXTERNAL-NAME

IN-EXTERNAL-CLASS is represented in decimal format when a mnemonic for the class has not been established.

[3.2](#) OUT RDATA Format

The RDATA section of the OUT RR in transmission contains RDLENGTH bytes of binary data. The OUT RDATA is formed by concatenating the CLASS bytes for the IN-EXTERNAL-CLASS and the octets representing the IN-EXTERNAL-NAME in the IN-EXTERNAL-CLASS. CLASS assignments are maintained by IANA according to the assignment criteria set out in [BCP42].

[3.3](#) Examples

[3.3.1](#) Example 1

A lookup is made against a DNS record in class IN for a device which runs the IP version of BACNet (Building Automation and Control Network; see Annex J of ANSI/ASHRAE Standard 135-2001). The property manager for the property at which the device is present maintains information on all its BACNet devices in its own namespace within the private use CLASS space set out in [BCP42]. In addition to reporting the data available within CLASS IN, the DNS server reports the IN-EXTERNAL-CLASS and NAME at which more information is available.

```
firealarm.example.com 3600 IN A          10.0.0.20
firealarm.example.com 3600 IN OUT 65410 SA_TEMP.AHU4.BLDG20.PAO
```

A lookup in the private class 65410 might return some set of BACNet properties associated with that device:

```
SA_TEMP.AHU4.BLD20.PAO 3600 65410 OBJ_ID      2492492
SA_TEMP.AHU4.BLD20.PAO 3600 65410 OBJ_NAME   SA_TEMP
SA_TEMP.AHU4.BLD20.PAO 3600 65410 OBJ_TYPE   Analog.Input
SA_TEMP.AHU4.BLD20.PAO 3600 65410 Vendor_ID Example_Technologies
```

Note that the property manager may choose to keep all of its BACNet devices within this namespace, whether or not the devices are IP-addressable. Note also that this example relates to a private namespace maintained by a property manager and should not be taken to imply that the above CLASS or RRs have been specified for use with BACNet.

[3.3.2](#) Example 2

A lookup is made against a DNS record in class IN for a server which is part of a Content Delivery Network. The Content Delivery Network maintains its own DNS namespace within the private use CLASS space set out in [BCP42]. In addition to reporting the data available within CLASS IN, it reports the NAME at which further information is available within its own namespace.

```
newsservice.example.com. 600 IN A      10.0.0.1
newsservice.example.com. 600 IN OUT 65500 surrogate1.cluster4.west.nam.
```

A DNS client configured to use private use CLASS 65500 can then present the NAME surrogate1.cluster4.west.nam to the appropriate DNS servers and retrieve resources records from that private address space. Within a CDN context, this might provide

information on the state of that surrogate, which might include information from resource record types that do not exist within CLASS IN. For example:

```
Surrogate1.cluster4.west.nam 600 65300 USERS newsservice.example.com
Surrogate1.cluster4.west.nam 600 65300 USERS homepage.example.com
Surrogate1.cluster4.west.nam 600 65300 USERS info-gov.example.org
Surrogate1.cluster4.west.nam 600 65300 USERS sports-scores.example.net
```

might report the set of users for a specific surrogate to those who have configured their DNS for lookups into that private use CLASS.

[3.3.3](#) Example 3

A lookup is made against a DNS record in class IN for a host which subscribes to a service that reports other proximate participating hosts. By storing a pointer to its NAME in that service's private use CLASS, a host advertises to those looking it up in CLASS IN that it participates and what to use to lookup related records. A query in CLASS IN thus might return:

```
hostname.example.com.      600 IN A      10.0.0.100
hostname.example.com.      600 IN OUT   65400 u103.rwc.ca.us.loc.
```

A user of the proximity service can then present the NAME u103.rwc.ca.us.loc. to the DNS servers for CLASS 65400 and receive appropriate information. For a proximity service a lookup might yield something like:

```
u103.rwc.ca.us.loc.  100 65400 1000m u202.mp.ca.us.loc.
u103.rwc.ca.us.loc.  100 65400 1000m u002.pa.ca.us.loc.
u103.rwc.ca.us.loc.  100 65400 1000m u333.rwc.ca.us.loc.
```

reporting the 4 hosts which were within 1000 meters. This data again contains resource records (and might well correspond to query types) that do not exist with CLASS IN.

[4.](#) Use of the OUT RR

This RR MUST NOT be used for pointers within CLASS IN. Use of this resource record as a pointer to a NAME within CLASS IN could easily create conflicts with the information published for the original NAME. If, for example, a lookup resulted in a CNAME record and an OUT record pointing to CLASS IN, conflicts over which resource record's data to use to derive an IP address might occur. In certain circumstance, it may also create a resolution loop that current use of the DNS system is not designed to detect. OUT Resource Records pointing to CLASS IN MUST be treated as an error by applications which receive them, and it would be useful if

server implementations reported attempts to use such pointers to their administrators.

5. Alternatives

The OUT resource record presumes a benefit to mapping a name in one namespace to a name in another namespace. In some cases, it can be difficult to see when that mapping between namespaces is required and when an additional resource record type in the existing namespace is sufficient. In general, two or more namespaces are required when there are distinguishable collections of identifiers, rather than when there are additional attributes to existing identifiers. In gross terms, the OUT RR allows the DNS to indicate that something identified within one set (CLASS IN) is also a member of another set, possibly under a different identifier; using additional RRs within CLASS IN, even private RRs within CLASS IN, does not imply the existence of another set.

6. Security Considerations

The storage of data within the DNS makes available information to an Internet-scale audience. Revealing information about the namespaces in which data is stored provides a hint to potential attackers about the information and should be released only after due consideration of the benefits and risks associated with that release.

7. IANA Considerations

IANA is requested to allocate an RR type number within CLASS IN for the OUT resource record type.

8. IAB Considerations

The use of pointers within CLASS IN to namespaces outside of CLASS IN risks creating a situation analogous to a split root for the DNS; indeed, it is a basic principal that the different DNS CLASSES do not need to share a single root. This document does not propose to segment reachability information critical to the operation of the Internet into different namespaces, but it does posit that some information belongs in namespaces outside CLASS IN. That risks the application namespace of the Internet being fragmented, even if the Internet's reachability information is not. This fragmentation may well be appropriate in certain instances and inappropriate in others, and due care must be given to the creation and use of namespaces outside CLASS IN.

9. References

TBA

10. Acknowledgments

Paul Mockapetris, David Conrad, and Andreas Gustafsson were all kind enough to provide feedback on early drafts of this document; that kindness should not be construed, however, as approval of the ideas contained in the document.

11. Author's Address

Ted Hardie
Nominum, Inc.
2385 Bay Road
Redwood City, CA 94063
USA

EMail: Ted.Hardie@nominum.com

12. Full Copyright Statement

Copyright (C) The Internet Society (2002). All Rights Reserved.

This document and translations of it may be copied and furnished to others, and derivative works that comment on or otherwise explain it or assist in its implementation may be prepared, copied, published and distributed, in whole or in part, without restriction of any kind, provided that the above copyright notice and this paragraph are included on all such copies and derivative works. However, this document itself may not be modified in any way, such as by removing the copyright notice or references to the Internet Society or other Internet organizations, except as needed for the purpose of developing Internet standards in which case the procedures for copyrights defined in the Internet Standards process must be followed, or as required to translate it into languages other than English.

The limited permissions granted above are perpetual and will not be revoked by the Internet Society or its successors or assigns.

This document and the information contained herein is provided on an "AS IS" basis and THE INTERNET SOCIETY AND THE INTERNET ENGINEERING TASK FORCE DISCLAIMS ALL WARRANTIES, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO ANY WARRANTY THAT THE USE OF THE INFORMATION HEREIN WILL NOT INFRINGE ANY RIGHTS OR ANY IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE.

Acknowledgment

Funding for the RFC editor function is currently provided by the Internet Society.