

ENUM
Internet-Draft
Expires: January 6, 2006

L. Conroy
Roke Manor Research
July 5, 2005

Non-Terminal NAPTR Processing: A Modest Proposal
<[draft-conroy-enum-modestproposal-00.txt](#)>

Status of this Memo

By submitting this Internet-Draft, each author represents that any applicable patent or other IPR claims of which he or she is aware have been or will be disclosed, and any of which he or she becomes aware will be disclosed, in accordance with [Section 6 of BCP 79](#).

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>.

This Internet-Draft will expire on January 6, 2006.

Copyright Notice

Copyright (C) The Internet Society (2005).

Abstract

Recent Discussions within the IETF and in other fora have highlighted differences in interpretation of the set of standards associated with ENUM and DDDS, on which it relies. Specifically, the operation and semantics surrounding support for non-terminal NAPTRs has led to some confusion. This document is an attempt to add clarification to non-terminal NAPTR processing. In this, it clarifies [RFC3403](#). A subsequent document will build on this one to extend [RFC3761](#) further, permitting registration of non-terminal Enumservices.

Table of Contents

1.	Terminology	3
2.	Introduction	4
2.1	The Problem	4
3.	The solution	6
3.1	Clarification of RFC3403 Section 4.1	6
4.	Security Considerations	8
5.	IANA Considerations	9
6.	References	10
6.1	Normative References	10
6.2	Informative References	10
	Author's Address	11
	Intellectual Property and Copyright Statements	12

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 [BCP 14](#), [RFC2119](#) [[1](#)].

2. Introduction

[RFC3403](#) [2] defines the NAPTR resource record. It forms part of the set of standards ([3][4][2]) that collectively specify the Dynamic Delegation Discovery System (DDDS). Note that the examples given in the second half of [RFC3403](#) ([section 6](#) of that document) are exemplary rather than normative. The normative definitions of those DDDS applications are in [RFC3404](#) [5]/[6] and [RFC3761](#) [7]. In particular, note that [RFC3761](#) uses a slightly different syntax from the examples shown in this section.

The core algorithm for DDDS is specified in [RFC3402](#). This shows that the DDDS process is capable of looping through records that hold "rules", extracted from a DDDS-application specific rule database, until one such rule provides a final result or causes an exit from the DDDS process, or all rules are exhausted. Intermediate rules that do not cause such an exit are called "non-terminal" and processing such a non-terminal rule generates a new key that is used to extract further rules from the database.

One potential database that can be used with DDDS is the Domain Name System (DNS) [8]. A DNS Resource Record type suitable for carrying DDDS rules is the NAPTR, specified in [RFC3403](#).

For historical reasons (i.e. the original specification of NAPTRs preceded the development of DDDS), the fields that are defined for the NAPTR are a superset of the fields used in the DDDS algorithm. In particular, the DDDS priority field is represented within a NAPTR by the combination of the NAPTR Order and Preference elements, and the DDDS Substitution Expression is represented by the alternative NAPTR Regexp or Replacement elements.

The flags NAPTR element directly reflects the flags field used in the DDDS algorithm to specify the expected output of a rule. The flags field also indicates whether this rule is to be interpreted as terminal or non-terminal.

2.1 The Problem

The current DDDS specifications are not completely clear on how these NAPTR elements are used to reflect the DDDS rule fields. Individual DDDS application specifications have clarified the interpretation of the NAPTR Order and Preference element values when used with these DDDS applications. However, the main issue lies with the interpretation of the roles of the two elements that collectively reflect the DDDS Substitution Expression field; the NAPTR Regexp and Replacement elements.

[RFC3403](#) is specific; these two elements are mutually exclusive. This means that if the Regexp element is not empty then the Replacement element must be empty, and vice versa. However, it does not specify which is used with terminal and non-terminal rules.

The descriptive text of [section 4.1 of RFC3403](#) for the NAPTR Replacement element shows that this element holds an uncompressed domain name. Thus it is clear that this element cannot be used to deliver the terminal string for any DDDS application that does not have a domain name as its intended output.

However, the first paragraph of descriptive text for the NAPTR Regexp element has led to some confusion; it appears that the Regexp element is to be used to find "the next domain name to lookup". A client program processing the DDDS application may need to examine each NAPTR to decide whether the Regexp element or instead the Replacement element is to be used to construct the key (a domain name) to be used next in non-terminal rule processing.

Given that a NAPTR holding a terminal rule (a "terminal NAPTR") must use the Substitution expression field to generate the expected output of that DDDS application, the Regexp element is also used in such rules. Indeed, unless that DDDS application has a domain name as its output, the Regexp element is the only possibility.

Thus from the descriptive text of this section, a Replacement element can be used only in NAPTRs holding a non-terminal rule (a "non-terminal NAPTR") unless that DDDS Application has a domain name as its terminal output, whilst the alternative Regexp element may be used either to generate a domain name as the next key to be used in the non-terminal case, or to generate the output of the DDDS application.

Note that each DDDS application is free to specify the set of flags to be used with that application. This includes specifying whether a particular flag is associated with a terminal or non-terminal rule, and also to specify the interpretation of an empty flags field (i.e. whether this is to be interpreted as a terminal or non-terminal rule, and if it is terminal, then the expected output).

The general case in which a client program must check which of the two elements to use in non-terminal NAPTR processing complicates implementation, and this interpretation has NOT been made in current ENUM examples "out in the wild". It would be useful to define exactly when a client program can expect to process the Regexp element and when to expect to process the Replacement element, if only to improve robustness.

3. The solution

3.1 Clarification of [RFC3403 Section 4.1](#)

In those DDDS application specifications that have been released so far, the empty flags field has been used to indicate a non-terminal rule. As described in [RFC3403](#), the DDDS application is also free to specify a flag as being associated with a non-terminal rule.

A two-part convention is proposed for all DDDS applications that use NAPTRs to store their rules.

In the first part of this convention, the empty flag field **MUST** be interpreted as being associated with a non-terminal rule, and that in this case, that non-terminal rule **MUST** use the (non-empty) Replacement element to hold the domain name that forms the "next key" output from this non-terminal rule.

In the second part of the convention, where a DDDS application defines a flag as being associated with a non-terminal rule, the NAPTR containing this rule **MUST** use the Regexp element to generate and output the "next key".

This convention allows the client program to decide, merely by inspection of the flags element of the NAPTR, whether it should expect to process the Replacement or Regexp element. It also allows more rigorous validation to be applied if required against the output of the Regexp processing; it will be clear from the specific flag whether this is to be interpreted as a domain name or some other output (such as a URI, in the case of ENUM). Finally, this convention does not change any existing DDDS application specification - it merely clarifies what is currently not completely specified.

Thus, it is proposed that the descriptive text for the Flags element within [section 4.1 of RFC3403](#) (in particular, the second paragraph) should be interpreted as follows:

FLAGS

A <character-string> containing flags to control aspects of the rewriting and interpretation of the fields in the record. Flags are single characters from the set A-Z and 0-9. The case of the alphabetic characters is not significant. The field can be empty.

Where the field is empty, the rule is interpreted as non-terminal, and the Replacement field is used to hold the next key output by the rule. Where the field is non-empty, it is up to the

Application specifying it is using this Database to define the Flags in this field. It must define which ones are terminal and which ones are not. In the case where a flag is defined as being non-terminal, the Regexp field will be used to generate the next key output by this rule.

4. Security Considerations

The clarification described in this document does not appear to have any security considerations over and above those already analysed in the DDDS specifications. The intent of this document is to specify more clearly what fields should be used within a NAPTR resource record for terminal and non-terminal DDDS processing and to "tighten" the specification of NAPTR field content. Whilst this does reduce the flexibility of DDDS applications to make their own choices on field content and interpretation, it does simplify the processing required of clients that handle NAPTRs.

5. IANA Considerations

Under the framework defined in [RFC3402](#), IANA accepts registration requests for new DDDS applications only after the specifications that define the operation of these DDDS applications have been processed by the IETF. Thus there are no further requirements on IANA, as it is assumed that this document will have been considered as part of the evaluation of any new DDDS application by the IETF.

6. References

6.1 Normative References

- [1] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", [RFC 2119](#), [BCP 14](#), March 1997.
- [2] Mealling, M., "Dynamic Delegation Discovery System (DDDS) Part Three: The Domain Name System (DNS) Database", [RFC 3403](#), October 2002.
- [3] Mealling, M., "Dynamic Delegation Discovery System (DDDS) Part One: The Comprehensive DDDS", [RFC 3401](#), October 2002.
- [4] Mealling, M., "Dynamic Delegation Discovery System (DDDS) Part Two: The Algorithm", [RFC 3402](#), October 2002.
- [5] Mealling, M., "Dynamic Delegation Discovery System (DDDS) Part Four: The Uniform Resource Identifiers (URI)", [RFC 3404](#), October 2002.
- [6] Mealling, M., "Dynamic Delegation Discovery System (DDDS) Part Five: URI.ARPA Assignment Procedures", [RFC 3405](#), October 2002.
- [7] Faltstrom, P. and M. Mealling, "The E.164 to Uniform Resource Identifiers (URI) Dynamic Delegation Discovery System (DDDS) Application (ENUM)", [RFC 3761](#), April 2004.
- [8] Mockapetris, P., "DOMAIN NAMES - CONCEPTS AND FACILITIES", [RFC 1034](#), November 1987.

6.2 Informative References

- [9] Bradner, S., "The Internet Standards Process -- Revision 3", [RFC 2026](#), [BCP 9](#), October 1996.
- [10] Bradner, S., "IETF Rights in Contributions", [BCP 78](#), [RFC 3978](#), March 2005.
- [11] Bradner, S., "Intellectual Property Rights in IETF Technology", [BCP 79](#), [RFC 3979](#), March 2005.

Author's Address

Lawrence Conroy
Roke Manor Research
Roke Manor
Romsey
United Kingdom

Phone: +44-1794-833666

Email: lwc@roke.co.uk

Intellectual Property Statement

The IETF takes no position regarding the validity or scope of any Intellectual Property Rights or other rights that might be claimed to pertain to the implementation or use of the technology described in this document or the extent to which any license under such rights might or might not be available; nor does it represent that it has made any independent effort to identify any such rights. Information on the procedures with respect to rights in RFC documents can be found in [BCP 78](#) and [BCP 79](#).

Copies of IPR disclosures made to the IETF Secretariat and any assurances of licenses to be made available, or the result of an attempt made to obtain a general license or permission for the use of such proprietary rights by implementers or users of this specification can be obtained from the IETF on-line IPR repository at <http://www.ietf.org/ipr>.

The IETF invites any interested party to bring to its attention any copyrights, patents or patent applications, or other proprietary rights that may cover technology that may be required to implement this standard. Please address the information to the IETF at ietf-ipr@ietf.org.

Disclaimer of Validity

This document and the information contained herein are provided on an "AS IS" basis and THE CONTRIBUTOR, THE ORGANIZATION HE/SHE REPRESENTS OR IS SPONSORED BY (IF ANY), THE INTERNET SOCIETY AND THE INTERNET ENGINEERING TASK FORCE DISCLAIM 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.

Copyright Statement

Copyright (C) The Internet Society (2005). This document is subject to the rights, licenses and restrictions contained in [BCP 78](#), and except as set forth therein, the authors retain all their rights.

Acknowledgment

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

