Network Working Group
Request for Comments: 3397
Category: Standards Track

B. Aboba
Microsoft
S. Cheshire
Apple Computer, Inc.

November 2002

Dynamic Host Configuration Protocol (DHCP) Domain Search Option

Status of this Memo

Copyright Notice

Abstract

This document defines a new Dynamic Host Configuration Protocol (DHCP) option which is passed from the DHCP Server to the DHCP Client to specify the domain search list used when resolving hostnames using DNS.

Table of Contents

1. Introduction

The Dynamic Host Configuration Protocol (DHCP) [RFC2131] provides a mechanism for host configuration. [RFC2132] and [RFC2937] allow DHCP servers to pass name service configuration information to DHCP clients. In some circumstances, it is useful for the DHCP client to be configured with the domain search list. This document defines a new DHCP option which is passed from the DHCP Server to the DHCP Client to specify the domain search list used when resolving hostnames with DNS. This option applies only to DNS and does not apply to other name resolution mechanisms.

1.1. Terminology

This document uses the following terms:

DHCP client
: A DHCP client or "client" is an Internet host using DHCP to obtain configuration parameters such as a network address.

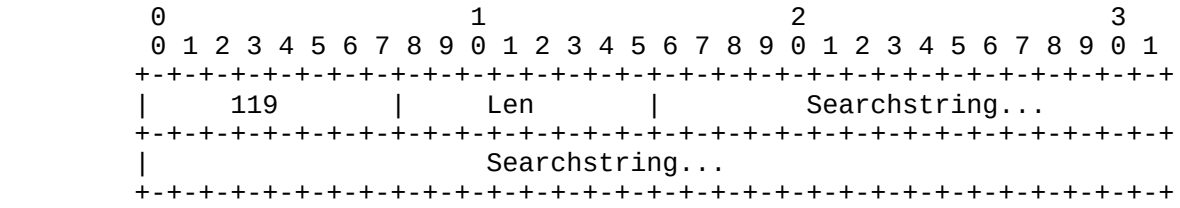
DHCP server
: A DHCP server or "server" is an Internet host that returns configuration parameters to DHCP clients.

1.2. Requirements Language

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 "Key words for use in RFCs to Indicate Requirement Levels" [RFC2119].

2. Domain Search Option Format

The code for this option is 119.



In the above diagram, Searchstring is a string specifying the searchlist. If the length of the searchlist exceeds the maximum permissible within a single option (255 octets), then multiple options MAY be used, as described in "Encoding Long Options in the Dynamic Host Configuration Protocol (DHCPv4)" [RFC3396].

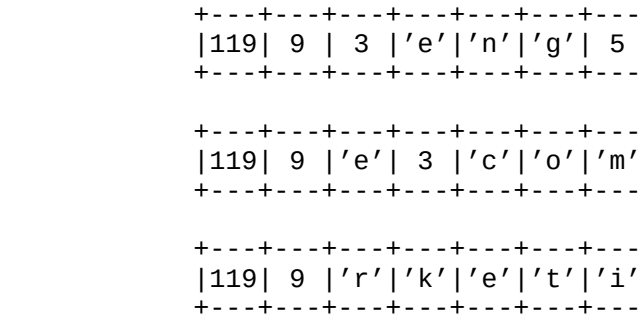
To enable the searchlist to be encoded compactly, searchstrings in the searchlist MUST be concatenated and encoded using the technique described in section 4.1.4 of "Domain Names - Implementation And Specification" [RFC1035]. In this scheme, an entire domain name or a list of labels at the end of a domain name is replaced with a pointer to a prior occurrence of the same name. Despite its complexity, this technique is valuable since the space available for encoding DHCP options is limited, and it is likely that a domain searchstring will contain repeated instances of the same domain name. Thus the DNS name compression is both useful and likely to be effective.

For use in this specification, the pointer refers to the offset within the data portion of the DHCP option (not including the preceding DHCP option code byte or DHCP option length byte).

If multiple Domain Search Options are present, then the data portions of all the Domain Search Options are concatenated together as specified in "Encoding Long DHCP Options in the Dynamic Host Configuration Protocol (DHCPv4)" [RFC3396] and the pointer indicates an offset within the complete aggregate block of data.

3. Example

Below is an example encoding of a search list consisting of "eng.apple.com." and "marketing.apple.com.":



Note:

i. The encoding has been split (for this example) into three Domain Search Options. All Domain Search Options are logically concatenated into one block of data before being interpreted by the client.

ii. The encoding of "eng.apple.com." ends with a zero, the null root label, to mark the end of the name, as required by RFC 1035.

iii. The encoding of "marketing" (for "marketing.apple.com.") ends with the two-octet compression pointer C004 (hex), which points to offset 4 in the complete aggregated block of Domain Search Option data, where another validly encoded domain name can be found to complete the name ("apple.com.").

Every search domain name must end either with a zero or with a two-octet compression pointer. If the receiver is part-way through decoding a search domain name when it reaches the end of the complete aggregated block of the searchlist option data, without finding a zero or a valid two-octet compression pointer, then the partially read name MUST be discarded as invalid.

4. Security Considerations

Potential attacks on DHCP are discussed in section 7 of the DHCP protocol specification [RFC2131], as well as in the DHCP authentication specification [RFC3118]. In particular, using the domain search option, a rogue DHCP server might be able to redirect traffic to another site.

For example, a user requesting a connection to "myhost", expecting to reach "myhost.bigco.com" might instead be directed to "myhost.roguedomain.com". Note that support for DNSSEC [RFC2535] will not avert this attack, since the resource records for "myhost.roguedomain.com" might be legitimately signed. This makes the domain search option a more fruitful avenue of attack for a rogue DHCP server than providing an illegitimate DNS server option (described in [RFC2132]).

The degree to which a host is vulnerable to attack via an invalid domain search option is determined in part by DNS resolver behavior. [RFC1535] discusses security weaknesses related to implicit as well as explicit domain searchlists, and provides recommendations relating to resolver searchlist processing. [RFC1536] section 6 also addresses this vulnerability, and recommends that resolvers:

[1] Use searchlists only when explicitly specified; no implicit searchlists should be used.

[2] Resolve a name that contains any dots by first trying it as an FQDN and if that fails, with the local domain name (or searchlist if specified) appended.

[3] Resolve a name containing no dots by appending with the searchlist right away, but once again, no implicit searchlists should be used.

In order to minimize potential vulnerabilities it is recommended that:

[a] Hosts implementing the domain search option SHOULD also implement the searchlist recommendations of [RFC1536], section 6.

[b] Where DNS parameters such as the domain searchlist or DNS servers have been manually configured, these parameters SHOULD NOT be overridden by DHCP.

[c] Domain search option implementations MAY require DHCP authentication [RFC3118] prior to accepting a domain search option.

5. Normative References

[RFC1035] Mockapetris, P., "Domain Names - Implementation and Specification", STD 13, RFC 1035, November 1987.

[RFC1536] Kumar, A., Postel, J., Neuman, C., Danzig, P. and S. Miller, "Common DNS Implementation Errors and Suggested Fixes", RFC 1536, October 1993.

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.

[RFC2131] Droms, R., "Dynamic Host Configuration Protocol", RFC 2131, March 1997.

[RFC3118] Droms, R. and W. Arbaugh, "Authentication for DHCP Messages", RFC 3118, June 2001.

[RFC3396] Lemon, T. and S. Cheshire, "Encoding Long Options in the Dynamic Host Configuration Protocol (DHCPv4)", RFC 3396, November 2002.

6. Informative References

[RFC1535] Gavron, E., "A Security Problem and Proposed Correction With Widely Deployed DNS Software", RFC 1535, October 1993.

[RFC2132] Alexander, S. and R. Droms, "DHCP Options and BOOTP Vendor Extensions", RFC 2132, March 1997.

[RFC2535] Eastlake, D., "Domain Name System Security Extensions", RFC 2535, March 1999.

[RFC2937] Smith, C., "The Name Service Search Option for DHCP", RFC 2937, September 2000.

7. IANA Considerations

The IANA has assigned DHCP option code 119 to the Domain Search Option.

8. Acknowledgments

The authors would like to thank Michael Patton, Erik Guttman, Olafur Gudmundsson, Thomas Narten, Mark Andrews, Erik Nordmark, Myron Hattig, Keith Moore, and Bill Manning for comments on this memo.

9. Intellectual Property Statement

10. Authors' Addresses

Bernard Aboba
Microsoft Corporation
One Microsoft Way
Redmond, WA 98052

Phone: +1 425 706 6605
EMail: bernarda@microsoft.com

Stuart Cheshire
Apple Computer, Inc.
1 Infinite Loop
Cupertino
California 95014
USA

Phone: +1 408 974 3207
EMail: rfc@stuartcheshire.org

11. Full Copyright Statement

Acknowledgement

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