Network Working Group
Request for Comments: 2186
Category: Informational

D. Wessels
K. Claffy
National Laboratory for Applied
Network Research/UCSD

September 1997

Internet Cache Protocol (ICP), version 2

Status of this Memo

This memo provides information for the Internet community. This memo does not specify an Internet standard of any kind. Distribution of this memo is unlimited.

Abstract

This document describes version 2 of the Internet Cache Protocol (ICPv2) as currently implemented in two World-Wide Web proxy cache packages[3,5]. ICP is a lightweight message format used for communicating among Web caches. ICP is used to exchange hints about the existence of URLs in neighbor caches. Caches exchange ICP queries and replies to gather information to use in selecting the most appropriate location from which to retrieve an object.

This document describes only the format and fields of ICP messages. A companion document (RFC2187) describes the application of ICP to Web caches. Several independent caching implementations now use ICP, and we consider it important to codify the existing practical uses of ICP for those trying to implement, deploy, and extend its use for their own purposes.

1. Introduction

ICP is a message format used for communicating between Web caches. Although Web caches use HTTP[1] for the transfer of object data, caches benefit from a simpler, lighter communication protocol. ICP is primarily used in a cache mesh to locate specific Web objects in neighboring caches. One cache sends an ICP query to its neighbors. The neighbors send back ICP replies indicating a "HIT" or a "MISS."

In current practice, ICP is implemented on top of UDP, but there is no requirement that it be limited to UDP. We feel that ICP over UDP offers features important to Web caching applications. An ICP query/reply exchange needs to occur quickly, typically within a second or two. A cache cannot wait longer than that before beginning to retrieve an object. Failure to receive a reply message most likely means the network path is either congested or broken. In either case we would not want to select that neighbor. As an indication of immediate network conditions between neighbor caches, ICP over a lightweight protocol such as UDP is better than one with the overhead of TCP.

In addition to its use as an object location protocol, ICP messages can be used for cache selection. Failure to receive a reply from a cache may indicate a network or system failure. The ICP reply may include information that could assist selection of the most appropriate source from which to retrieve an object.

ICP was initially developed by Peter Danzig, et. al. at the University of Southern California as a central part of hierarchical caching in the Harvest research project[3].

ICP Message Format

The ICP message format consists of a 20-octet fixed header plus a variable sized payload (see Figure 1).

NOTE: All fields must be represented in network byte order.

Opcode
One of the opcodes defined below.

Version
The ICP protocol version number. At the time of this writing, both versions two and three are in use. This document describes only version two. The version number field allows for future development of this protocol.

Message Length

```
     0                   1                   2                   3
    0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |     Opcode    |    Version    |         Message Length        |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |                         Request Number                        |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |                            Options                            |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |                          Option Data                          |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |                       Sender Host Address                     |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |                                                               |
   |                            Payload                            |
   /                                                               /
   /                                                               /
   |                                                               |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

FIGURE 1: ICP message format.

The total length (octets) of the ICP message. ICP messages MUST not exceed 16,384 octets in length.

Request Number
An opaque identifier. When responding to a query, this value must be copied into the reply message.

Options
A 32-bit field of option flags that allows extension of this version of the protocol in certain, limited ways. See "ICP Option Flags" below.

Option Data
A four-octet field to support optional features. The following ICP features make use of this field:

The ICP_FLAG_SRC_RTT option uses the low 16-bits of Option Data to return RTT measurements. The ICP_FLAG_SRC_RTT option is further described below.

Sender Host Address

The IPv4 address of the host sending the ICP message. This field should probably not be trusted over what is provided by getpeername(), accept(), and recvfrom(). There is some ambiguity over the original purpose of this field. In practice it is not used.

Payload

The contents of the Payload field vary depending on the Opcode, but most often it contains a null-terminated URL string.

2. ICP Opcodes

The following table shows currently defined ICP opcodes:

Value	Name
0	ICP_OP_INVALID
1	ICP_OP_QUERY
2	ICP_OP_HIT
3	ICP_OP_MISS
4	ICP_OP_ERR
5-9	UNUSED
10	ICP_OP_SECHO
11	ICP_OP_DECHO
12-20	UNUSED
21	ICP_OP_MISS_NOFETCH
22	ICP_OP_DENIED
23	ICP_OP_HIT_OBJ

ICP_OP_INVALID

A place holder to detect zero-filled or malformed messages. A cache must never intentionally send an ICP_OP_INVALID message. ICP_OP_ERR should be used instead.

ICP_OP_QUERY

A query message. NOTE this opcode has a different payload format than most of the others. First is the requester's IPv4 address, followed by a URL. The Requester Host Address is not that of the cache generating the ICP message, but rather the address of the caches's client that originated the request. The Requester Host Address is often zero filled. An ICP message with an all-zero Requester Host Address address should be taken as one where the requester address is not specified; it does not indicate a valid IPv4 address.

ICP_OP_QUERY payload format:

```
  0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                      Requester Host Address                   |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                                                               |
/                        Null-Terminated URL                    /
/                                                               /
|                                                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

In response to an ICP_OP_QUERY, the recipient must return one of: ICP_OP_HIT, ICP_OP_MISS, ICP_OP_ERR, ICP_OP_MISS_NOFETCH, ICP_OP_DENIED, or ICP_OP_HIT_OBJ.

ICP_OP_SECHO

Similar to ICP_OP_QUERY, but for use in simulating a query to an origin server. When ICP is used to select the closest neighbor, the origin server can be included in the algorithm by bouncing an ICP_OP_SECHO message off it's echo port. The payload is simply the null-terminated URL.

NOTE: the echo server will not interpret the data (i.e. we could send it anything). This opcode is used to tell the difference between a legitimate query or response, random garbage, and an echo response.

ICP_OP_DECHO

Similar to ICP_OP_QUERY, but for use in simulating a query to a cache which does not use ICP. When ICP is used to choose the closest neighbor, a non-ICP cache can be included in the algorithm by bouncing an ICP_OP_DECHO message off it's echo port. The payload is simply the null-terminated URL.

NOTE: one problem with this approach is that while a system's echo port may be functioning perfectly, the cache software may not be running at all.

One of the following six ICP opcodes are sent in response to an ICP_OP_QUERY message. Unless otherwise noted, the payload must be the null-terminated URL string. Both the URL string and the Request Number field must be exactly the same as from the ICP_OP_QUERY message.

ICP_OP_HIT

An ICP_OP_HIT response indicates that the requested URL exists in this cache and that the requester is allowed to retrieve it.

ICP_OP_MISS

An ICP_OP_MISS response indicates that the requested URL does not exist in this cache. The querying cache may still choose to fetch the URL from the replying cache.

ICP_OP_ERR

An ICP_OP_ERR response indicates some kind of error in parsing or handling the query message (e.g. invalid URL).

ICP_OP_MISS_NOFETCH

An ICP_OP_MISS_NOFETCH response indicates that this cache is up, but is in a state where it does not want to handle cache misses. An example of such a state is during a startup phase where a cache might be rebuilding its object store. A cache in such a mode may wish to return ICP_OP_HIT for cache hits, but not ICP_OP_MISS for misses. ICP_OP_MISS_NOFETCH essentially means "I am up and running, but please don't fetch this URL from me now."

Note, ICP_OP_MISS_NOFETCH has a different meaning than ICP_OP_MISS. The ICP_OP_MISS reply is an invitation to fetch the URL from the replying cache (if their relationship allows it), but ICP_OP_MISS_NOFETCH is a request to NOT fetch the URL from the replying cache.

ICP_OP_DENIED

An ICP_OP_DENIED response indicates that the querying site is not allowed to retrieve the named object from this cache. Caches and proxies may implement complex access controls. This reply must be be interpreted to mean "you are not allowed to request this particular URL from me at this particular time."

Caches receiving a high percentage of ICP_OP_DENIED replies are probably misconfigured. Caches should track percentage of all replies which are ICP_OP_DENIED and disable a neighbor which exceeds a certain threshold (e.g. 95% of 100 or more queries).

Similarly, a cache should track the percent of ICP_OP_DENIED messages that are sent to a given address. If the percent of denied messages exceeds a certain threshold (e.g. 95% of 100 or more), the cache may choose to ignore all subsequent ICP_OP_QUERY messages from that address until some sort of administrative intervention occurs.

ICP_OP_HIT_OBJ

Just like an ICP_OP_HIT response, but the actual object data has been included in this reply message. Many requested objects are small enough that it is possible to include them in the query response and avoid the need to make a subsequent HTTP request for the object.

CAVEAT: ICP_OP_HIT_OBJ has some negative side effects which make its use undesirable. It transfers object data without HTTP and therefore bypasses the standard HTTP processing, including authorization and age validation. Another negative side effect is that ICP_OP_HIT_OBJ messages will often be much larger than the path MTU, thereby causing fragmentation to occur on the UDP packet. For these reasons, use of ICP_OP_HIT_OBJ is NOT recommended.

A cache must not send an ICP_OP_HIT_OBJ unless the ICP_FLAG_HIT_OBJ flag is set in the query message Options field.

ICP_OP_HIT_OBJ payload format:

```
  0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                                                               |
/                       Null-Terminated URL                     /
/                                                               /
|                                                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|         Object Size           |                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+                               |
|                                                               |
/                           Object Data                         /
/                                                               /
|                                                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

The receiving application must check to make sure it actually receives Object Size octets of data. If it does not, then it should treat the ICP_OP_HIT_OBJ reply as though it were a normal ICP_OP_HIT.

NOTE: the Object Size field does not necessarily begin on a 32-bit boundary as shown in the diagram above. It begins immediately following the NULL byte of the URL string.

UNRECOGNIZED OPCODES

ICP messages with unrecognized or unused opcodes should be ignored, i.e. no reply generated. The application may choose to note the anomalous behaviour in a log file.

3. ICP Option Flags

0x80000000 ICP_FLAG_HIT_OBJ

This flag is set in an ICP_OP_QUERY message indicating that it is okay to respond with an ICP_OP_HIT_OBJ message if the object data will fit in the reply.

0x40000000 ICP_FLAG_SRC_RTT

This flag is set in an ICP_OP_QUERY message indicating that the requester would like the ICP reply to include the responder's measured RTT to the origin server.

Upon receipt of an ICP_OP_QUERY with ICP_FLAG_SRC_RTT bit set, a cache should check an internal database of RTT measurements. If available, the RTT value MUST be expressed as a 16-bit integer, in units of milliseconds. If unavailable, the responder may either set the RTT value to zero, or clear the ICP_FLAG_SRC_RTT bit in the ICP reply. The ICP reply MUST not be delayed while waiting for the RTT measurement to occur.

This flag is set in an ICP reply message (ICP_OP_HIT, ICP_OP_MISS, ICP_OP_MISS_NOFETCH, or ICP_OP_HIT_OBJ) to indicate that the low 16-bits of the Option Data field contain the measured RTT to the host given in the requested URL. If ICP_FLAG_SRC_RTT is clear in the query then it MUST also be clear in the reply. If ICP_FLAG_SRC_RTT is set in the query, then it may or may not be set in the reply.

4. Security Considerations

The security issues relating to ICP are discussed in the companion document, RFC2187.

5. References

[1] Fielding, R., et. al, "Hypertext Transfer Protocol -- HTTP/1.1", RFC 2068, UC Irvine, January 1997.

[2] Berners-Lee, T., Masinter, L., and M. McCahill, "Uniform Resource Locators (URL)", RFC 1738, CERN, Xerox PARC, University of Minnesota, December 1994.

[3] Bowman M., Danzig P., Hardy D., Manber U., Schwartz M., and Wessels D., "The Harvest Information Discovery and Access System", Internet Research Task Force - Resource Discovery, http://harvest.transarc.com/.

[4] Wessels D., Claffy K., "ICP and the Squid Web Cache", National Laboratory for Applied Network Research, http://www.nlanr.net/~wessels/Papers/icp-squid.ps.gz

[5] Wessels D., "The Squid Internet Object Cache", National Laboratory for Applied Network Research, http://squid.nlanr.net/Squid/

6. Acknowledgments

The authors wish to thank Paul A Vixie <paul@vix.com> for providing excellent feedback on this document.

7. Authors' Addresses

Duane Wessels
National Laboratory for Applied Network Research
10100 Hopkins Drive
La Jolla, CA 92093

EMail: wessels@nlanr.net

K. Claffy
National Laboratory for Applied Network Research
10100 Hopkins Drive
La Jolla, CA 92093

EMail: kc@nlanr.net