

Service Specific Multicast Address Assignment
for use with the Service Location Protocol
<[draft-ietf-svrloc-service-mcast-00.txt](#)>

This document is an Internet-Draft. 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.''

To learn the current status of any Internet-Draft, please check the ``1id-abstracts.txt'' listing contained in the Internet-Drafts Shadow Directories on ftp.is.co.za (Africa), nic.nordu.net (Europe), munnari.oz.au (Pacific Rim), ds.internic.net (US East Coast), or ftp.isi.edu (US West Coast).

Abstract

One feature of the Service Location Protocol is that Services may be assigned a Service Specific Multicast address to aid in service discovery. Use of this address for service discovery will not burden systems which do not advertise the service type. Two open questions are answered in the following proposal: How are the multicast addresses to be assigned? How are these multicast addresses known by SLP Service and User Agents?

[1.0](#) Introduction

SLP User Agents can request information about services by multicasting requests to service specific multicast addresses. Only Service Agents which have service information for these services will receive the request. Other Service Agents will never receive them, as the multicast request will be filtered out by the link layer.

There are two unanswered questions which must be answered in order to successfully make use of Service Specific Multicast addresses. The first is how will new multicast addresses be assigned. The second is how will new assignments be known and by UAs and SAs.

2.0 String Hashing of Service Types to Multicast Addresses

A range of (for example) 1024 contiguous multicast addresses will be requested from IANA for use with service location. The service specific multicast address is the string hash of the service's "service type" string identifier. For example "nfs" would be hashed, then the last 10 significant bits of the hash would be used to determine the multicast address within the range.

Service Specific Multicast addresses do not need to be unique in order to server their purpose. If spurious requests arrive at an SA due to hashing collisions, the SA can always discard the request, as it would do anyway if it received a request on the Service Location General Multicast address. What will occur is that SAs representing very few services will not receive nearly as many spurious requests as if all requests were made to the Service Location General Multicast address.

This solves the assignment problem. All service types, whether they are standardized or not, can be hashed into a particular multicast address. Discovery of service types will now effectively also discover Service Specific Multicast addresses. Finally, the effort of standardizing new service types will be disassociated from that of Multicast Address assignments. This will reduce the necessary interaction with the IANA required to standardize new service types. It will also make it possible for nonstandard service types to make use of Service Specific Multicast addresses. This simplification changes [\[SRV\]](#) and [\[SLP\]](#)=20 which specify that these addresses are directly assigned by IANA. =20

The string hash function is attributed to Chris Torek:=20

```
unsigned long SLPhash(const char *pc) {
    unsigned long h =3D 0;
    for (; *pc; pc++) {
        h *=3D 33;
        h +=3D *pc;
    }
    return (0x2FF & h); /* round to a range of 0-1023 */
}
```

The string which should be passed into this function is the service type string, as a null terminated ASCII string. The service type will always be expressable as a string even if the service URL is in another character set. That is because the grammar definition of a URL is such that all allowable characters in a URL are in a very restricted set of characters; all of which fall within the ASCII range.

3.0 Security Considerations

There are no security ramifications to mapping service type strings to multicast addresses.

4.0 Internationalization Considerations

The service type string must be expressed in subset of the ASCII character set. This is a restriction posed by the URL encoding standard [[URL](#)]. Unfortunately, this limits the representation of service type names, which will make it difficult to name services in languages which use characters outside of this restricted set.

5.0 Acknowledgments

This internet draft spells out a proposal made by Thomas Narten last year to provide a simple and powerful solution to both multicast address assignment and disseminating knowledge of those assignments.

6.0 References

- [SLP] J. Veizades, E. Guttman, C. Perkins & S. Kaplan, "Service Location Protocol", Work in progress, January 1997.
- [SRV] E. Guttman, "The service: URL Scheme", Work in progress, November 1996.
- [URL] T. Berners-Lee, L. Masinter & M. McCahill, "Uniform Resource Locators (URL)", [RFC 1738](#). December 1994.

7.0 Author's Address

Erik Guttman

Sun Microsystems, Inc.
Gaisbergstr. 6
D-69115 Heidelberg
Germany

Phone: +49 6221 601649
Fax : +49 6221 161019
email: Erik.Guttman@eng.sun.com

This memo expires on August 12, 1997

