Internet Engineering Task Force
INTERNET DRAFT

C. Perkins
Sun Microsystems
13 March 1998

DHCP Options for Service Location Protocol
draft-ietf-dhc-slp-03.txt

Status of This Memo

Abstract

The Dynamic Host Configuration Protocol provides a framework for passing configuration information to hosts on a TCP/IP network. Entities using the Service Location Protocol need to find out the address of Directory Agents in order to transact messages. Another option provides an assignment of scope for configuration of SLP User and Service Agents.

1. Introduction

The Dynamic Host Configuration Protocol [4] provides a framework for passing configuration information to hosts on a TCP/IP network. Entities using the Service Location Protocol [7] need to find out the address of Directory Agents in order to transact messages and obtain the correct scope to be used in messages which are exchanged using the Service Location Protocol.

The scope MUST be encoded using the UTF8 character encoding [8] and have the values referred by the MIBEnum value. Note that each character may require two or more octets of data for its representation.

Note that each option listed below MAY be included multiple times in the same DHCPOFFER or DHCPREQUEST. If so, then the options SHOULD be included in order of decreasing preference.

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 [2].

2. Typed Scope Lists

In Service Location Protocol, multiple service types can be hosted on the same network node. However, DHCP typically configures computers based on their IP address. It is possible that different service types on the same computer would be administered from different scopes. Thus, options 78 and 79 have additional syntax to allow this more detailed style of service configuration.

In particular, the list of scopes contained in the options is syntactically separated into lists pertaining to each service type.

Grammatically, a typed-scope-list in a DHCPOFFER is structured as follows:

```
typed-scope-list = one or more maybe-typed-scope-items,
```

```
                        separated by commas
    maybe-typed-scope-item = typed-scope-item, or scope-list
    typed-scope-item = '(' service-type '=' scope-list ')'
    scope-list = one or more scope-items, comma-separated
```

A typed-scope-list in a DHCPREQUEST is structured as follows:

```
    typed-scope-list = one or more maybe-typed-scope-items,
                        separated by commas
    maybe-typed-scope-item = typed-scope-item, or
```



```
                                maybe-empty-scope-list
    typed-scope-item = '(' service-type '=' maybe-empty-scope-list ')'
    maybe-empty-scope-list = zero or more scope-items, comma-separated
```

A service type has the format defined in [5], and a scope-item has the format defined in [6] for "strval". Basically, a scope-item is a character string that has alphanumeric characters not including control characters or `(',`)',`,', \',`!',`<',`=',`>', or `~' Service schemes are special cases of schemes as defined for general URLs [1].

The typed-scope-list MAY contain both untyped-scope-lists and typed-scope-lists. Each scope-item in each untyped-scope-list applies to every service type on the node.

As an example, the scope-list ``A,B,C'' denotes scopes A, B and C for all service types on the client. In a DHCPREQUEST, this scope string would indicate that the client wishes a directory agent which supports ANY of these three scopes. In a DHCPOFFER, the scope indicates that the directory agent supports ALL of the three scopes.

Suppose instead that service types "netman" and "proxystuff" are residing on a DHCP client. Then, the typed-scope-list in a DHCPOFFER could be,

```
    (netman=mgmt),(proxystuff=math-dept,labs)
```

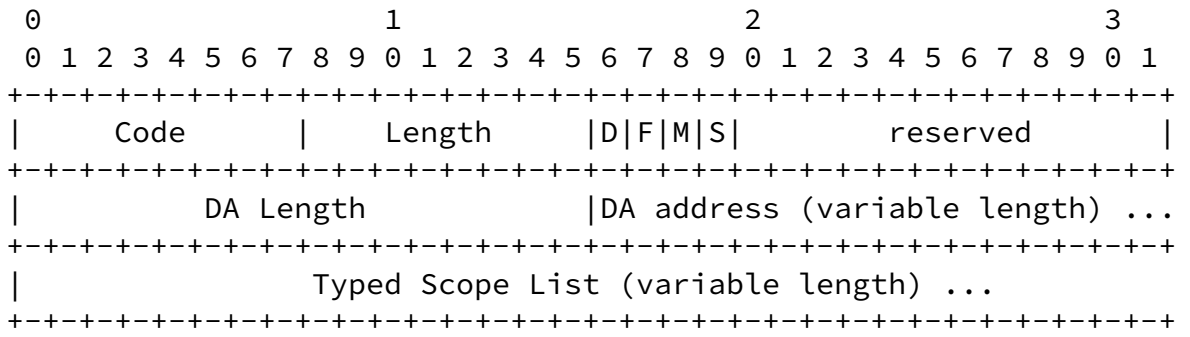
Assuming the DHCP client with two service types "netman" and "proxystuff" did not make any scope restriction, a corresponding typed-scope-list in a DHCPREQUEST could be,

```
    (netman=),(proxystuff=)
```

asking for scopes for those service types.

3. Directory Agent Option

This option requests or specifies a Directory Agent (DA), along with zero or more scopes supported by that directory agent.



Code 78

Length (variable) The length of the option in bytes.

D If the 'D' bit is set, the Directory Agent field and the DA Length fields are present.

F If the 'F' bit is set, the Directory Agent is indicated by including its variable length host name or Fully Qualified Domain Name (FQDN) instead of its IP address.

M If the 'M' bit is set, the Directory Agent address is the only one that may be used, and multicast methods for discovering Directory Agents MUST NOT be used.

S If the 'S' bit is set, the scope is present.

rsv reserved; ignored upon reception; MUST be sent as zero

DA Length The length (in octets) of the Directory Agent field.

Directory Agent
The Fully Qualified Domain Name (FQDN), host name, or IP

address of the Directory Agent.

Typed Scope List
The characters denoting the scope (see Section reftsl).

In order to simplify administration of the configuration of Directory Agents for Service Location Protocol clients, the Directory Agent can be indicated by presenting its FQDN or host name instead of its IP address. This allows renumbering to proceed more smoothly [3]. When the FQDN or host name is used, the server sets the 'F' bit. The host name can be distinguished from the FQDN by the presence of a '.' character. In any case, the DA length field is set to be the length of the Directory Agent field. When the 'F' bit is not set, the DA Length MUST be 4.

Note that more than one Directory Agent option may be present in a DHCP message. Each such option may have the same or different scope.

The client may request any Directory Agent with a particular scope, by including the Directory Agent option in a DHCP Request message with no Directory Agent address included (the 'D' bit set to zero), and the characters denoting the scope.

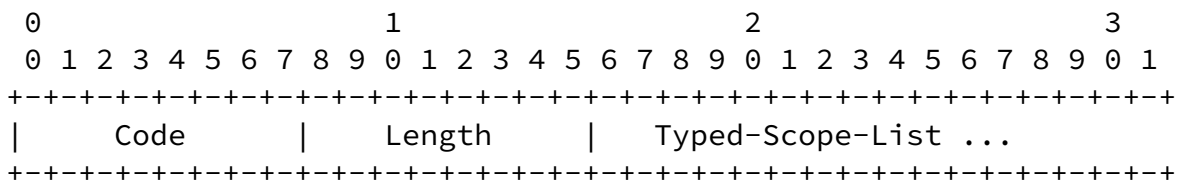
The length of the Typed Scope List is only indicated implicitly by the overall length of the option. This string is NOT null terminated.

The format of the Typed Scope List field is described in section 2.

Option 78 MUST include one or more scopes if a DA address is returned. Using option 78, it is not possible for different service types on the same node to be configured with different directory agents. In other words, all service types on the same node will be configured with the same directory agent.

4. Service Scope Option

This option indicates one or more that should be used by a Service Agent (SA) [7], when responding to Service Request messages as specified by the Service Location Protocol.



Code 79

Length (variable) The length of the option in bytes.

scope the characters denoting the scope.

The Typed-Scope-List is described in Section 2. The DHCP client (i.e., user agent or service agent) which receives this option will use the indicated scope for in all SLP requests and registrations. The scope string must be UTF8 character encoded. This string is not null terminated.

DHCP clients MAY use Option 79 to request scopes for one or more particular service types.

5. Security Considerations

If a malicious host is able to insert fraudulent information in DHCPOFFER packets sent to a prospective client of the Service Location Protocol, then the client will be unable to obtain service, or may unwittingly be directed to use the incorrect services.

Many opportunities for denial of service exist. A service agent could find that it might rely on fraudulent or otherwise malicious directory agents to advertise its services. DHCPOFFERs could prevent the regular SLP framework from functioning by directing clients to not use multicast, to use nonexistent directory agents and so on.

These difficulties are inherited from the much larger and more serious problem, viz. securing or authenticating any information whatsoever from a DHCP server (or client!) is not possible in common DHCP deployments.

6. Acknowledgements

Thanks to Erik Guttman for his helpful suggestions in the creation ane revision of this draft.

References

[1] T. Berners-Lee, L. Masinter, and M. McCahill. Uniform Resource Locators (URL). RFC 1738, December 1994.

[2] S. Bradner. Key words for use in RFCs to Indicate Requirement Levels. RFC 2119, March 1997.

[3] B. Carpenter and Y. Rekhter. Renumbering needs work. RFC 1900, February 1996.

[4] R. Droms. Dynamic Host Configuration Protocol. RFC 2131, March 1997.

[5] E. Guttman, C. Perkins, and J. Kempf. Service Templates and service: Schemes. draft-ietf-svrloc-service-scheme-05.txt, November 1997. (work in progress).

[6] E. Guttman, C. Perkins, J. Veizades, and M. Day. Service Location Protocol version 2. March 1998. (work in progress). draft-ietf-svrloc-protocol-v2-04.txt,

[7] J. Veizades, E. Guttman, C. Perkins, and S. Kaplan. Service Location Protocol. RFC 2165, July 1997.

[8] F. Yergeau. UTF-8, a transformation format of unicode and ISO 10646. RFC 2279, January 1998.

Author's Address

Questions about this memo can be directed to:

Charles E. Perkins
Technology Development Group
Mail Stop MPK15-214
Room 2682
Sun Microsystems, Inc.
15 Network Circle
Menlo Park, CA 94025
ph# 1-650-786-6464
fax# 1-650-786-6445
email: charles.perkins@Sun.COM
charles.perkins@Eng.sun.com
cperkins@Eng.sun.com
Web: http://www.svrloc.org/~charliep