

## Internationalized PTR Resource Record (IPTR)

### 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/1id-abstracts.txt>

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

### Abstract

This draft attempts to address the problem of how an IP address SHOULD be properly mapped to a set of Internationalized Domain Names(IDNs). It is currently unspecified how a PTR record can be used for this purpose. In addition, the syntax of the PTR resource record may be too restrictive for such a mapping in a more culturally meaningful context. This document suggests a new TYPE called IPTR using EDNS0 and a mechanism to combined language information with such a mapping.

### **1. Introduction**

Reverse mapping is a very important and essential function in the DNS. In today's Domain Name System, PTR RRs are used to support address-to-domain mappings. However, a current PTR RR does not provide support for proper address-to-IDN mappings, without certain modifications. Modifying the PTR structure will also affect the current reverse

mapping architecture. This document describes a new RR TYPE named IPTR to provide address-to-IDN mappings and it also specifies that on receiving of a IPTR query a name server should respond with all the corresponding IPTR RRs in one response. In short, "one IP several IDNs".

### **1.1 Terminology**

The key words "MUST", "SHALL", "REQUIRED", "SHOULD", "RECOMMENDED", and "MAY" in this document are to be interpreted as described in [RFC 2119](#) [[RFC2119](#)].

### **1.2 Background and Designs**

When Internationalized Domain Names come into wide use, an Internet host is likely to have domain names in different languages. In today's Internet, even though the [[RFC2181](#)] redefine the consideration of PTR, because of the design of the PTR mapping algorithm and implementation of most resolvers, IP address to domain names mapping is still limited to "one IP one domain name".

For example, BIND treats PTRs specially so that the normal sorting preference (e.g. cyclic/random) doesn't apply. But as usual, "fixed" order is always used. So a client that is querying a BIND server and doesn't look beyond the first PTR RR, no matter how many times it queries the name. In other words, PTR RRset is different from A RRset, where the first record in the RRset might differ from query to query.

This is more restrictive in a world of IDNs, for choosing some names in a particular language. Briefly, according to the use of PTR, it is no meaning of returning an IDN in an unknown language.

The authors also believe that putting language information into address-to-name mappings will be beneficial to future applications.

The design purpose of the IPTR RR type is to provide a mechanism that can map an IP address to the corresponding IDN per language. It also means that IPTR suggests a new mapping algorithm for the reverse mapping by using an language information.

CNAME MUST continue to work for IPTR as it works now for PTR records.

The behavior of a resolver on the use of IPTR will be specified in a separate draft or a later version of this draft.

### **1.3 Functional Description**

DNS query and responses involving IPTR type MUST have the following



properties:

- When the QTYPE is IPTR, the corresponding IDNs SHOULD be returned in one response.
- The characters in the label MUST be encoded using UTF-8 [[RFC2279](#)].
- The entire label MUST be encoded EDNS [[RFC2671](#)].
- An exceptional handling of PTR for the IDN is REQUIRED.

## 2. IPTR definition

The structure of an IPTR RR is somewhat like the MX RR. In addition to the IP address in the IN-ADDR.ARPA domain and the domain name field (similar to a PTR RR), a new field called LANGUAGE has been defined. A domain name in an IPTR RR MUST be encoded in UTF8. And IDN in this document MUST be NAMEPREPPED. [[NAMEPREP](#)] Below is an example of an IPTR RR:

```
1.2.3.4.IN-ADDR.ARPA.      IPTR  "LANGUAGE" "name-in-utf8"
```

[RFC1766] describes the ISO 639/ISO 3166 conventions. A language name is always written in lower case, while country codes are written in upper case. At here, the "LANGUAGE" field in an IPTR RR SHOULD be done in a case-insensitive manner and MUST follow the conventions defined in [[RFC1766](#)].

For Example:

```
4.3.2.1.IN-ADDR.ARPA.      IPTR  "zh-CN"   "name-in-utf8"
4.3.2.1.IN-ADDR.ARPA.      IPTR  "zh-TW"   "name-in-utf8"
4.3.2.1.IN-ADDR.ARPA.      IPTR  "ja-JP"   "name-in-utf8"
4.3.2.1.IN-ADDR.ARPA.      IPTR  "ko-KR"   "name-in-utf8"
```

The notion of canonical names and aliases described in 3.6.2 [[RFC1034](#)], and 10.2 [[RFC2181](#)] MUST be preserved for IPTR record types. An IPTR RR SHOULD be limited to one primary IDN per LANGUAGE, similar to the a PTR RR.

## 3. IPTR on IPv6



Mapping IPv6 to IDNs can be similarly supported. This document recommends to continue using the IP6.INT domain defined in [RFC1886] for IPTR mappings. For example, the lookup corresponding to the address 4321:0:1:2:3:4:567:89ab would be:

```
b.a.9.8.7.6.5.0.4.0.0.0.3.0.0.0.2.0.0.0.1.0.0.0.0.0.0.1.2.3.4.IP6.INT.
IPTR  "LANGUAGE" "name-in-utf8"
```

#### 4. Packet format for IPTR

EDNS0[RFC2671] is REQUIRED to implement IPTR.

```

      0               1       2       3               4
bits 0 1 2 3 4 5 6 7 8 9 0 1...9 0...8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 ...
      +-+-+-+-+-+//+-+-+//+-+-+//+-+-+//+-+-+//+-+-+//+-+-+//+-+-+
      |0 1|      ELT      |  LANGUAGE      |      Size      | IDN label... |
      +-+-+-+-+-+//+-+-+//+-+-+//+-+-+//+-+-+//+-+-+//+-+-+//+-+-+
```

LANGUAGE: An argument for IPTR to define the kind of language used in the following IDN label. The size is 2 octets.

ELT: To be defined in [IDNE].

#### 5. Coexistence

##### 5.1 IDN Consideration

IPTR described above is based on "a set of IDNs", strictly speaking, a set of canonical IDNs. On the other hand, confusion about IDN, such as "IDN MUST exist with ASCII domain name" has led to a belief that PTR record should have exactly RRs in its RRSets. In short, the phenomenon "IDN ONLY" will exist. Thus, the exceptional handling of PTR is REQUIRED.

On the other hand, IDN is still RECOMMENDED to exist with more than one ASCII domain name.

##### 5.2 PTR Extension

In the case of "IDN ONLY", if IPTR RR is not NULL, PTR RR MUST contain a domain name in ACE to coexist with those IDN unaware systems. Else a "Syntax Error" message SHOULD be sent back, when an administrator configures DNS zone files.

##### 5.3 IPTR and PTR

It is a kind of backward compatible handle for those IDN unaware systems that can not provide the IPTR function. Besides, if a client can



not find the corresponding LANGUAGE IDN finally, then the corresponding PTR RR SHOULD be used as the answer.

## **6. IPTR query/response**

When the QTYPE is IPTR in a query, all of the corresponding IPTR RRs SHOULD be returned in one response. DNS messages are limited to 512 octets or less in size when sent over UDP. Therefore, if all the RRs cannot fit in one UDP packet, this draft describe two solutions. One is for recent environment and the other is for the near future.

### **6.1 Transport**

Today, DNS queries and responses are carried in UDP datagrams or over TCP connections. [\[RFC1034\]](#) specifies, IPTR RSet is RECOMMENDED to be returned in one response. The size of a DNS message could exceed 512 octets, when multiple RRs are present. Therefore, this draft makes the two following recommendations.

- "Use UDP first, if UDP is not large enough then change to TCP" is RECOMMENDED.

The server MUST send back the response with the TC bit set. Then the resolver SHOULD resend the query using TCP on server port 53(decimal). This behavior is consistent with the current DNS specification [\[RFC1035\]](#).

- In future, EDNS0 is REQUIRED to send large packets.

Then, before a client send a query to ask for IPTR record, it MUST query the server whether it knows the EDNS0 first. If the server knows EDNS0, then the client MAY send the IPTR query. Else, unfortunately, the client MUST change the QTYPE to PTR.

Hence, the size of the UDP payload is no longer limited to 512 octets any more.

### **6.2 Standard sample**

A resolver who wants to find the IDNs corresponding to an IP address 1.2.3.4 whould pursue a query of the form QTYPE=IPTR, QCLASS=IN, QNAME=4.3.2.1.IN-ADDR.ARPA, and would receive:





```

Header      +-----+
            | OPCODE=SQUERY, RESPONSE, AA          |
            +-----+
Question    | QNAME=4.3.2.1.IN-ADDR.ARPA., QCLASS=IN, QTYPE=IPTR      |
            +-----+
Answer      | 4.3.2.1.IN-ADDR.ARPA. IPTR  "zh-CN" "name1-in-utf8" |
            | 4.3.2.1.IN-ADDR.ARPA. IPTR  "zh-TW" "name2-in-utf8" |
            | 4.3.2.1.IN-ADDR.ARPA. IPTR  "zh-JP" "name3-in-utf8" |
            | 4.3.2.1.IN-ADDR.ARPA. IPTR  "ko-KR" "name4-in-utf8" |
            +-----+
Authority   | ...                                                    |
            +-----+
Additional  | ...                                                    |
            +-----+

```

## 7. IPTR Usage

The "foo1.example" in following samples MAY or MAY NOT be represented in the same characters.

```

4.3.2.1.IN-ADDR.ARPA  IPTR  "zh-TW" "[foo1.example] in utf8"
                      IPTR  "zh-CN" "[foo1.example] in utf8"
                      IPTR  "ja-JP" "[foo1.example] in utf8"
                      IPTR  "ko-KR" "[foo1.example] in utf8"

```

Moreover,

```

4.3.2.1.IN-ADDR.ARPA  IPTR  "zh-TW" "[foo1.example] in utf8"
                      IPTR  "zh-TW" "[foo2.example] in utf8"
                      ...
                      IPTR  "zh-CN" "[foo1.example] in utf8"
                      IPTR  "zh-CN" "[foo2.example] in utf8"
                      ...
                      IPTR  "ja-JP" "[foo1.example] in utf8"
                      IPTR  "ja-JP" "[foo2.example] in utf8"
                      ...
                      IPTR  "ko-KR" "[foo1.example] in utf8"
                      IPTR  "ko-KR" "[foo2.example] in utf8"
                      ...

```

will exist also. And "foo2.example" MUST be different from "foo1.example", if they are in signed with same LANGUAGE. Or a "Syntax Error" SHOULD be sent back, when an administrator configures the zone files. Furthermore "foo2.example" in the samples above MAY or MAY NOT be represented in the same characters.



Thus,

```
4.3.2.1.IN-ADDR.ARPA  IPTR "zh-TW" "[samefoo.sample] in utf8"
                        IPTR "zh-TW" "[samefoo.sample] in utf8"
```

occurs a "Syntax Error".

And,

```
4.3.2.1.IN-ADDR.ARPA  IPTR "zh-TW" "[samefoo.sample] in utf8"
                        IPTR "zh-TW" "[difffoo.sample] in utf8"
                        IPTR "zh-CN"  "[samefoo.sample] in utf8"
                        IPTR "ja-JP"  "[samefoo.sample] in utf8"
                        IPTR "ko-KR"  "[samefoo.sample] in utf8"
```

is allowed.

## 8. Open Issues

1. Is it necessary to let a IDN aware server to send back all of the corresponding IDNs to a resolver? Meanings,

Header	OPCODE=SQUERY, RESPONSE, AA	
Question	QNAME=4.3.2.1.IN-ADDR.ARPA., QCLASS=IN, QTYPE=IPTR	
Answer	4.3.2.1.IN-ADDR.ARPA. IPTR "zh-CN" "name1-in-utf8"	
	4.3.2.1.IN-ADDR.ARPA. IPTR "zh-CN" "name2-in-utf8"	
	4.3.2.1.IN-ADDR.ARPA. IPTR "zh-CN" "name3-in-utf8"	
	4.3.2.1.IN-ADDR.ARPA. IPTR "zh-TW" "name4-in-utf8"	
	4.3.2.1.IN-ADDR.ARPA. IPTR "ko-KR" "name5-in-utf8"	
	4.3.2.1.IN-ADDR.ARPA. IPTR "ko-KR" "name6-in-utf8"	
Authority	...	
Additional	...	

Or, just using current fixed/cyclic/random options to return one of the corresponding IDNs per LANGUAGE? In short, "one IP one IDN per LANGUAGE". Such like



```

Header      |-----+
            | OPCODE=SQUERY, RESPONSE, AA          |
            |-----+
Question    | QNAME=4.3.2.1.IN-ADDR.ARPA.,QCLASS=IN,QTYPE=IPTR |
            |-----+
Answer      | 4.3.2.1.IN-ADDR.ARPA. IPTR  "zh-CN" "name1-in-utf8" |
            | 4.3.2.1.IN-ADDR.ARPA. IPTR  "zh-TW" "name4-in-utf8" |
            | 4.3.2.1.IN-ADDR.ARPA. IPTR  "ko-KR" "name5-in-utf8" |
            | 4.3.2.1.IN-ADDR.ARPA. IPTR  "ko-KR" "name6-in-utf8" |
            |-----+
Authority    | ...                                     |
            |-----+
Additional  | ...                                     |
            |-----+

```

2. If QTYPE is IPTR, should an IDN aware server send all of the corresponding IDNs to the resolver? Is this kind of behavior friendly to implement the resolver? How about letting a server just feedback the corresponding PTR record, if a server doesn't find the corresponding LANGUAGE IDN that a client requires.

In the following case, it is wasteful to return all the corresponding IDNs to the clients.

```

4.3.2.1.IN-ADDR.ARPA  IPTR  "zh-TW" "[foo1.example] in utf8"
                       IPTR  "zh-TW" "[foo2.example] in utf8"
                       ...
                       IPTR  "zh-CN" "[foo1.example] in utf8"
                       IPTR  "zh-CN" "[foo2.example] in utf8"
                       ...
                       IPTR  "ja-JP" "[foo1.example] in utf8"
                       IPTR  "ja-JP" "[foo2.example] in utf8"
                       ...
                       IPTR  "ko-KR" "[foo1.example] in utf8"
                       IPTR  "ko-KR" "[foo2.example] in utf8"
                       ...

```

The benefit of the IPTR is introducing LANGUAGE. It SHOULD be used in query from clients, then servers can select minimum size of corresponding IDNs. For working this effectively, you should introduce default LANGUAGE if no corresponding LANGUAGE exists. The default MUST be ASCII. So that default IPTR can be natural extension of PTR. I.E.



4.3.2.1.in-addr.arpa. IN PTR ASCII-domain-name

is equivalent to

4.3.2.1.in-addr.arpa. IN IPTR "default" ASCII-domain-name

Of course, ASCII includes ACE.

3. According to the consideration above, how about the following thinking? That means a response MAY include not only a corresponding IDN in a specific LANGUAGE but also the LANGUAGE tags of the corresponding IDNs. And the client will load these LANGUAGE tags in the DNS cache for the next IPTR query.

## References

[IDNREQ] Zita Wenzel & James Seng, "Requirements of Internationalized Domain Names", [draft-ietf-idn-requirements](#).

[IDNE] Marc Blanchet & Paul Hoffman, "Internationalized domain names using EDNS", [draft-ietf-idn-idne](#).

[NAMEPREP] Paul Hoffman & Marc Blanchet, "Preparation of Internationalized Host Names", [draft-ietf-idn-nameprep](#).

[RFC1034] P. Mockapetris, "DOMAIN NAMES - CONCEPTS AND FACILITIES", November 1987, [RFC1034](#)

[RFC1035] P. Mockapetris, "DOMAIN NAMES - IMPLEMENTATION AND SPECIFICATION", November 1987, [RFC1035](#)

[RFC1766] H. Alvestrand, "Tags for the Identification of Languages", March 1999, [RFC 1766](#)

[RFC1886] S. Thomson, C. Huitema, "DNS Extensions to support IP version 6", December 1995, [RFC1886](#)

[RFC2181] R. Elz, R. Bush, "Clarifications to the DNS Specification", July 1997, [RFC2181](#)

[RFC2279] Francois Yergeau, "UTF-8, a transformation format of ISO 10646", January 1998, [RFC 2279](#).

[RFC2671] Paul Vixie, "Extension Mechanisms for DNS (EDNS0)", August 1999, [RFC 2671](#).

[ISO 639] ISO 639:1988 (E/F) - Code for the representation of names





of languages - The International Organization for Standardization, 1st edition, 1988 17 pages Prepared by ISO/TC 37 - Terminology (principles and coordination).

[ISO 3166] ISO 3166:1988 (E/F) - Codes for the representation of names of countries - The International Organization for Standardization, 3rd edition, 1988-08-15.

#### Acknowledgements

James Seng and Yoshiro Yoneya have given many comments in our e-mail discussions. Harald Alvestrand, Mark Davis have given many suggestions in the idn-wg mailing list discussions. And there are also a lot of people who have given us their comments in the idn-wg and BIND-user mailing list discussions.

#### Authors' Information

Hongbo Shi  
Waseda University  
3-4-1 Okubo, Shinjyuku-ku  
Tokyo, 169-8555 Japan  
shi@goto.info.waseda.ac.jp

Jiang Ming Liang  
i-DNS.net  
8 Temasek Boulevard  
#24-02 Suntec Tower Three  
Singapore 038988  
jiang@i-DNS.net

