Malloc Working Group   Roger Kermode
Internet Engineering Task Force   Motorola
INTERNET-DRAFT
19 April 2000
Expires 27 December 2000

MADCAP Multicast Scope Nesting State Option
<draft-ietf-malloc-madcap-nest-opt-05.txt>

Status of this Memo

Abstract

    This document defines a new option to the Multicast Address Dynamic
    Client Allocation Protocol (MADCAP) to support nested scoping. The
    new option's purpose is to allow clients to learn which scopes nest
    inside each other, and hence may be used for expanding scope searches
    or hierarchical multicast transport.

Copyright Notice

Contents

1. Introduction

The Multicast Address Dynamic Client Allocation Protocol (MADCAP)
[RFC2730] affords client applications the ability to request
multicast address allocation services from multicast address
allocation servers. As part of the Multicast Address Allocation
Architecture [MAAA], MADCAP gives clients the ability to reserve,
request, and extend leases on multicast addresses.

A new MADCAP option, the "Multicast Scope Nesting State" option is
proposed to allow clients to learn not only which scopes exist via
the existing "Multicast Scope List" option, but how these scopes nest
inside each other. This new option will also afford clients the
ability to make better scope selections for a given session and also
to construct hierarchies of administratively scoped zones. These
hierarchies may then be used to perform expanding scope searches
instead of the expanding ring or increasing-TTL searches. Expanding
scope searches do not suffer from the Split-Horizon Effect present in
expanding ring searches, and therefore both simplify protocol design
and provide better localization.

1.1 Time-To-Live (TTL) Scoping Split Horizon Effect

Multicast searching and localized recovery transport techniques that
rely on TTL scoping are known to suffer when deployed in a wide scale
manner. The failing lies in the split horizon effect shown below in
Figure 1. Here a requestor and responder must each use a TTL that is
sufficiently large that they will reach the other. When they are
separated by many hops the TTL becomes large and the number of
receivers within the multicast tree that only receive either the
request or the response can become very large.

```
            .......      *******
         ...        ***       ***        A Only hears S
       ..        **    ..        **       B hears S and R
      .        *        .          *      C Only hears R
     .        *           .          *
     .       S<------->R          *      . TTL Boundary for S
     .        *          .          *     * TTL Boundary for R
     .    A    *   B   .   C     *
      ..        **    ..        **
       ...        ***       ***
          .......      *******
```
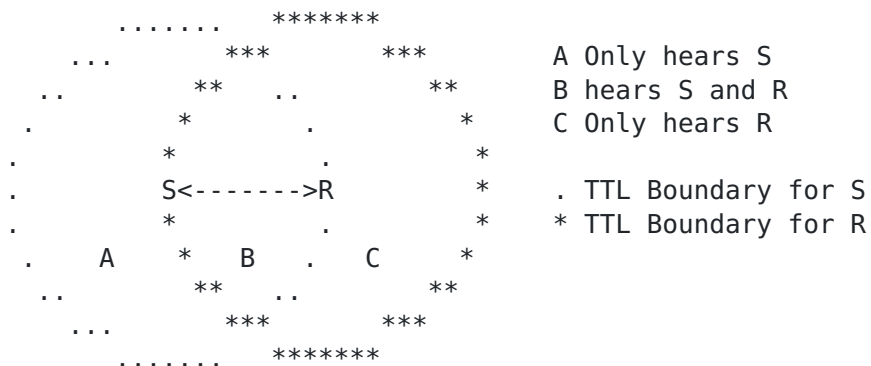
            Figure 1 : Split Horizon Problem from TTL scoping

1.2 Eliminating the Split Horizon Effect with Administrative Scoping

   Ideally, a mechanism that either eliminates or minimizes the size of
   the A and C regions in Figure 1. as shown in Figure 2. is needed to
   solve this problem. One mechanism that affords this ability is
   administrative scoping [RFC2365], in which routers prevent the
   passing of packets within a certain range of multicast addresses.
   Routers that have this feature can be configured to provide a
   perimeter around a region of the network. This perimeter is said to
   encompass an administratively scoped zone inside of which traffic
   sent to that particular range of multicast addresses can neither
   leave nor enter. Routers can construct and manage administratively
   scoped zones using the MZAP [RFC2776] protocol.

```
                   ........................
               .                          .
               .           many hops       .
              .S<----------------------->R.
               .                          .
               .                          .
                   ........................
```
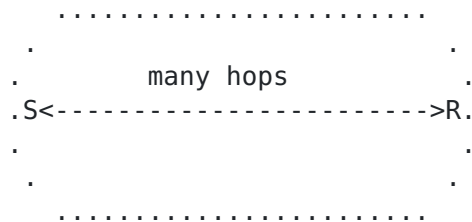
          Figure 2 : Eliminating the Split Horizon Effect

   MZAP also includes the ability to determine whether or not
   administratively scoped regions nest inside one another. This allows
   hierarchies such as that shown in Figure 1. to be constructed.

```
      . . . . .  . . . . . . . . . . . . .
     .               scope a               .      Scope Boundaries
    .                                        .      . = scope  a
   . _____    _____    .      - = scopes b,c
   . /    scope b    \   /   scope c     \ .      # = scopes d,e,f, & g
   .|                 | |                 |.
   .|  #####    ##### | |  #####    ##### |.
   .| #scope#  #scope#| | #scope#  #scope# |.
   .\ # d #  # e   #| | # f  #  #  g # /.
    .\ ####     #####/    \ #####    #### /.
     ._____/       _____/.
      . . . . . . . . . . . . . . . . .
```
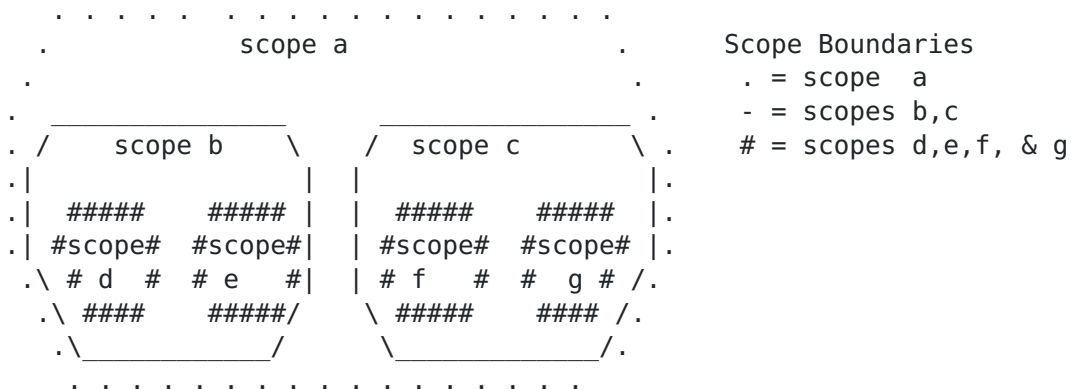
          Figure 3 : Admin Scope Zone Nesting Hierarchy example

   A generic expanding scope search algorithm [KERM] that exploits the
   existence of a hierarchy of administratively scoped zones is:

1) Starting with the smallest known scope for the session, a
   requestor in that session issues a request and waits for a reply.

2) If a node within that scope hears a request at a certain scope
   that it can satisfy it sends a response at that same scope,
   possibly after some random delay to reduce duplicate responses.

3) Nodes that receive a response to a particular request while
   waiting to send a response to that request, suppress their own
   response.

4) If a requestor issues a request to a scope, and does not hear a
   response after a specified amount of time, it retransmits its
   request at the same scope a small number of additional times.
   Should these retries fail to elicit a response, the requestor
   increases the scope to the next largest scope and tries
   again.

5) Requestors increase the scope of the request according to
   step 4 until either a response is received, or the largest
   legal scope for the session is reached. Should attempts to
   elicit a response at the largest possible scope for the
   session fail to yield a response, the requestor may conclude
   that the request cannot be met.

1.3. 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 RFC 2119 [RFC2119].

Throughout the rest of this document, the words "server" or "MADCAP
server" refer to a host providing multicast address allocation
services via MADCAP. The words "client" or "MADCAP client" refer to a
host requesting multicast address allocation services via MADCAP.

2. Multicast Nested Scoping State

Two scopes, X and Y, can be related in one of four possible ways.
 1) X nests inside Y,
 2) Y nests inside X,
 3) X and Y do not nest (the overlap case), and
 4) X and Y nest inside each other.

The fourth case SHOULD be interpreted as meaning that X and Y have
exactly the same border. This does not mean that X and Y are the same
scope since X and Y may correspond to different ranges of the
multicast address space.

This state MUST be stored in the MADCAP servers which MUST allow the
state to be updated as network conditions change. Each MADCAP server
SHOULD therefore define two pieces of state that describe whether
"scope X nests in scope Y" or vice versa. For the remainder of this
document the nesting relationship shall be denoted as the "/" where
X/Y defines the relation "X nests inside Y". This relation shall be
understood to take one of the values "true", or "false." Nesting
relationship state that is indeterminate is considered to be "false."

3 Multicast Scope Nesting State Option

The "Multicast Scope Nesting State" option is proposed to augment the
"Multicast Scope List" option within the MADCAP protocol by providing
additional information to applications about how scopes nest. The
proposed option is OPTIONAL, that is MADCAP servers MAY implement
this new option, however they are not required to.

MADCAP servers shall learn this additional nesting information by
means of static configuration or via some other protocol such as MZAP
[RFC2776] that manages administrative scopes in a dynamic fashion.

3.1 Multicast Scope List Option

To understand the "Multicast Scope Nesting State" option one must
first understand the "Multicast Scope List" option.

The Multicast Scope List option in MADCAP is used by MADCAP servers
to inform MADCAP clients of which zones are visible. Visible scopes
are enumerated inside the option as successive tuples, where each
tuple consists of the following information:

   o Scope ID:
      The smallest address for the range of multicast addresses
      covered by this scope.

   o Last Address:
      The largest address for the range of multicast addresses
      covered by this scope.

   o TTL:
      The TTL to be used when sending messages to this scope.

   o Name(s):
      One or more language specific names for the scope.

3.2 Representing the Multicast Scope Nesting State

   Given a Multicast Scope List containing descriptions for n scopes one
   can form n(n-1)/2 pairings. As was shown in section 2 each pairing
   can take on one of four possible states. Thus, for a list of n scopes
   there exists 2 pieces of information for each pairing for a total of
   n(n-1) pieces of information regarding which scopes do and do not
   nest inside each other.

   There are several ways to represent this state using full matrices,
   sparse-matrices, and using lists of variable length lists. In the
   interests of maximal efficiency and flexibility, the Multicast
   Nesting State Option uses a bit-packed matrix approach. In this
   approach a matrix is constructed using pieces of X/Y state where X is
   the row and Y is the column. A "1" in the matrix means that the
   relationship "row nests inside column" is true, while a "0" means
   that this relationship is either false or indeterminate. The
   diagonal of the matrix is removed, since this is the case where X is
   the same as Y, and each row is then zero-padded to the next byte
   boundary to give the final representation.

   An example of how a matrix would be constructed for the following
   scope nestings S1/S2, S2/S3, S2/S4, S3/S5, S4/S5, S5/S6, and S6/S7.
   Note that a number of additional nesting relationships are implied
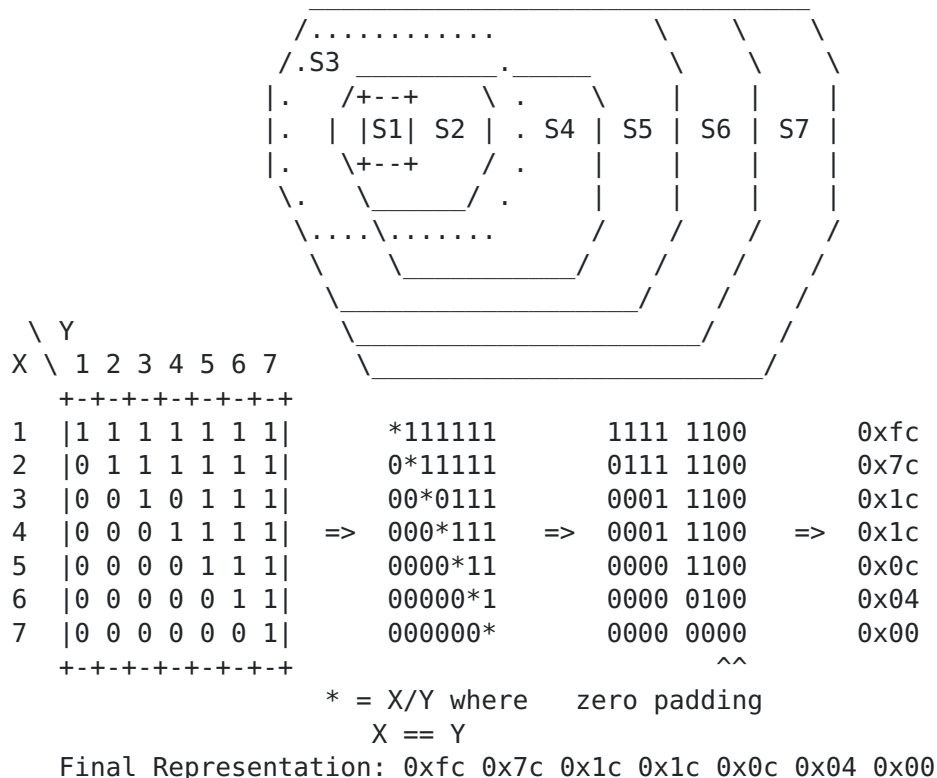   from this set.

```
                   _____
           /...........          \     \     \
          /.S3 _____._____     \     \      \
          |.    /+--+    \ .     \    |     |     |
          |.   | |S1| S2 | . S4 | S5 | S6 | S7 |
          |.    \+--+    /  .    |     |     |     |
          \.     _____/ .     |     |     |     |
           \....\.......         /     /     /     /
            \     _____/     /     /     /
             _____/     /     /
 \ Y          _____/     /
 X \ 1 2 3 4 5 6 7    _____/
    +-+-+-+-+-+-+-+
  1 |1 1 1 1 1 1 1|      *111111       1111 1100      0xfc
  2 |0 1 1 1 1 1 1|      0*11111       0111 1100      0x7c
  3 |0 0 1 0 1 1 1|      00*0111       0001 1100      0x1c
  4 |0 0 0 1 1 1 1|  =>  000*111   =>  0001 1100   => 0x1c
  5 |0 0 0 0 1 1 1|      0000*11       0000 1100      0x0c
  6 |0 0 0 0 0 1 1|      00000*1       0000 0100      0x04
  7 |0 0 0 0 0 0 1|      000000*       0000 0000      0x00
    +-+-+-+-+-+-+-+                          ^^
                 * = X/Y where   zero padding
                   X == Y
      Final Representation: 0xfc 0x7c 0x1c 0x1c 0x0c 0x04 0x00

      Figure 4. Scope Nesting Example
```

3.3 Multicast Scope Nesting State Option Usage

The "Multicast Scope Nesting State" option is dependent upon the
"Multicast Scope List" option. This decision was made according to
the following reasoning. The Multicast Nest State Option requires
that the scopes be identified along with their nesting properties.
Since the information needed to describe a scope is contained in the
Multicast Scope List option and this information can change, the
MADCAP messages that contain the Multicast Scope Nesting State option
must be atomic and therefore must include the "Multicast Scope List
Option."

Thus, the "Multicast Scope Nesting State" option MUST only be used in
messages that carry the "Multicast Scope List" option, specifically:

     ACK (in response to GETINFO)

Since the Multicast Nest State option is dependent upon the Multicast
Scope List option, it MUST NOT be included without the Multicast
Scope List option.

Clients that need to explicitly learn the nesting relationships
between scopes should therefore send a GETINFO message to the server
with the "Multicast Scope List" AND "Multicast Scope Nesting State"
option codes listed in an Option Request option.

4. Managing Dynamically Nested Scopes

Scopes can either be manually or automatically configured. When
scopes are manually configured the relationships between them will
also be static, assuming that network does not partition due to
router failure. Should the network partition or heal after a
partition it is highly likely that the nesting relationships will
change. Scope nesting relationships will also change as a network is
brought up or when a change is deliberately made to a router either
through manual reconfiguration or by some automatic means.

To ensure that nesting relationships are correctly determined when
scope boundaries undergo change MADCAP servers MUST include a
mechanism that allow for:

 a) whether the nesting decision is still under consideration or
    can be considered definitive, and therefore be announced to
    MADCAP clients.

 b) whether one or both scopes for a particular nesting state entry
    have been destroyed, and hence whether the nesting state should
    therefore be discarded.

 c) whether the scope boundaries have changed so that whereas scope
    X did or did not nest inside scope Y, the opposite is now true.

To realize a) and b) MADCAP servers MUST implement the following two
timers; NEST_NO_DECISION_TIMER, ZONES_EXIST_TIMER.

The first timer, NEST_NO_DECISION_TIMER, is used to mark time between
a MADCAP server's first hearing of a scope and making a decision
about its relationship to other zones. Up until the time this timer
expires MADCAP servers MUST NOT conclude that the scope nests within
another.

The NEST_NO_DECISION_TIMER timer will also be used to timeout X/Y =
"false" state to allow X/Y to be reset to true in the event that the
boundaries for zone X and zone Y change so that zone X now nests
inside zone Y.

The second timer ZONES_EXIST_TIMER will be used to timeout the
internal state between two scopes in the event that one or both
scopes are destroyed.

4.1 MADCAP Server processing of MZAP messages

When MZAP is used to discover the nesting relationship between scopes
MADCAP servers will eavesdrop into the MZAP messages that are
periodically transmitted by the Zone Border Routers (ZBR) during the
normal course of administrative scope boundary maintenance. In this
way they will be able to learn which scopes exist (via Zone
Announcement Messages, ZAMs) and which of these scopes do not nest
(via Not Inside Messages, NIMs). This state must be cached within the
MADCAP server.

When a MADCAP server S receives a NIM from a ZBR containing
information that scope X does not nest in scope Y, it MUST update its
internal state in the following manner.

    1) S MUST update its internal X/Y state to "false".
    2) S MUST restart NEST_NO_DECISION_TIMER for the newly updated
       X/Y state.

4.2 Updating State for Dynamic Scopes due to timer expiration.

MADCAP servers will update X/Y nesting state upon the expiration of
timers in the following manner.

 o If the NEST_NO_DECISION_TIMER expires for a state entry X/Y AND no
   MADCAP messages have been received that indicate scope X does not
   nest inside scope Y, a MADCAP Server, S, MUST conclude that scope
X
   nests inside scope Y. As a result S will change X/Y from "false"
   to "true".

   When a state change from "false" to "true" occurs for X/Y, S must
   also start the ZONES_EXIST_TIMER timer for X/Y. The
   ZONES_EXIST_TIMER should only reset when a Zone Announcement
Message
   (ZAM) has been received for both zone X and zone Y since the
   last time it was restarted. This ensures that both zone X and
   zone Y are known to still exist.

 o If the ZONES_EXIST_TIMER expires for a state entry X/Y, S
   SHOULD conclude that either zone Y or zone X no longer exists and
   hence that both X/Y and Y/X state should be destroyed.

5. Multicast Scope Nesting State Option Format

```
        Code          Len     Count  Nest State Matrix
   +-----+-----+-----+-----+-----+-----+-...-+-----+
   | OPTID     |   p     | m   | N1  |     | Nm  |
   +-----+-----+-----+-----+-----+-----+-...-+-----+
```

Code: 16 bits
     Option identifier (OPTID), to be assigned by IANA.

Len: 16 bits
     The length of the option in bytes.

Count: 8 bits
     The number of zones present in the Nest State Matrix. This value
     MUST be identical to the Count field in the preceding Multicast
     State List option. If this is not the case the scope nesting
     state information MUST BE ignored.

Nest State Matrix:
     The compressed bit-packed representation of the matrix, derived
     in the same manner as shown in Figure 4. Note for N scopes
     the compressed matrix will be N times ceil((N-1)/8) bytes long,
     where ceil() is the function that rounds up to the nearest
integer.
     The scopes corresponding to the rows and columns of this matrix
     list in the same order as they appear in the Multicast Scope
     List Option.

6. Constants

[OPTID] The option identifier for the Multicast Scope Nesting State
        option, to be assigned by IANA.

[NEST_NO_DECISION_TIMER] The time after which a MADCAP server or
        client can assume that a message announcing that two zones
        do not nest should not be received. The length of this timer
        is dependent upon the zone announcement protocol used to
        inform the MADCAP router of which zones currently exist.
        When MZAP [RFC2776] is used this value should be greater than
        the MZAP timeout value NIM-INTERVAL +30%. This corresponds
        to a timeout value of 1800 + 30% = 2340 seconds (39 minutes).

[ZONES_EXIST_TIMER] The time after which a MADCAP server or client
        should assume that the zone in question does not exist when
        zones are detected dynamically. The length of this timer is
        dependent upon the zone announcement protocol used to inform
        the MADCAP router of which zones currently exist. When MZAP

         [RFC2776] is used this value should be no less than the MZAP
         timeout value NIM-HOLDTIME, which has a default of
         5460 seconds (91 minutes).

7. Security Considerations

   Since this draft proposes an extension to the MADCAP protocol via the
   addition of a new option, the same set of security concerns apply.

   In addition to these concerns are those that would arise were the
   information in the Multicast Scope Nesting State option to be
   falsified. In this case the clients would be misinformed as to which
   scopes nest inside one another. In this event, the client would then
   make incorrect decisions regarding the order in which to use the
   scopes. The effect of this would be to use larger scopes than
   necessary, which would effectively flatten any scope hierarchy
   present and nullify the advantage afforded by the hierarchy's
   presence.

   Thus a malformed or tampered Multicast Scope Nesting option may cause
   protocols that rely upon the existence of a scoping hierarchy to
   scale less well, but it would not prevent them from working.

8. IANA Considerations

   The Multicast Nesting State Option will require the assignment of an
   additional option code to the MADCAP [RFC2730] protocol.

9. Acknowledgments

   The Author would like to acknowledge Mark Handley and Dave Thaler for
   the helpful discussions and feedback which helped shape and refine
   this document.

10. References

   [KERM]   Kermode, R., "Smart Network Caches: Localized Content
              and Application Negotiated Recovery Mechanisms for
              Multicast Media Distribution", Ph.D. Thesis, MIT
              Media Laboratory, June 1998.

   [MAAA]   Handley, M., Thaler, D., and D. Estrin, "The Internet
              Multicast Address Allocation Architecture", Internet
              Draft, December 1997.

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

[RFC2365]   Meyer, D., "Administratively Scoped IP Multicast", BCP,
            RFC 2365, July 1998.

[RFC2730]   Patel. B.V., Shah, M., Hanna, S.R., "Multicast Address
            Dynamic Client Allocation Protocol (MADCAP)", RFC 2730,
            December 1999

[RFC2776]   Handley, M., Thaler, D., Kermode, R., "Multicast-Scope
            Zone Announcement Protocol (MZAP)", RFC 2776, February
            2000

11. Author's Address

Roger Kermode
Motorola Australian Research Centre
Locked Bag 5028
Botany, NSW 1455
Australia
Email: Roger.Kermode@motorola.com

12. Full Copyright Statement