Server Index Query (SIQ) Protocol
draft-irtf-asrg-iar-howe-siq-03.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/1id-abstracts.html

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

Abstract

The Server Index Query (SIQ) protocol is intended to provide a standard means by which a mail exchange (MX) server can query one or more external services for scoring based on facts or reputation of an IP-domain pair. This document specifies the communication protocol used to transmit the IP/domain query and return the query response. The implementation, correctness of results, and/or management of SIQ servers is beyond the scope of this document.

Table of Contents

1. Introduction
   1.2 Terminology
2. Overview
3. Query & Response UDP
   3.1 UDP Query Format
   3.2 UDP Response Format
4. Query & Response HTTP
   4.1 HTTP Query Format
   4.2 HTTP Response Format
   4.3 HTTP Query & Response Example
5. Rationale & Supplemental Information
   5.1 About Query Type
   5.2 Use of IPv4-compatible IPv6 address format
1. Introduction

The proposed SIQ protocol is intended to provide a light, reliable way for an inbound email server to query a "reputation" service and receive a useful response. Because an IP/domain pair is included in SIQ protocol queries, the response may score the IP network, domain ownership, and the quality of the relationship (denied, affirmed, inferred, undetectable) between the IP and domain.

A variety of anti-forgery techniques have been proposed in recent years. Many of the proposals require a domain owner to announce which outbound servers are authorized sources of mail for their domain [SPF] and/or use cryptographic methods of signing a message by domain [DKIM]. These proposals only solve one piece of a problem by providing some means of authentication.

With authentication inbound mail systems, will over time, only accept mail from authenticated sources. While this might prevent forgeries and phishing schemes, it does not prevent abusive behaviour ie. a junk mailer can use authentication just as easily as a legitimate sender. So most of the authentication schemes foresee the need for external reputation systems that provide inbound mail servers with information concerning an outbound mail source. The SIQ protocol is put forth as a protocol for inbound servers to use in communicating with such reputation services or systems.

One possible niche for SIQ protocol servers is to do the heavy lifting - discovering, caching and collating all that can be divined from knowing an IP, a domain name, and the relationship of the IP to the domain name for the purposes of sending emails. The result for any particular IP/domain pair query may then be efficiently returned in composite form to the inbound server (SIQ protocol query client).

The basis for the "reputation" scoring may be objective factors such as longevity, stability, and identifiability. Some reputation services may choose subjective factors such as judgements about content, morality, historical business practices, etc. The distinction between objective and subjective reputation scoring is beyond the scope of this document; the authors do want to point out that services in the class of "reputation", MAY be objectively based on measurable and observable facts, rather than based on opinion or payment.

The SIQ protocol supports differentiated pre-DATA [RFC2821] and post-DATA queries. Pre-DATA queries have a limited scope of information
they can provide; they refer to the connecting SMTP client IP and the MAIL FROM (aka envelope-from) domain. Post-DATA queries may pose queries about domains in URLs or email domains found in the body of message, domains in particular headers such as Errors-To, Reply-To, From, Resent-From, or even message checksums or fingerprints, etc. Thus any SIQ based reputation server may respond appropriately, according to the specific query type.

As a specific example, query clients may be designed for anti-phishing functions with post-DATA queries, such as via marking the suspicious emails with a warning. The criteria for evaluating these queries would be very different from the criteria for evaluating the pre-DATA, MAIL FROM domain and sending server IP address pairs.

1.2 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 [RFC2119].

2. Overview

The SIQ protocol uses a query & response model over UDP with an alternative method over TCP/HTTP. Implementation and specification borrows ideas from the Domain Name Service (DNS) defined by [RFC1035] and the Hypertext Transfer Protocol (HTTP/1.1) defined by [RFC2616].

Upon receiving a new inbound SMTP connection, an MX sends one or more queries via UDP or HTTP. The queries consist of the connecting client IP address, a query domain, along with other housekeeping bits which will be described in detail later in this document. The query domain is either the MAIL FROM domain or a domain found in the DATA content. See section 5.1.

Depending on the type of query made, the SIQ score returned can be used to reject or accept, with optional tagging for sorting or further processing. Support for per RCPT domain processing is anticipated by the protocol design and may optionally be provided for multi-RCPT messages, dependent on query client implementation.

3. Query & Response UDP

A client attempts to contact one or more SIQ servers with a query packet on port UDP/6262 (see section 7) for a response using an exponential backoff algorithm or similar query retry scheme. A SIQ server implementation MUST support UDP queries and responses.

In the event that no answer is found, the client MUST assume an 'UNKNOWN' result and continue to handle the message subject to local policy.

3.1 UDP Query Format

Query & Response UDP packets MUST NOT be longer than 512 octets. If a query packet would be longer than 512 octets, an HTTP request MUST be performed instead.

The Query packet has the following format:
where:

VERSION       The packet format described by this document is
               version one (1).

QT            Type of query (see section 5.1):
               0 = MAIL FROM
               1 = DATA

ID            A 16 bit identifier assigned by the program that
               generates any kind of query. This identifier is
               copied into the corresponding reply and can be used
               by the requester to match up replies to outstanding
               queries.

IPV6          The octets of the IPv6 address for the connecting mail
               client in network order. IPv4 addresses are encoded
               according to [RFC2373] section 2.5.3 "IPv4-compatible
               IPv6 address". See section 5.2.

QD-LENGTH     An 8-bit unsigned value that gives the number of
               octets in the QD section.

QD            The US-ASCII octets of the MAIL domain argument or a
               domain from the DATA content.

EXTRA-LENGTH  An 8-bit unsigned value that gives the number of
               octets in the EXTRA section.

EXTRA-ID      A 4 octet identifier or version number used to
               indicate a specific service format used for the
               EXTRA section.
EXTRA          Any octets possibly containing supplemental data
intended for the server. The use of this field is
optional and specific to the client / server
implementation.

NOTE that the overall length of the packet must not exceed 512
octets. In instance where both the QD and EXTRA would be 255
octets each, then the QD section takes precedence over the
EXTRA section or switch to using the HTTP format. See section 4.

3.2 UDP Response Format

A response packet has the following format:

```
+0  |        VERSION        |        SCORE          |
+2  |                      ID                       |
+4  |       IP-SCORE        |     DOMAIN-SCORE      |
+6  |       REL-SCORE       |      TEXT-LENGTH      |
+8  |                      TTL                      |
+10 |       DEVIATION       |     EXTRA-LENGTH      |
+12 |                     TEXT                      |
+14 |                     EXTRA                      |
```

where:

VERSION        The packet format described by this document is
version one (1).

ID             A 16 bit identifier assigned by the client that
generates any kind of query. This identifier is
copied into the corresponding reply and can be used
by the client to match up replies to outstanding
queries.

SCORE          An 8-bit signed value between -127 and 127, where
values other than -5 through 100 are reserved.

-4   A generic ERROR result. The TEXT section SHOULD
     contain human readable log information.
-3 TEMP-REDIRECT the request is to be redirected to the SIQ server given in the TEXT section, which is US-ASCII octets, with the given format:

```
ADDRESS <space> PORT
```

ADDRESS is either an IPv6 address, where IPv4 addresses are encoded according to [RFC2373] section 2.5.3 "IPv4-compatible IPv6 address". See section 5.2; or a fully qualified domain name of a machine.

Note that the length of the TEXT section is limited to 255 octets and so if an ADDRESS specified as a FQDN exceeds this length, then the IPv6 address MUST be used instead.

PORT is an unsigned 16-bit decimal value for the server port to be used.

A TEMP-REDIRECT response MUST NOT be cached.

-2 A TEMPFAIL result. The mail server is recommended to reject the current message with a 4xx response.

-1 An UNKNOWN result. The server has no data pertaining to the IP and/or domain queried.

0..100 Composite Score: The following interpretations of the score SHOULD be applied:

0 Recommend that message be quarantined, tagged, or rejected according to local policy.

50 A neutral score indicates that the server’s data is inconclusive.

100 The message SHOULD be accepted without need for further filter processing.

The TEXT section MAY contain a report summary suitable for logging purposes and/or inclusion in informational message headers.

Any permanent or temporary errors, for example a lookup error or service restarting, MUST be represented as an ERROR result. An ERROR result MAY be treated like an UNKNOWN response and MUST not be cached.

In the event of an UNKNOWN response, the mail server SHOULD continue to handle the message subject to local policy.

The composite score value is the one intended for primary use and is generated by some server specific function based on the other scores. How this value is computed is beyond the scope of this document and intentionally left unspecified.
IP-SCORE          Each is an 8-bit value between -1 and 100. Each is a
DOMAIN-SCORE      standardized percentage based on SIQ server data
REL-SCORE         pertaining to an individual IP, individual domain, and
                   the relationship between IP and domain respectively.

-1   An UNKNOWN result, possibly due to insufficient
data.

0..100
Percentage score, 0 is unfavourable, 50 is neutral, and 100 is favourable. See SCORE
above.

These scores MAY be provided as supplemental information by the server. They may be ignored,
logged, and/or used for supplemental tests by the query client.

DEVIATION         An 8-bit signed value between -1 and 100. See
                   section 5.4.

-1   An UNKNOWN result for when the server does not provide this information.

0..100
It is the standard deviation rounded down to the nearest integer and indicates the dispersion
around the SCORE.

TTL               Time-To-Live is a 16 bit unsigned integer that
                   specifies the time interval in seconds that this
                   response may be cached before the source of the
                   information should again be consulted. Zero values
                   are interpreted to mean that response can only be
                   used for the transaction in progress, and should not
                   be cached.

TEXT-LENGTH       An 8-bit unsigned value that gives the number of
                   octets in the TEXT section.

TEXT              US-ASCII octets that provide a brief human readable
                   commentary about the response suitable for logging
                   purposes.

EXTRA-LENGTH      An 8-bit unsigned value that gives the number of
                   octets in the EXTRA section.

EXTRA-ID          A 4 octet identifier or version number used to
                   indicate a specific service format used for the
                   EXTRA section.

EXTRA             Any octets possibly containing supplemental data
                   intended for the client. The use of this field is
                   optional and specific to the client / server
                   implementation.

NOTE that the overall length of the packet must not exceed 512 octets. In instance where both the
TEXT and EXTRA would be 255 octets each, then the EXTRA section should take precedence over
the informational TEXT section.
4. Query & Response HTTP

The HTTP/1.1 protocol [RFC2616] (see section 5.3) is used to provide a reliable means of validating an IP and domain name combination. It MAY be used instead of the UDP method to obtain a more detailed answer. A SIQ server SHOULD support HTTP queries & responses.

A client query consists of a GET, HEAD, or POST request to an HTTP or HTTPS server on the standard ports 80 or 443 respectively. This allows for the efficient use of caching web proxies, to enable it to also serve a dual function allowing standard web browsers to query it, or provide privacy through an encrypted connection. Other ports MAY be used by used if desired.

The HEAD method SHOULD be used by unattended clients to minimize network traffic. A POST request is used in the event that the GET/HEAD request results in a response code 414 (URI too long) or to obtain an uncached result.

The HEAD method SHOULD be used by unattended clients to minimize network traffic. A POST request is used in the event that the GET/HEAD request results in a response code 414 (URI too long) or to obtain an uncached result.

4.1 HTTP Query Format

The following is the general format for the HTTP request and adheres to [RFC2616] :

```
HEAD /siq/protocol-VERSION HTTP/1.1
Host: SIQ-SERVER:PORT
SIQ-Query-Type: QT
SIQ-Query-IP: IPV6
SIQ-Query-Domain: QD
SIQ-Extra-ID: EXTRA-ID
SIQ-Extra: EXTRA
```

Where:

VERSION        The URI format described by this document is version one (1). See example below.
SIQ-SERVER     The fully qualified domain name of the SIQ server being consulted. This header is required by [RFC2616]. See example below.
PORT           When the HTTP port is something other than port 80, then :PORT is that port number. The :PORT is optional when the port is 80 (see [RFC2616]). See example below.
IPV6           The IPv6 address of the connecting mail client using colon notation, in the US-ASCII charset. IPv4 addresses are encoded according to [RFC2373] section 2.5.3 using "IPv4-compatible IPv6 address". This field MUST be specified. See section 5.2.
QT             See QT defined in section 3.1. This field MUST be specified.
QD See QD defined in section 3.1. This field MUST be specified.

EXTRA-ID See EXTRA-ID defined in section 3.1.

EXTRA See EXTRA defined in section 3.1.

It is possible to supply additional information through HTTP request headers. Headers beginning with SIQ- are reserved for future revisions of this document. Vendors can supply additional request headers and the SHOULD use a vendor specific prefix to denote their name space. Any header that contains binary data, MUST be MIME word encoded according to [RFC2047].

4.2 HTTP Response Format

The HTTP server may be implemented in any manner (see section 5.3), such as a custom and dedicated server or as generic HTTP server using a script or CGI to process the request. The choice and manner of the server side implementation is beyond the scope of this document.

The response format adheres to [RFC2616]:

```
HTTP/1.1 STATUS REASON
SIQ-Score: SCORE
SIQ-Comment: TEXT
SIQ-IP-Score: IP-SCORE
SIQ-Domain-Score: DOMAIN-SCORE
SIQ-Relationship-Score: REL-SCORE
SIQ-Deviation: DEVIATION
SIQ-TTL: TTL
SIQ-Extra-ID: EXTRA-ID
SIQ-Extra: EXTRA
```

where:

STATUS The standard HTTP response code. A successful response can be either "200 OK" or "204 No Content", with the remainder of the response placed in reserved SIQ-response headers. In the case of a "200 OK" response, the body is optional and unspecified. A "301 Moved Permanently", "302 Found", "303 See Other", or "307 Temporary Redirect" status codes and a Location: header indicate that the client MUST redirect the request to another server. The [RFC2616] semantics of these codes MUST be respected.

A "404 Not Found" MUST be treated as an UNKNOWN result.

A "414 URI Too Long" indicates the client MUST either request again using POST, or treat as an ERROR.

The semantics of the remaining [RFC2616] response codes SHOULD be respected if relevant, such as 401 Unauthorized for authentication, or treated as an ERROR, such as 403 Forbidden.

REASON Standard HTTP textual description of the STATUS.
See section 3.2 for the definition of each SIQ- header’s value.

It is possible to supply additional information through HTTP response headers. Headers beginning with SIQ- are reserved for future revisions of this document. Vendors MAY supply additional response headers in which case they SHOULD use a vendor specific prefix to denote their name space. Any header that contains binary data, MUST be MIME word encoded according to [RFC2047].

4.3 HTTP Query & Response Example

The following is an example of a HTTP based SIQ query:

```
HEAD /siq/protocol-1 HTTP/1.1
Host: siq-host.example.com
SIQ-Query-Type: 0
SIQ-Query-IP: 0:0:0:0:0:0:C000:0225
SIQ-Query-Domain: from.domain.tld
```

And its subsequent HTTP response:

```
HTTP/1.1 204 No Content
SIQ-Score: 95
SIQ-Comment: Hi Mom! Look no hands.
SIQ-IP-Score: 100
SIQ-Domain-Score: 80
SIQ-Relationship-Score: 90
SIQ-Deviation: 0.234
SIQ-TTL: 3600
```

5. Rationale & Supplemental Information

5.1 About Query Type

There are two types of requests: MAIL FROM or DATA.

MAIL FROM requests use the MAIL FROM domain along with the connecting server IP address. This pre-DATA type of request is intended for the purpose of accepting or rejecting mail prior to accepting the message content.

DATA requests use a domain found in an email address, URL, or some other network reference extract from the content of a message combined with the connecting server IP. The post-DATA type of request allows for reputation content filtering, anti-phishing, URLBL, etc.

This protocol intentionally never passes the user portion of email addresses to the third party SIQ server. Only the domain part of MAIL FROM or DATA element arguments is passed, for privacy reasons and to prevent SIQ servers from being used for data mining and surveillance purposes.

5.2 Use of IPv4-compatible IPv6 address format

[RFC2373] section 2.5.4 specifies two ways to encode IPv4 addresses: "IPv4-compatible IPv6 address" and "IPv4-mapped IPv6 address". The former is essentially the IPv4 address zero padding
the high order bits to IPv6 and the latter specifies padding the IPv4 with 16 one bits then the remainder as zero bits. The former was chosen for ease of coding and because "IPv4-mapped IPv6 address" states for IPv4-only nodes, which the SIQ client cannot distinguish; how are we to know if the client connection does or does not support IPv6 just because they connected from IPv4 space.

5.3 HTTP 1.0 and 1.1

Preliminary versions of this document used the HTTP/1.0 version number [RFC1945] for HTTP requests to signal to the server that the request is not a persistent connection, without the need of extra headers. It also allows the SIQ protocol to be implemented on older HTTP servers.

However, for consistency and clarity of interpretation, HTTP/1.1 [RFC2616] is assumed for complete and conforming client and server implementations. HTTP/1.0 semantics may be used for minimalist implementations, though not recommended for production systems.

Also HTTP/1.1 allows for persistent connections, which are better suited to multiple queries (pre-DATA and post-DATA) reducing the overhead of building up and tearing down individual TCP connections.

The use of the MIME type "multipart/form-data" for encoding POST requests is not explicitly disallowed, since many web servers provide the necessary framework to support its use. However, it does added extra complexity to implementation that are created from scratch and so its use is discouraged.

HTTP clients MAY be validated using the methods provided for by [RFC2616] WWW-Authenticate: and Authorization: headers and/or using TLS/SSL certificates. A minimum SIQ implementation MUST support HTTP Basic authentication [RFC2617]. In the interest of stronger security, a SIQ implementation SHOULD support HTTP Digest authentication [RFC2617].

5.4 Deviation Value.

During discussions about the merits of including some form of confidence field, it was pointed out that server implementations that use some form of weighted rule system could provide a crafted SCORE where 50% would indicate uncertain/poor data and that the SCORE implementation could lean more and more heavily towards 0% or 100% to reflect both the score and its confidence in a unified manner. Some believed this was not clear enough or that not all implementations would lend themselves well to such a use.

Instead the standard deviation as defined by statistical probability was added as it has a clear definition and can be used to express confidence. Briefly, the standard deviation is the dispersion of data around the average or mean.

If $X$ is your data sample and the SCORE is equivalent to the statistical mean computed from the distribution of $X$, then the DEVIATION is the standard deviation of $X$ expressed as an integer. The standard deviation is an expression of the spread or dispersion of the probability distribution. If you have a large standard deviation, then the distribution of $X$ is spread over a large interval around the mean; a small standard deviation indicates that the distribution of $X$
clusters more closely around the mean; a standard deviation of zero indicates the distribution of X is singular with all the probability being concentrated on a single point that is the mean. In other words, the smaller the standard deviation, then the more confident that future random draws from X will be closer or equal to the mean probability.

The use of -1 UNKNOWN are for implementations where the SCORE is not based on computation, is not appropriate, or of no utility, for example certification based on opinion or payment. Such implementations are strongly discourage from simply stating the DEVIATION as a constant zero (0) as this may have legal implications in some jurisdictions in the event of a dispute.

5.5 Message Score

It was indicated that some reputation services may desire information such as a message finger print or checksum value from the SIQ client and would desire to return a score based on that information instead of IP and domain.

However, it was not clear how to allocate the necessary space in the UDP query packet structure without getting into the details of specific message finger print or checksum implementations, which this document avoids specifying.

Future versions of this protocol may specify a message finger print or checksum implementation and additional query & response fields.

5.6 An Exponential Backoff Algorithm

Use of this particular UDP query retry algorithm is not required, but specified here as supplemental information. It is based on a similar technique used by some DNS client implementations.

Let S be the number of servers available to query. Let T be the initial timeout. Let R be the current round, where a round is defined to be S query attempts. Let t be the timeout-per-query before trying the next server in turn. Then the following function can be used to compute t:

\[
\begin{align*}
    t &= T & \text{for } R = 0 \\
    t &= 2^R \cdot T & \text{for } R > 0 \\
    t &= \lfloor \frac{T}{S} \rfloor & \text{for } R > 0
\end{align*}
\]

For example, starting with an initial timeout-per-round value of 5 seconds and a maximum of 4 rounds, the maximum time spent making a query:

<table>
<thead>
<tr>
<th>S</th>
<th>R=0</th>
<th>R=1</th>
<th>R=2</th>
<th>R=3</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 server: 5+ 10+ 20+ 40  = 75 seconds</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2 servers: 5+5+ 5+5+ 10+10+ 20+20  = 80 seconds</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3 servers: 5+5+5+ 3+3+3+ 6+6+6+ 13+13+13  = 87 seconds</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

For example, starting with an initial timeout value of 3 seconds and a maximum of 4 rounds:

<table>
<thead>
<tr>
<th>S</th>
<th>R=0</th>
<th>R=1</th>
<th>R=2</th>
<th>R=3</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 server: 3+ 6+ 12+ 24  = 45 seconds</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
2 servers: 3+3+ 3+3+ 6+6+ 12+12 = 48 seconds
3 servers: 3+3+3+ 2+2+2+ 4+4+4+ 8+8+8 = 51 seconds

6. Security Considerations

UDP queries benefit from their compactness and speed, but they are sent in the clear and lack any real form of authentication. The concept of a query ID being returned in a response packet is similar to that used in DNS [RFC1035]. Its purpose is for tracking requests. A response containing a matching query ID should not be relied upon as proof that the response came from a known and reliable SIQ server, even when verified against the source IP, any more than one would rely on a response from a UDP DNS server as guaranteed to be accurate. IP address spoofing and man in the middle attacks are an issue, since they could be used to falsify queries or responses. Where security is a higher priority than performance, HTTP queries SHOULD be used instead, preferably over a TLS/SSL connection.

The information passed using either the UDP packet queries or HTTP queries, such as the combination of sender’s IP, MAIL FROM, and the RCPT TO domain may pose some privacy issues. Similar information already appears in message trace headers and those headers may have already been viewed and logged by intermediate MX servers during transit. Taking this perspective, the queries make use of information that may have already been revealed else where.

However, with today’s Internet privacy paranoia, a SIQ client MAY choose to make HTTP queries over a TLS/SSL connection at the sake of the speed and convenience offered by UDP queries or unencrypted HTTP queries.

7. IANA Considerations

Application is to be made for a User (Registered) Port Number using TCP and UDP. This port number would replace the proposed value of 6262 outlined above.

8. Normative References


[RFC2045] Multipurpose Internet Mail Extensions (MIME) Part One: Format of Internet Message Bodies; N. Freed, N. Borenstein November 1996

[RFC2047] MIME (Multipurpose Internet Mail Extensions) Part Three: Message Header Extensions for Non-ASCII Text; K. Moore;
November 1996


9. Informational References


[ DKIM ] DomainKeys Identified Mail (DKIM). E. Allman, et. al. IETF draft-allman-dkim-base-00.txt

10. Authors’ Addresses

Anthony C. Howe
42 av. Isola Bella
06400 Cannes, France
<achowe@snert.com>

April Lorenzen
PO Box 293, Jamestown
RI 02835, USA
<ietf.siq@codelock.com>

Petru Paler
Brasov, Romania
<petru@paler.net>

Derek J. Balling
557 Broadway Apt 37A
Port Ewen, NY 12466
<dredd@megacity.org>

Acknowledgements

We would like to thank Eric Allman for his constructive feedback and suggestions in latter revisions to this document.

Comments

Comments on this draft are welcome. In the interests of openness, rather than contacting the authors directly, please post to:

http://ors.blogs4change.org/

Copyright Statement

Copyright (C) The Internet Society (2006). 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.

Disclaimer

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.

Acknowledgment

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

Expires

This Internet Draft expires 14 September 2006.