                   Candidate Access Router Discovery (CARD)

Status of This Memo

   This memo defines an Experimental Protocol for the Internet
   community. It does not specify an Internet standard of any kind.
   Discussion and suggestions for improvement are requested.
   Distribution of this memo is unlimited.

Copyright Notice

Abstract

   To enable seamless IP-layer handover of a mobile node (MN) from one
   access router (AR) to another, the MN is required to discover the
   identities and capabilities of candidate ARs (CARs) for handover
   prior to the initiation of the handover. The act of discovery of
   CARs has two aspects: identifying the IP addresses of the CARs and
   finding their capabilities. This process is called "candidate access
   router discovery" (CARD). At the time of IP-layer handover, the CAR,
   whose capabilities are a good match to the preferences of the MN, is
   chosen as the target AR for handover. The protocol described in this
   document allows a mobile node to perform CARD.

Table of Contents

1. Introduction

   IP mobility protocols, such as Mobile IP, enable mobile nodes to
   execute IP-level handover among access routers. Work is underway
   [Kood03][Malk03] to extend the mobility protocols to allow seamless
   IP handover. Seamless IP mobility protocols will require knowledge
   of candidate access routers (CARs) to which a mobile node can be
   transferred. The CAR discovery protocol enables the acquisition of
   information about the access routers that are candidates for the
   mobile node's next handover.

   CAR discovery involves identifying a CAR's IP address and the
   capabilities that the mobile node might use for a handover decision.
   There are cases in which a mobile node has a choice of CARs. The
   mobile node chooses one according to a match between the mobile
   node's requirements for a handover candidate and the CAR's
   capabilities. However, the decision algorithm itself is out of the
   scope of this document.

The problem statement for CAR discovery is documented in [TKCK02].
In this document, a protocol is described to perform CAR discovery.
Section 3 describes two main functions of the CAR discovery protocol.
Section 4 describes the core part of the CARD protocol operation.
The protocol message format is described in Section 5. Section 6
discusses security considerations, and Section 7 contains a table of
protocol parameters. Appendix A contains two alternative techniques
for dynamically constructing the CAR table mapping between the access
point L2 ID and Access Router IP address, which is necessary for
reverse address translation. The default method is static
configuration. Appendix B contains two sample scenarios for using
CARD.

2. 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 [Brad97].

   This document uses terminology defined in [MaKo03].

   In addition, the following terms are used:

   Access Router (AR)

      An IP router residing in an access network and connected to one or
      more APs. An AR offers IP connectivity to MNs.

   Candidate AR (CAR)

      An AR to which an MN has a choice when performing IP-level
      handover.

   Capability of an AR

      A characteristic of the service offered by an AR that may be of
      interest to an MN when the AR is being considered as a handover
      candidate.

   L2 ID

      An identifier of an AP that uniquely identifies that AP. For
      example, in 802.11, this could be a MAC address of an AP.

CARD Initiating Trigger

An L2 trigger used to initiate the CARD process. For example, a
MN can initiate CARD as soon as it detects the L2 ID of a new AP
during link layer scan.

Access Point (AP)

A wireless access point, identified by a MAC address, providing
service to the wired network for wireless nodes.

3. CARD Protocol Functions

The CARD protocol accomplishes the following functions.

3.1. Reverse Address Translation

If an MN can listen to the L2 IDs of new APs prior to making a
decision about IP-level handover to CARs, a mechanism is needed for
reverse address translation. This function of the CARD protocol
enables the MN to map the received L2 ID of an AP to the IP address
of the associated CAR that connects to the AP. To get the CAR's IP
address, the MN sends the L2 ID of the AP to the current AR, and the
current AR provides the associated CAR's IP address to the MN.

3.2. Discovery of CAR Capabilities

Information about the capabilities of CARs can assist the MN in
making optimal handover decisions. This capability information
serves as input to the target AR selection algorithm. Some of the
capability parameters of CARs can be static, whereas others can
change with time.

A definition of capabilities is out of the scope of this document.
Encoding rules for capabilities and the format of a capability
container for capability transport are specified in Section 5.

4. CARD Protocol Operation

The CARD protocol allows MNs to resolve the L2 ID of one or more APs
to the IP addresses of the associated CARs. The L2 IDs are typically
discovered during an operation by the MN and are potential handover
candidates. Additionally, CARD allows MNs to discover particular
capabilities associated with the CARs, such as available bandwidth,
that might influence the handover decision of the MN. Furthermore,
the protocol allows ARs to populate and maintain their local CAR
table (Section 4.1) with the capabilities of CARs. For this, the
CARD protocol makes use of CARD Request and CARD Reply messages

between an MN and its current AR (Section 5.1.2), and between an MN's
current AR and individual CARs, respectively (Section 5.2.2).

To allow an MN to retrieve a CAR's address and capability
information, the CARD Request and CARD Reply messages used between an
MN and its current AR may contain one or more access points' L2 IDs
and the IP addresses of associated CARs, respectively. Optionally,
the CARD Reply messages can also contain a CAR's capability
information. A CAR's capabilities are specified as a list of
attribute-value pairs, which are conveyed in a Capability Container
message parameter.

Information about CARs and associated capabilities MAY be used by the
MN to perform target access router selection during its IP handover.
The current AR returns replies according to its CAR table (see
Section 4.1) and returns a RESOLVER ERROR (see Section 5.1.3.1) if
the request cannot be resolved.

The CARD protocol also enables an MN to optionally indicate its
preferences on capabilities of interest to its current AR by
including the Preferences message parameter in the CARD Request
message. The MN's current AR MAY use this information to perform
optional capability pre-filtering for optimization purposes, and it
returns only these capabilities of interest to the requesting MN.
The format of this optional Preferences message parameter is
described in Section 5.1.3.2.

Optionally, the MN can provide its current AR with a list of
capability attribute-value pairs, indicating not only the capability
parameters (attributes) required for capability pre-filtering, but
also a specific value for a particular capability. This allows the
MN's current AR to perform CAR pre-filtering and to send only address
and capability information of CARs whose capability values meet the
requirements of the MN back to the requesting MN. The format of this
optional Requirements message parameter is described in Section
5.1.3.3.

For example, using the optional Preferences message parameter, an MN
may indicate to its current AR that it is interested only in
IEEE802.11a interface-specific capability parameters, as this is the
only interface the MN has implemented. The MN's current AR sends
back only CARs with IEEE802.11a-specific capabilities. Similarly,
using the optional Requirements message parameter, an MN may indicate
to its current AR that it is only interested in CARs that can satisfy
a given QoS constraint. Here, an MN sends the respective QoS
attribute with the QoS constraint value to its current AR using the
optional Requirements message parameter. The QoS constraint is
denoted as an attribute-value pair and encapsulated with the

Requirements message parameter, which is appended to the MN-
originated CARD Request message. The Requirements message parameter
may be used to indicate the cutoff values of the capabilities for any
desired CARs. According to the received optional list of attributes
in the Preferences parameter or a list of attribute-value pairs in
the Requirements message parameter, the MN's current AR MAY use these
parameters for deciding the content of the solicited CARD Reply
message, which is to be sent back to the MN. Alternatively, if the
MN's current AR does not perform optimization with regard to
capability or CAR pre-filtering, the current AR MAY choose to
silently ignore the optional Requirements and Preferences message
parameter as received in the CARD Request message.

The MN can additionally request from the AR a certification path that
is anchored at a certificate from a shared, trusted anchor. The MN
includes in the CARD Request message a list of trusted anchors for
which the MN has a certificate, and the AR replies with the
certification path. If no match is found, the AR returns the trusted
anchor names from the CARD Request. The MN can ask for a chain for
either the current AR or a CAR. If the trusted anchor list is
accompanied by an AP L2 ID for the MN's current AP, the returned
chain is for the current AR. If the L2 ID is for an AP that the MN
has heard during scanning and is not connected to the current AR, the
returned chain is for a CAR. The chain is returned as a sequence of
CARD Reply messages, each message containing a single certificate,
the L2 identifier for the AP sent in the CARD Request, and a router
address for the CAR (or for the AR itself if a request was made for
the AR). When the chain is complete, the MN can use it to obtain the
AR's certified key and thereby validate signatures on CARD messages
and other messages between the MN and the current AR. The MN only
has to send the trusted anchor option if it does not have the
certification path for the AR already cached. If the MN has the
certification path cached, through preconfiguration, through previous
receipt of the chain from this router, or by having received the
chain through a previous router, then the trusted anchor does not
have to be sent. More information about certificate exchange and its
use in CARD security can be found in Section 6.

The CARD protocol operation, as described in this section,
distinguishes signaling messages exchanged between an MN and its
current AR from those exchanged between ARs. Hence, descriptions of
signaling messages in the following sections have preceding
identifiers referring to the associated interface. Messages that are
exchanged between an MN and AR are designated as "MN-AR", and
messages between ARs are designated as "AR-AR".

```
      +--------------+  (1a)AR-AR CARD Request  +----------+
      |   Current    |------------------------->|   CAR    |
      |     AR       |<-------------------------|          |
      +--------------+  (2a)AR-AR CARD Reply    +----------+
          ^        |
          |        |    MN-AR
    MN-AR  |        | CARD Reply(3m)
  CARD Request(2m)  V
          +--------------+
          |    Mobile    |
          |     Node     |<-- CARD Init Trigger
          +--------------+        (1m)
```
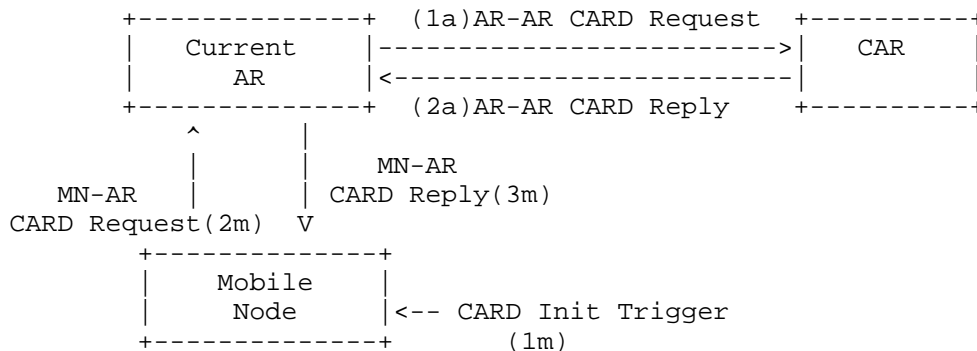
            Figure 1: MN-initiated CARD Protocol Overview

   Figure 1 describes the operation of the MN-AR CARD Request/Reply
   protocol and AR-AR CARD Request/Reply protocol. On receipt of the
   access points' L2 IDs or the appearance of a CARD initiation trigger
   (1m), the MN may pass on one or more AP L2 IDs to its current AR
   using the MN-AR CARD Request message (2m). If the MN wants its AR to
   perform capability discovery in addition to reverse address
   translation, this must be indicated in the MN-AR CARD Request message
   by setting the C-flag. If the C-flag is not set, the AR receiving
   the CARD Request message will perform only reverse address
   translation. The MN's current AR resolves the L2 ID to the IP
   address of the associated CAR or, if the MN has not attached any L2
   ID message parameters, just reads out all CARs' IP address
   information using the reverse address translation information (L2 ID
   to IP address mapping) from its local CAR table. The current AR then
   returns to the MN using the MN-AR CARD Reply message (3m), the IP
   addresses of any CARs, each CAR's set of L2 IDs with CANDIDATE
   indicated in the L2 ID sub-option status field, and, if capability
   information has been requested, associated capabilities.

   For the AR-AR CARD Request/Reply protocol, the requesting AR sends a
   CARD Request message to its peer when the CAR table entries time out
   (1a). The peer returns a CARD Reply message with the requested
   information (2a).

4.1. Conceptual Data Structures

   ARs SHALL maintain an L2-L3 address mapping table (CAR table) that is
   used to resolve L2 IDs of candidate APs to the IP address of the
   associated CAR. By default, this address-mapping table is configured
   statically for the CARD protocol operation. Optionally, the CAR
   table MAY be populated dynamically. Two possible approaches are
   described in Appendices A.1 and A.2.

ARs SHOULD also keep and maintain individual CARs' capabilities in
the local CAR table, with the associated capability lifetime taken
into account. If the lifetime of an individual capability entry has
expired, the respective capability information is updated. An AR may
also initiate capability exchange prior to expiration of the
capabilities associated with a CAR in the CAR table, thereby
populating its CAR table. The AR's CAR table may be implemented
differently; therefore additional details are not provided here. ARs
MUST maintain their own AP-to-AR mappings and capability information
in their CAR tables, in order to provide newly booted MNs with this
information so that an MN can obtain the AR's certification path.

MNs SHOULD maintain discovered address and capability information of
CARs in a local cache to avoid requesting the same information
repeatedly and to select an appropriate target AR from the list of
CARs as quickly as possible when a handover is imminent.

4.2. Mobile Node - Access Router Operation

4.2.1. Mobile Node Operation

To initiate CARD, an MN sends a CARD Request to its current AR,
requesting it to resolve the L2 ID of nearby access points to the IP
address of associated CARs and also obtain capability parameters
associated with these CARs. If the requesting MN wants its current
AR to resolve specific L2 IDs, the MN-AR CARD Request MUST contain
the CARD protocol-specific L2 ID message parameters. If the MN wants
its AR to perform only reverse address translation without appending
the CARs' capabilities, the MN refrains from setting the C-flag in
the CARD Request message. If the MN wants to perform capability
discovery, the MN MUST set the C-flag in the CARD Request message.
The CARD Request MAY also contain the Preferences or Requirements
message parameter, indicating the MN's preferences on capability
attributes of interest or its requirements on CARs' capability
attribute-value pairs.

If the MN appends multiple L2 ID sub-options to a CARD Request, the
AR MUST assume that each L2 ID is associated with an AP that connects
to a different CAR. Since L2 IDs, address information, and
capability information are transmitted with separate sub-options,
each sub-option carries a Context-ID, to allow parameters that belong
together to be matched. Therefore, the MN MUST assign different
Context-ID values to the L2 ID sub-options it appends to the CARD
Request message. The Status-Code field of the L2 ID sub-option MUST
always be set to NONE (0x00) by the MN. The MN MUST set the sequence
number to a randomly generated value, and the AR MUST include the
sequence number in all messages of the reply. If the reply spans
multiple messages, each message contains the same sequence number.

Upon receipt of the corresponding MN-AR CARD Reply message, the MN
correlates the CARD Reply with the appropriate CARD Request message
and then processes all MN-AR CARD Reply message parameters to
retrieve its CAR's address and capability information. If the MN is
unable to correlate the CARD Reply with any previously sent CARD
Request messages, the MN SHOULD silently discard the reply. This may
happen when the MN reboots after sending a CARD Request message to
the connected AR.

An MN uses exponential backoff to retransmit the CARD Request in the
event that a CARD Reply is not received within CARD_REQUEST_RETRY
seconds. The retransmitted CARD Request MUST have the same sequence
number as the original. With the exception of certification paths,
which are large by nature, an AR SHOULD attempt to limit the
information in a CARD Reply to a single message. Should that be
impossible, the AR MAY send the reply in multiple messages. The last
message of a reply MUST always have the L-flag set in the CARD Reply
option to indicate that the message is the last for the associated
sequence number. An AR retransmitting replies to a CARD Request MUST
always send the full CARD Reply sequence. The Trusted Anchor sub-
option and the Router Certificate sub-option provide a means whereby
the MN can request specific certificates in a certification path, in
the event that the CARD Reply carrying a certification path spans
multiple messages and one of them is lost. However, a request for
specific certificates that were not received in the initial CARD
Reply MUST be treated as a new request by the MN and MUST use a
different sequence number.

Processing the Context-ID of Address sub-options allows the MN to
assign the resolved IP address of a specific CAR to an L2 ID.

In some cases, an L2 ID parameter is present in a CARD Reply message.
The Status-Code field in the L2 ID parameter indicates one of the
following reasons for its being sent toward the MN.

RESOLVER ERROR Status-Code indication:
   If the MN's current AR could not resolve a particular L2 ID, this
   status code is returned to the MN.

MATCH Status-Code indication:
   If an L2 ID is encountered that shares a CAR with a previously
   resolved L2 ID, the AR returns MATCH to the MN. This status code
   indicates that the Context-ID of this particular L2 ID sub-option
   has been set to the Context-ID of the associated CAR's Address and
   Capability Container sub-option, which is sent with this CARD
   Reply message. This approach avoids sending the same CAR's
   address and capability information multiple times with the same
   CARD Reply message in case two or more L2 IDs resolve to the same

CAR. An MN uses the Context-ID received in the L2 ID sub-option
as the key to find the serving CAR of the given AP from the
content of the received CARD Reply message.

CANDIDATE Status-Code indication:
   If the MN does not append any L2 ID to the CARD Request, the AR
   sends back the L2 ID and address information of all CARs. Because
   the received parameters' Context-IDs cannot be correlated with an
   L2 ID's Context-ID of a previously sent request, the AR chooses
   values for the Context-ID and marks these candidate L2 IDs with
   CANDIDATE in the status code of the distributed L2 IDs. However,
   individual values of L2 IDs' Context-ID allow the MN to assign a
   particular L2 ID to the associated Address and the possibly
   received Capability Container sub-option.

   As described in Section 4.5, an MN can use CARD when it initially
   boots up to determine whether piggyback operation is possible. An
   MN can also use CARD initially to determine the capabilities and
   certificates for an AR on which it boots up or if it cannot obtain
   the certificates beforehand. To do this, the MN includes an L2
   Identifier option with its current AP L2 ID and the requested
   information. The AR replies with its own information.

4.2.2. Current Access Router Operation

   Upon receipt of an MN's MN-AR CARD Request, the connected AR SHALL
   resolve the requested APs' L2 ID to the IP address of any associated
   CARs. If no L2 ID parameter has been sent with the MN-AR CARD
   Request message, the receiving AR retrieves all CARs' IP addresses
   and, if the C-flag was set in the request, the capability
   information.

   In the first case, where the AR resolves only requested L2 IDs, the
   AR does not send back the L2 ID to the requesting MN. If, however,
   two or more L2 IDs match the same CAR information, the L2 ID sub-
   option is sent back to the MN, indicating a MATCH in the Status-Code
   field of the L2 ID. Furthermore, the AR sets the Context-ID of the
   returned L2 ID to the value of the resolved CAR's L2 ID, Address, and
   Capability Container sub-option. If an AR cannot resolve a
   particular L2 ID, an L2 ID sub-option is sent back to the MN,
   indicating a RESOLVER ERROR in the L2 ID sub-option's Status-Code
   field.

   In the second case, where the AR did not receive any L2 ID with a
   CARD Request, all candidate APs' L2 IDs are sent to a requesting MN
   with the CARD Reply message. The AR marks the Status-Code of
   individual L2 IDs as CANDIDATE, indicating to the MN that the

associated Context-ID cannot be matched with the ID of a previously
sent request.

In any case, the AR MUST set the Context-ID of the Address and the
Capability Container sub-option to the same value as that of the
associated L2 ID sub-option.

Optionally, when allowed by local policies and supported by
respective ARs for capability discovery, the AR MAY retrieve a subset
of capabilities or CARs, satisfying the optionally appended
Preferences and Requirement message parameter, from its local CAR
table. CARs' address information and associated capabilities are
then delivered to the MN using the MN-AR CARD Reply message. The
CARs' IP address and the capabilities SHALL be encoded according to
the format for CARD protocol message parameters as defined in Section
5.1.3 of this document. The capabilities are encoded as attribute-
value pairs, which are encapsulated in a Capability Container message
parameter according to the format defined in Section 5.1.3.4. The
responding current AR SHALL copy the sequence number received in the
MN-AR CARD Request to the MN-AR CARD Reply.

4.3. Current Access Router - Candidate Access Router Operation

4.3.1. Current Access Router Operation

The MN's current AR MAY initiate capability exchange with CARs either
when it receives an MN-AR CARD Request or when it detects that one or
more of its local CAR table's capability entries' lifetimes are about
to expire. An AR SHOULD preferentially utilize its CAR table to
fulfill requests rather than signal the CAR directly, and it SHOULD
keep the CAR table up to date for this purpose, in order to avoid
injecting unnecessary delays into the MN response.

The AR SHOULD issue an AR-AR CARD Request to the respective CARs if
complete capability information of a CAR is not available in the
current AR's CAR table, or if such information is expired or about to
expire. The AR-AR CARD Request message format is defined in Section
5.2.2. The sequence number on the AR-AR interface starts with zero
when the AR reboots. The sending AR MUST increment the sequence
number in the CARD Request by one each time it sends a CARD Request
message.

The AR MAY append its own capabilities, which are encoded as
attribute-value pairs and encapsulated with the Capability Container
message parameter, to the released AR-AR CARD Request. If the AR-AR
CARD Request conveys the current AR's capabilities to the CAR, the
associated Capability Container can have any value set for the
Context-ID, as there is no need for the receiving CAR to process this

field due to the absence of an L2 ID and an Address sub-option.
Furthermore, the current AR MAY set the P-flag in the Capability
Container sub-option to inform the CAR about its own capability to
perform CARD protocol message piggybacking.

Optionally, a current AR MAY append the Preferences sub-option to the
AR-AR CARD Request to obtain only capability parameters of interest
from a CAR.

Upon receipt of the AR-AR CARD Reply, sent by the CAR in response to
the previously sent request, the MN's current AR SHALL extract the
capability information from the payload of the received message and
store the received capabilities in its local CAR table. The lifetime
of individual capabilities is to be set according to the lifetime
indicated for each capability received. The values of the table
entries' timeouts shall depend upon the nature of individual
capabilities.

Optionally, CARs can send unsolicited CARD Reply messages to globally
adjacent ARs if the configuration of their APs or capabilities
changes dynamically. If the current AR receives an unsolicited CARD
Reply message from a CAR for which there is an entry in its local CAR
table, the current AR checks that the sequence number of the received
CARD Reply has increased compared to that of the previously received
unsolicited CARD Reply message, which has been sent from the same
CAR. Then, the current AR can update its local CAR table according
to the received capabilities. If a new CAR is added, an AR may
receive a CARD Reply from a CAR that is not in its CAR table, or from
a CAR that has rebooted. In this case, the sequence number is 0.
The requirement that ARs share an IPsec security association,
detailed in Section 6, ensures that an AR never accepts CARD
information from an unauthenticated source.

4.3.2. Candidate Access Router Operation

Upon receipt of an AR-AR CARD Request, a CAR shall extract the
sending AR's capabilities, if the sending AR has included its
capabilities. The CAR SHALL store the received capabilities in its
CAR table and set the timer for individual capabilities
appropriately. The values of the table entries' timeouts depend on
the nature of capabilities in the AR-AR CARD Reply message. The CAR
must include the same sequence number in the AR-AR CARD Reply Message
as that received in the AR-AR CARD Request Message. The AR-AR CARD
Reply shall include the CAR's capabilities as list of attribute-value
pairs in the Capability Container message parameter. If the sending
AR has appended an optional Preferences sub-option, the CAR MAY
perform capability filtering and send back only those capabilities of
interest to the requesting AR, identified according to the

Preferences sub-option. Because the AR-AR CARD Reply is based on a previously received AR-AR CARD Request, the CAR MUST set the U-flag of the AR-AR CARD Reply to 0.

Optionally, the CAR MAY send an unsolicited CARD Reply message to globally adjacent ARs if one or more of its capability parameters change. Each unsolicited CARD Reply message should have as destination address the adjacent AR's unicast address and must have the U-flag set. Consecutive unsolicited CARD Reply messages MUST have the sequence number incremented accordingly, starting with 0 when the AR boots.

4.4. CARD Protocol Message Piggybacking on the MN-AR Interface

CARD supports another mode of CAR information distribution, in which the capabilities are piggybacked on fast handover protocol messages. To allow MNs and ARs appending the ICMP-option type CARD Request and CARD Reply (Section 5.1.2) to the ICMP-type Fast Mobile IPv6 [Kood03] signaling messages, the MN and AR should know about the signaling peer's capability for CARD protocol message piggybacking. This requires dynamic discovery of piggybacking capability using the P-flag in the MN-AR CARD Request and the MN-AR CARD Reply message, as well as in the Capability Container message parameter. The format of these messages and parameters is described in Section 5.1.

The MN sends the very first CARD Request to its current AR using the ICMP-type CARD main header for transport, as described in Section 4.2.1. If the MN supports CARD-protocol message piggybacking, the P-flag in this very first CARD Request message is set. On receipt of the CARD Request message, the current AR learns about the MN's piggybacking capability. To indicate its piggybacking capability, the AR sets the P-flag in the CARD Reply message. If the AR does not support piggybacking, all subsequent CARD-protocol messages between the MN and the AR are sent stand-alone, using the CARD main header. If both nodes (the MN and its current AR) support CARD-protocol message piggybacking, subsequent CARD protocol messages can be conveyed as an option via the Fast Mobile IPv6 Router Solicitation for Proxy (RtSolPr) and Proxy Router Advertisement (PrRtAdv) messages. During the CARD process, an MN learns about CARs' piggybacking capability at the discovery phase, as the Capability Container (described in Section 5.1.3.4) also carries a P-flag. This allows the MN to perform CARD protocol message piggybacking immediately after a handover to a selected CAR, assuming that this CAR supports CARD protocol piggybacking.

If a MN prefers the reverse address translation function of the Fast Mobile IPv6 protocol, it can use CARD protocol message piggybacking to retrieve only the CARs' capability information. To indicate that

   reverse address translation is not required, the piggybacked CARD
   Request message MUST have the A-flag set. This causes the current AR
   to append only Capability Container sub-options. To associate a
   Capability Container sent as a parameter of the CARD Reply message to
   the IP address for the appropriate CAR, the Context-ID of an
   individual Capability Container MUST be used as an index, pointing to
   the associated IP address in the PrRtAdv message options. The
   Context-ID of individual Capability Containers is set appropriately
   by the MN's current AR. Details about how individual Context-ID
   values can be associated with a particular IP address option of the
   PrRtAdv message is out of the scope of this document.

5. Protocol Messages

5.1. CARD Messages for the Mobile Node-Access Router Interface

5.1.1. MN-AR Transport

   The MN-AR interface uses ICMP for transport. Because ICMP messages
   are limited to a single packet, and because ICMP contains no
   provisions for retransmitting packets if signaling is lost, the CARD
   protocol incorporates provisions for improving transport performance
   on the MN-AR interface. MNs SHOULD limit the amount of information
   requested in a single ICMP packet, as ICMP has no provision for
   fragmentation above the IP level.

   MNs and ARs use the Experimental ICMP-type main header [Ke04] when
   CARD protocol messages cannot be conveyed via ICMP-type Fast Mobile
   IPv6 [Kood03]. The MN-AR interface MUST implement and SHOULD use the
   CARD ICMP-type header for transport. If available, the MN-AR
   interface MAY use the ICMP-type Fast Mobile IPv6 [Kood03] for
   transport (Section 4.4).

```
     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
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
    |     Type      |     Code      |           Checksum            |
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
    |    Subtype    |                 Reserved                      |
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
    |   Options ...
    +-+-+-+-+-+-+-+-+-+-+- - - -
```
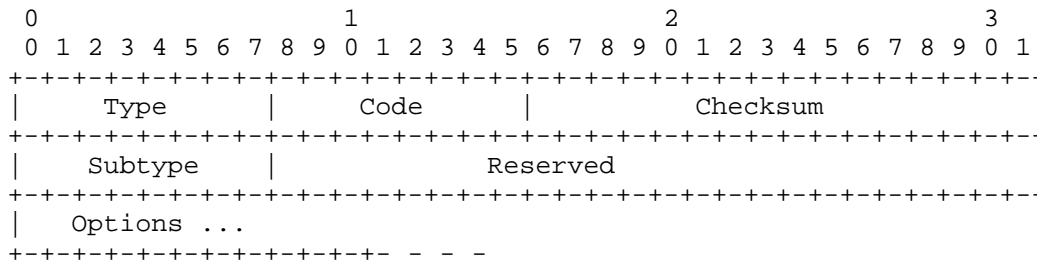
   IP Fields:

      Source Address:
                  An IP address assigned to the sending interface.

Destination Address:
              An IP address assigned to the receiving interface.

   Hop Limit: 255

ICMP Fields:

   Type: Experimental Mobility type (assigned by IANA for
                   IPv4 and IPv6, see [Ke04]).

   Code: 0

   Checksum: The ICMP checksum.

   Subtype: Experimental Mobility subtype for CARD; see [Ke04].

   Reserved: This field is currently unused. It MUST be
                   initialized to zero by the sender and MUST be
                   ignored by the receiver.

Valid Options:

   CARD Request: The CARD Request allows entities to request CARD-
                   specific information from ARs. To support
                   processing of the CARD Request message on the
                   receiver side, further sub-options may be carried,
                   serving as input to the reverse address translation
                   function and/or capability discovery function.

   CARD Reply: The CARD Reply carries parameters, previously
                   requested with a CARD Request, back to the sender
                   of the CARD Request.

Valid Sub-Options:

Support level is indicated in parentheses.

   Layer-2 ID (mandatory):
                   The Layer-2 ID sub-option [5.1.3.1] carries
                   information about the type of an access point as
                   well as the Layer-2 address of the access point
                   associated with the CAR whose IP address and
                   capability information is to be resolved.

Capability Container (mandatory):
              The Capability Container sub-option carries
              information about a single CAR's capabilities. The
              format of this sub-option is described in Section
              5.1.3.4.

Address (mandatory):
              The Address sub-option carries information on an
              individual CAR's resolved IP address. The format
              of the Address sub-option is described in Section
              5.1.3.5.

Trusted Anchor (mandatory):
              The Trusted Anchor sub-option carries the name of a
              trusted anchor for which the MN has a certificate.
              The format of the Trusted Anchor sub-option is
              described in Section 5.1.3.6.

Router Certificate (mandatory):
              The Router Certificate sub-option carries one
              certificate in the path for the current AR or for a
              CAR. The chain includes certificates starting at a
              trusted anchor, which the AR shares in common with
              the MN, to the router itself. The format of the
              Router Certificate sub-option is described in
              Section 5.1.3.7.

Preferences (optional):
              The Preferences sub-option carries information
              about attributes of interest to the requesting
              entity. Attributes are encoded according to the
              AVP encoding rule, which is described in Section
              5.1.4. For proper settings of AVP Code and Data
              field, see Section 5.1.3.2. This sub-option is
              used only if optional capability pre-filtering is
              performed on ARs, and it provides only capabilities
              of interest to a requesting MN.

Requirements (optional):
              The Requirements sub-option carries information
              about attribute-value pairs required for pre-
              filtering of CARs on the MN's current AR. This
              parameter conveys MN specific attribute-value pairs
              to allow the MN's current AR to send only
              information about CARs of interest back to the
              requesting MN. CARs are filtered on ARs according
              to the CARs' capability parameters and given policy
              or threshold, as encoded in the Requirements sub-

                        option. Attribute-value pairs are encoded
                        according to the AVP encoding rule, which is
                        described in Section 5.1.4. Rules for proper
                        setting of the AVP Code and Data field for the
                        Requirements sub-option are described in Section
                        5.1.3.3.

   CARD Requests that fail to elicit a response are retransmitted. The
   initial retransmission occurs after a CARD_REQUEST_RETRY wait period.
   Retransmissions MUST be made with exponentially increasing wait
   intervals (doubling the wait each time). CARD Requests should be
   retransmitted until either a response (which might be an error) has
   been obtained or CARD_RETRY_MAX seconds have occurred. ARs MUST
   discard any CARD Requests having the same sequence number after
   CARD_RETRY_MAX seconds. If a CARD Reply spans multiple ICMP
   messages, the same sequence number MUST be used in each message.

   MNs that retransmit a CARD Request use the same CARD sequence number.
   This allows the AR to cache its reply to the original request and
   then to send it again, should a duplicate request arrive. This
   cached information should only be held for a maximum of
   CARD_RETRY_MAX seconds after receipt of the request. Sequence
   numbers SHOULD be chosen randomly. Random sequence numbers avoid
   duplicates if MNs restart frequently and simplify sequence-number
   maintenance on both the MN and AR when MNs frequently appear and
   disappear due to movement between CARs.

5.1.2. CARD Options Format

   All options are of the following form:

```
    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
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |     Type      |    Length     |Vers.|          ...           |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   ~                             ...                              ~
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```
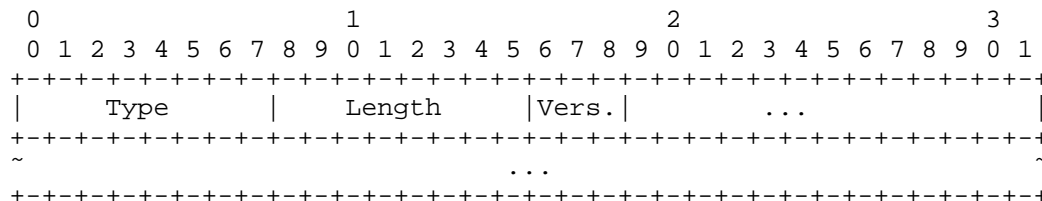
   Fields:

      Type: 8-bit identifier of the type of option, assigned by
                      IANA. See [Ke04] for CARD Request and CARD Reply
                      values.

      Length: 8-bit unsigned integer. The length of the option,
                      including the type and length fields in units of 8
                      octets. The value 0 is invalid.

        Vers.:   3-bit version code. For this specification,
                        Vers.=1.

5.1.2.1. CARD Request Option

```
     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
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
    |     Type      |     Length    |Vers.|P|C|A|T|     Reserved    |
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
    |                        Sequence Number                       |
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
    |     Sub-Options
    +-+-+-+-+-+-+-+-+-+-+-+-+ -   -   -
```
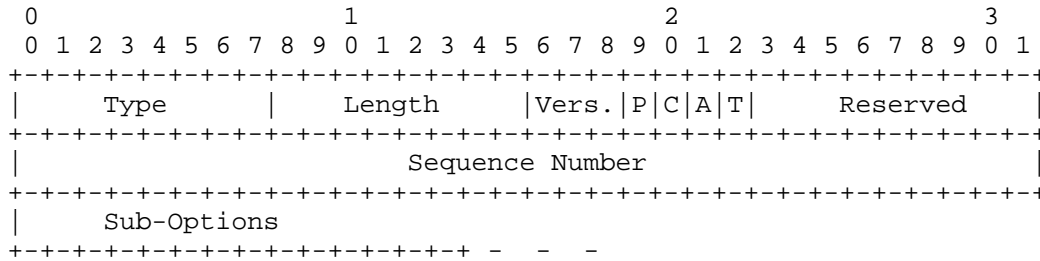
    Fields:

        Type: Assigned by IANA for IPv4 and IPv6; see [Ke04].

        Length: The length of the option in units of 8 octets, including
                 the type and length fields as well as sub-options.

        Vers.:   3-bit version code. For this specification, Vers.=1.

                 Flags: P-flag: Indicates the CARD-protocol message
                                   piggybacking capability of the CARD
                                   Request message sender. A description
                                   for proper use of this flag can be
                                   found in Section 4.4 of this document.

                          C-flag: Indicates that the requesting entity is
                                   also interested in associated CARs'
                                   capabilities. If the MN wants the AR
                                   to append CARs' capability parameters
                                   to the CARD Reply in addition to
                                   address information, the MN must set
                                   this flag.

                          A-flag: Indicates that the requesting entity
                                   does NOT want the receiver of this
                                   message to perform reverse address
                                   translation. This flag is set if CARD
                                   protocol messages are piggybacked with
                                   a protocol that performs reverse
                                   address translation. For details,
                                   refer to Section 4.4 of this document.

                        T-flag: Indicates that the requesting entity is
                                 interested in obtaining all
                                 certificates from the responder. This
                                 flag is only valid on the AR-AR
                                 interface.

             The flag combination A=1 and C=0 is invalid, and the flag
             T=1 is invalid on the MN-AR interface. The AR MUST
             discard an invalid message and log an appropriate error
             message.

        Reserved:
                Initialized to zero, ignored on receipt.

        Sequence Number:
                Allows requests to be correlated with replies.

     Valid Sub-Options:

        - L2 ID sub-option
        - Preferences sub-option
        - Requirements sub-option
        - Trusted Anchor sub-option

     To ensure that requirements on boundary alignment are met, individual
     sub-options MUST meet the 64-bit boundary alignment requirements
     respectively. This will ensure that the entire CARD Request option
     meets the 8n alignment constraint.

5.1.2.2. CARD Reply Option

     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
    +-+
    | Type | Length |Vers.|P|U|L| Reserved |
    +-+
    | Sequence Number |
    +-+
    | Sub-Options
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+ - - -
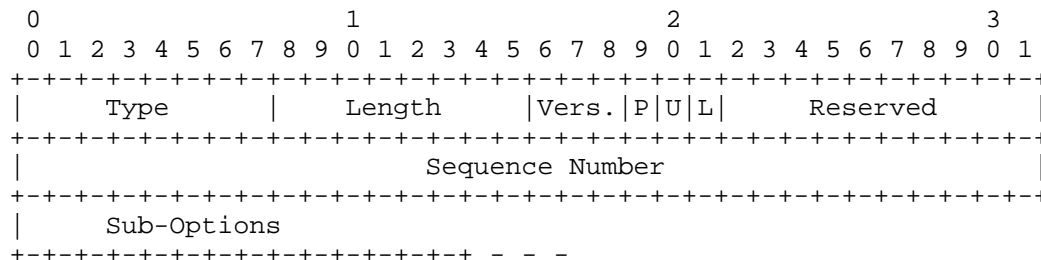
     Fields:

        Type: Assigned by IANA for IPv4 and IPv6 [Ke04].

        Length: The length of the option in units of 8 octets, including
                 the type and length fields as well as sub-options.

Vers.:   3-bit version code. For this specification, Vers.=1.

   Flags: P-flag: Indicates the CARD-protocol message
                     piggybacking capability of the CARD
                     Reply message sender. A description
                     for proper use of this flag can be
                     found in Section 4.4 of this document.

            U-flag: Indicates an unsolicited CARD Reply.
                     This flag is only valid on the AR-AR
                     interface.

            L-flag: Set if this message is the last message
                     in a multiple ICMP message reply. This
                     flag is only valid on the MN-AR
                     interface.

   The flag U=1 on an AR-MN message is invalid. An invalid
   message should be discarded and an appropriate error
   message logged.

Reserved:
        Initialized to zero, ignored on receipt.

Sequence Number:
        Allows requests to be correlated with replies.

Valid Sub-Options:

  - L2 ID sub-option
  - Capability Container sub-option
  - Address sub-option
  - Router Certificate sub-option

To ensure requirements on boundary alignment are met, individual
sub-options MUST meet 64-bit boundary alignment requirements
respectively. This will ensure that the entire CARD Request option
meets the 8n alignment constraint.

5.1.3. Sub-Options Format

All sub-options are of the following form:

```
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
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|Sub-Option Type|Sub-Option Len |      Sub-Option Data . . .
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```
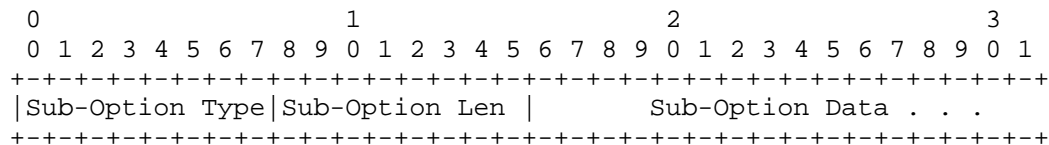
Sub-Option Type: 8-bit identifier of the type of option. The
                  sub-options defined in this document are listed
                  in the table below. The table also indicates
                  on which interfaces the sub-option is valid.

Description	Type	Interface	
		/	\
		MN-AR	AR-AR
L2 ID	0x01	x	
Address	0x02	x	
Capability Container	0x03	x	x
Preferences	0x04	x	x
Requirements	0x05	x	
Trusted Anchor	0x06	x	
Router Certificate	0x07	x	x

Sub-Option-Length: 8-bit unsigned integer indicating the length of
                   the sub-option, including the sub-option type and
                   sub-option length fields. Sub-option lengths are
                   in units of 8 octets, aligned on a 64-bit
                   boundary. Sub-options that are shorter are padded
                   with null octets; the extent of the padding is
                   determined by the sub-option contents.

5.1.3.1. L2 ID Sub-Option

```
 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
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|Sub-Option Type|Sub-Option Len |   Context-ID  |  Status Code  |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|    L2-Type                    |      L2 ID . . .
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+ - - -
```
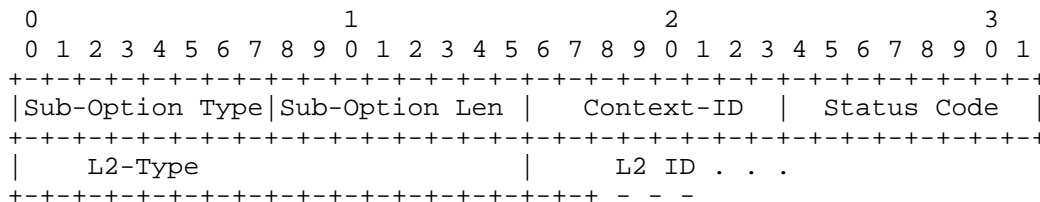
Sub-Option Type:
          0x01

Sub-Option Length:
          Length of the sub-option.

Context-ID: Associates the L2 ID, IP address and other parameters
               that belong to the same AR IP address but are encoded
               in separate sub-options.

Status Code: This field allows ARs to inform a requesting entity
                about processing results for a particular L2 ID. The
                L2 ID sub-option MUST be sent back to the requesting
                entity with a CARD Reply message.

                The following status codes are specified:

                0x00:   NONE - This value MUST be set when the L2 ID is
                        included in a CARD Request.

                0x01:   CANDIDATE - MUST be set in a CARD Reply when a
                        L2 ID sub-option is included with information
                        about candidate APs' L2 IDs. Candidate L2 IDs
                        are sent if the CARD Request did not include a
                        specific L2 ID for resolution. If CANDIDATE is
                        set, the AR MUST set the Context-ID field of
                        individual parameters to a value that allows
                        associated L2 ID, address, and capability
                        information to be matched on the receiver side.

                0x02:   MATCH - MUST be set in the CARD Reply to
                        identify that this L2 ID matches previously
                        resolved CAR information for a different L2 ID.
                        If MATCH is set, the AR sets the Context-ID in
                        the L2-ID sub-option to identify the matching
                        previously resolved L2 ID.

                0x03:   RESOLVER ERROR - MUST be set in the CARD Reply
                        if the L2 ID cannot be resolved. The AR sets
                        this value for the Status Code in the returned
                        L2 ID sub-option.

   L2 type: Indicates the interface type. Allocated by IANA
                   [Ke04].

   L2 ID: The variable length Layer-2 identifier of an
                   individual CAR's access point. The length without
                   padding is determined by the L2 type.

5.1.3.2. Preferences Sub-Option

```
 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
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|Sub-Option Type|Sub-Option Len |         Preferences
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```
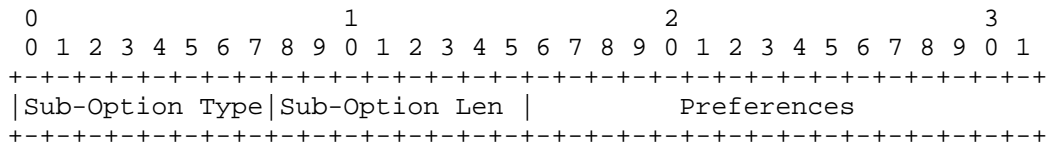
         Sub-Option Type:
                   0x04

         Sub-Option Length:
                   Length of the sub-option.

         Preferences: List of capability attribute values (see Section
                        5.1.4).

         Only ATTRIBUTE (AVP Code; see Section 5.1.4) fields MUST be present
         and set for individual capabilities, which are of interest to the
         requesting entity. The LIFETIME and VALUE (Data) indicator will not
         be processed and can be omitted. The AVP LENGTH indicator is also
         not present, as the preferences are indicated only with a list of
         16-bit encoded ATTRIBUTE fields. If 64-bit boundary alignment
         requirements cannot be met with the list of ATTRIBUTE values, padding
         the missing 16-bit MUST be done with an ATTRIBUTE value of 0x0000.
         An ATTRIBUTE code of 0x0 is reserved so that the end of the ATTRIBUTE
         code list can be determined when an ATTRIBUTE value of 0x0 is read.

         The use of the Preferences sub-option is optional and is for
         optimization purposes.

5.1.3.3. Requirements Sub-Option

```
  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
 +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
 |Sub-Option Type|Sub-Option Len |         Requirements
 +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```
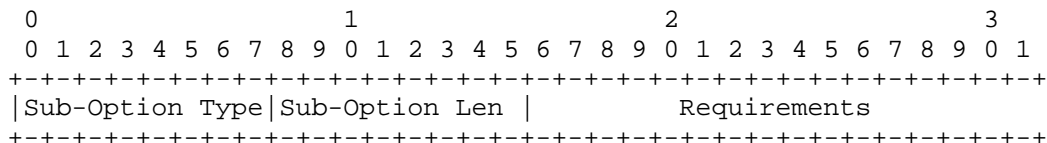
         Sub-Option Type:
                   0x05

         Sub-Option Length:
                   Length of the sub-option.

         Requirements: AVP-encoded requirements (see Section 5.1.4)

         AVPs MUST be encoded according to the rule described in Section
         5.1.4. Both the ATTRIBUTE (AVP Code) and VALUE (Data) fields MUST be
         present and set appropriately. The end of the Requirements list can
         be determined when an ATTRIBUTE value of 0x0 is read.

         The use of the Requirements sub-option is optional and is for
         optimization purposes.

5.1.3.4. Capability Container Sub-Option

```
     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
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
    |Sub-Option Type|Sub-Option Len |   Context-ID  |P|   Reserved  |
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
    |           AVPs
    +-+-+-+-+-+-+-+-+-+-+-+-+-+ - - -
```
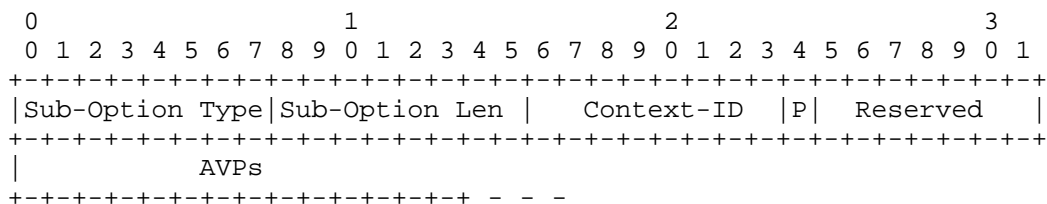
    Sub-Option Type:
                  0x03

    Sub-Option Length:
                  Length of the sub-option.

    Context-ID: Associates the L2 ID, IP address, and other parameters
                   that belong to the same AR IP address but are encoded
                   in separate sub-options.

    Flags: P-flag: Indicates piggybacking capability of the CAR
                            whose capabilities are conveyed in this
                            Capability Container. This flag allows an MN
                            to know after a CARD process whether a
                            selected new AR can perform piggybacking.

    Reserved: Initialized to zero, ignored on receipt.

    AVPs: AVPs are a method of encapsulating capability
                   information relevant for the CARD protocol. See
                   Section 5.1.4 for the AVP encoding rule and list
                   parsing.

5.1.3.5. Address Sub-Option

```
     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
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
    |Sub-Option Type|Sub-Option Len |  Context-ID   | Address Type  |
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
    |           Address . . .
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+- - - - -
```

    Sub-Option Type:
                  0x02

        Sub-Option Length:
                    Length of the sub-option. For IPv4, the length is 1
                    (8 octets); for IPv6 the length is 3 (24 octets).

        Context-ID: Associates the L2 ID, IP address, and other parameters
                      that belong to the same AR IP address but are encoded
                      in separate sub-options.

        Address Type: Indicates the type of the address.

                                    0x01 IPv4
                                    0x02 IPv6

        Address: The Candidate Access Router's IP address.
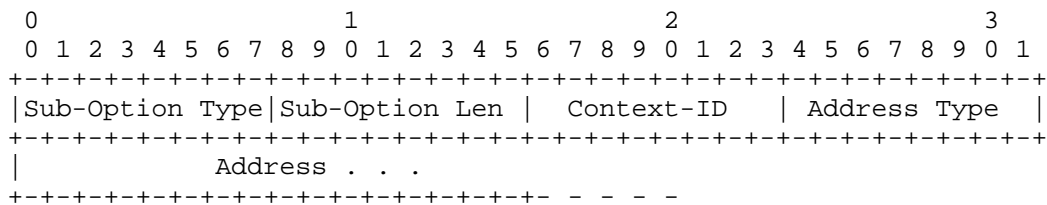
5.1.3.6. Trusted Anchor Sub-Option

        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
       +-+
       |Sub-Option Type|Sub-Option Len | Component |
       +-+
       | Trusted Anchor Name
       +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+- - - - -

        Sub-Option Type:
                    0x06

        Sub-Option Length:
                    Length of the sub-option.

        Reserved: Initialized to zero, ignored on receipt.

        Component: A 2 octet unsigned integer field set to 65,535 if the
                      sender desires to retrieve all the certificates in the
                      certification path. Otherwise, it is set to the
                      component identifier corresponding to the certificate
                      that the receiver wants to retrieve.

        Trusted Anchor Name:
                    DER encoding for the X.501 name of certification path
                    component(see [Arkko04] for more detail on
                    certification path component name encoding).

        A CARD Request message containing Trusted Anchor sub-options MUST NOT
        contain any other sub-options, except for a single L2 ID sub-option
        identifying the AP of interest.

   Trusted anchor sub-options SHOULD be retransmitted for individual
   components not received within CARD_REQUEST_RETRY seconds, rather
   than retransmitting a request for the whole list. Subsequent
   retransmissions SHOULD take into account any received options and
   only request those that have not been received.

5.1.3.7. Router Certificate Sub-Option

```
    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
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |Sub-Option Type|Sub-Option Len |  Context-ID   | Reserved      |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |         All Components        |          Component            |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |                                                               |
   +                                                               +
   |                         Certificate...                        |
   +                                                               +
   |                                                               |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |                          Padding...                           |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

   Sub-Option Type:
                0x07

   Sub-Option Length:
                Length of the sub-option.

   Context-ID: Associates the L2 ID, IP address and other parameters
                that belong to the same AR IP address but are encoded
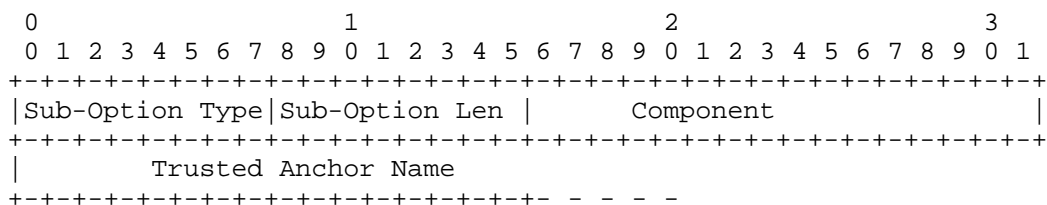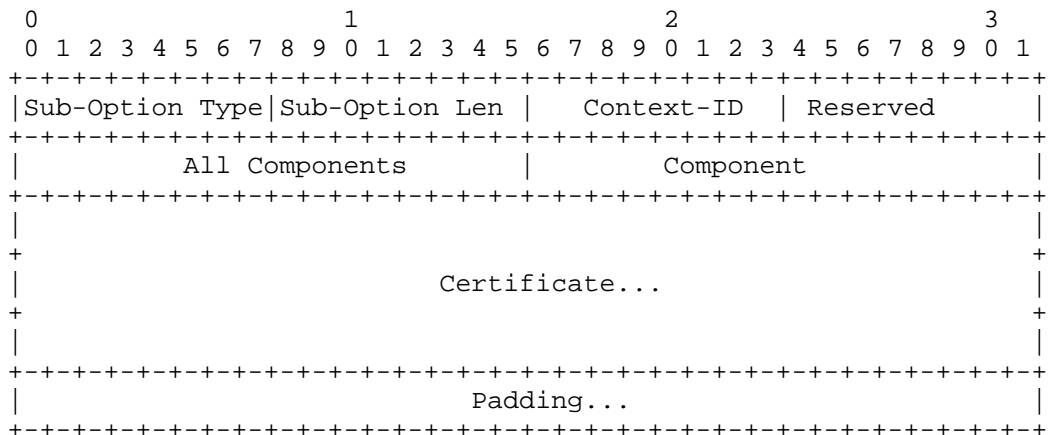   in separate sub-options.

   Reserved: Initialized to zero, ignored on receipt.

   All Components:
                2 octet unsigned integer giving the total number of
                certificates in the certification path.
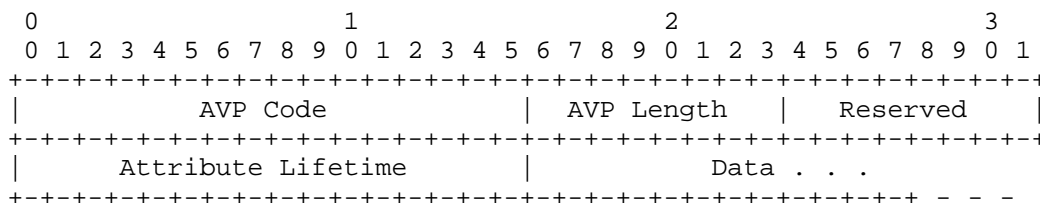
   Component: 2 octet unsigned integer giving the location of this
                certificate in the certification path.

   Certificate: Variable-length field containing the X.509v3 router
                certificate encoded in ASN.1 (see [Arkko04] for more
                detail on a certificate profile that includes
                encoding).

      Padding: Variable-length field making the option length a
                        multiple of 8, beginning after the ASN.1 encoding of
                        the certificate and continuing to the end of the
                        option, as specified by the Length field.

   A CARD Reply containing a Router Certificate sub-option MUST NOT
   include more than one such sub-option, and the CARD Reply MUST
   contain the matching L2 ID sub-option and router Address sub-option
   for the router possessing the chain with the Context-ID field set to
   a nonzero value, and with no other sub-options. Any other sub-
   options included in a CARD Reply SHOULD be ignored. If the reply
   spans multiple ICMP messages, the L2 ID sub-option and router Address
   sub-option MUST be included in the first message sent, and the
   Context-ID field in the Router Certificate sub-options in all the
   messages MUST be set to the same value as that in the L2 ID and
   Address sub-options. The replying AR SHOULD order the returned
   certification path so that the certificate immediately after the
   trust anchor in the path is the first certificate sent, in order to
   allow immediate verification. The trust anchor certificate itself
   SHOULD NOT be sent.

5.1.4. Capability AVP Encoding Rule

```
    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
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |          AVP Code          |   AVP Length  |    Reserved    |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |      Attribute Lifetime    |            Data . . .
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+ - - -
```

   AVP Code: Identifies the attribute uniquely. The AVP Code
                     0x0000 is reserved and MUST NOT be assigned to a
                     capability.

   AVP Length: The 2 octet AVP length field indicates the number of
                     octets in this AVP, including the AVP Code, AVP
                     Length, Reserved, Lifetime, and Data fields.

   Reserved: Initialized to zero, ignored on receipt.

   Lifetime: Specifies the lifetime of the encoded capability in
                     seconds. In the case of a static capability, the
                     Lifetime field MUST be set to the maximum value
                     (0xffff), which indicates that the lifetime of this
                     capability parameter never expires. A lifetime value
                     of 0x0000 deletes a capability entry.

   Data: This variable-length field has the Value of the
                   capability attribute encoded.

   Because an AVP Code of 0x0 is reserved, it can be used by the sub-
   option list parsing to determine when the end of a list of
   Capabilities has been reached and where the sub-option padding
   starts. AVPs themselves are not zero padded.

   Note: This document provides no detailed information on how to encode
   the individual capability attribute values, which is to be encoded in
   the Data field. Details on the interpretation of individual
   capability parameters are out of the scope of this document.

5.2. CARD Inter-Access Router Messages

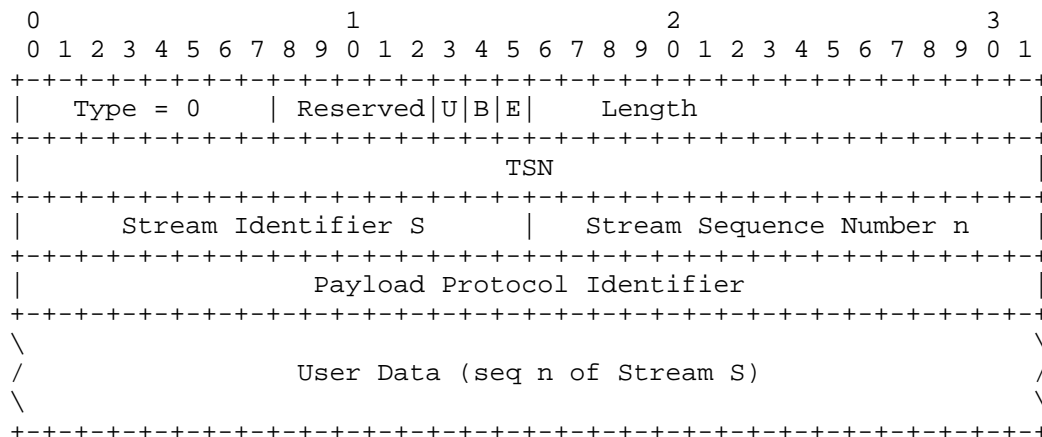5.2.1. AR-AR Transport

   Because the types of access networks in which CARD might be useful
   are not currently deployed or, if they have been deployed, have not
   been extensively measured, it is difficult to know whether congestion
   will be a problem for inter-router CARD. Part of the research task
   in preparing CARD for consideration as a candidate for possible
   standardization is to quantify this issue. However, in order to
   avoid potential interference with production applications (should a
   prototype CARD deployment involve running over the public Internet),
   it seems prudent to recommend a default transport protocol that
   accommodates congestion.

   This suggests that implementations of CARD MUST support and that
   prototype deployments of CARD SHOULD use the Stream Control Transport
   Protocol (SCTP) [Stew00] as the transport protocol between routers,
   especially if deployment over the public Internet is contemplated.
   SCTP supports congestion control, fragmentation, and partial
   retransmission based on a programmable retransmission timer. SCTP
   also supports many advanced and complex features, such as multiple
   streams and multiple IP addresses for failover, that are not
   necessary for experimental implementation and prototype deployment of
   CARD. The use of these SCTP features for CARD is not recommended at
   this time.

   The SCTP Payload Data Chunk carries the CARD messages. CARD messages
   on the inter-router interface consist of just the CARD Request or
   CARD Reply options. The User Data part of each SCTP message contains
   the CARD option for the message type. For instance, a CARD Reply
   message is constructed by including the CARD Reply option and all the
   appropriate sub-options within the User Data part of an SCTP message.

A single stream is used for CARD with in-sequence delivery of SCTP
messages. Each message, unless fragmented, corresponds to a single
CARD query or response. Unsolicited CARD Reply messages can also be
sent to peers to notify them of changes in network configuration or
capabilities. A single stream provides simplicity. Use of multiple
streams to prevent head-of-line blocking is for future study. Since
timeliness is not an issue with inter-router CARD, and since there
being more than one CARD transaction between two routers active at
any one time is unlikely, having ordered delivery simplifies the
implementation. The Payload Protocol Identifier in the SCTP header
is 'CARD'. CARD uses the Seamoby SCTP port number [Ke04].

The format of Payload Data Chunk taken from [Stew00] is shown in the
following diagram.

```
 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
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|   Type = 0    | Reserved|U|B|E|           Length              |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                              TSN                              |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|      Stream Identifier S      |   Stream Sequence Number n    |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                  Payload Protocol Identifier                  |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
\                                                               \
/                 User Data (seq n of Stream S)                 /
\                                                               \
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

        'U' bit   The Unordered bit. MUST be set to 0 (zero).
        'B' bit   The Beginning fragment bit. See [Stew00].

        'E' bit   The Ending fragment bit. See [Stew00].

        TSN   Transmission Sequence Number. See [Stew00].

        Stream Identifier S
                     Identifies the CARD stream.

        Stream Sequence Number n
                     Sequence number. See [Stew00].

        Payload Protocol Identifier
                     Set to 'CARD'.

        User Data   Contains the CARD message.

In order to avoid generating congestion on startup, ARs MUST wait a
random amount of time between 0 and CARD_STARTUP_WAIT seconds upon
reboot before sending an AR-AR CARD Request to one of its CARs. An
AR that receives a CARD Request from another AR that is not in its
CAR table MUST NOT solicit the AR but rather MUST wait until the AR
sends an unsolicited CARD Reply advertising the AR's information. An
AR that is starting up MUST send unsolicited CARD Replies to all its
CARs to make sure that their CAR tables are properly populated.

The frequency of unsolicited CARD Reply messages MUST be strictly
limited to CARD_MIN_UPDATE_INTERVAL, in order to avoid overwhelming
CARs with traffic. ARs are free to discard messages that arrive more
frequently.

If a CARD deployment will never run over the public Internet, and if
it is known that congestion is not a problem in the access network,
alternative transport protocols MAY be appropriate vehicles for
experimentation. Implementations of CARD MAY support UDP for such
purposes. In that case, the researcher MUST be careful to
accommodate good Internet transport protocol engineering practices,
such as using retransmits with exponential backoff. In addition,
whether SCTP is an appropriate transport protocol for all inter-
router CARD operations is an open research question. Investigation
of this issue (for example, to determine whether a lighter-weight
protocol might be more appropriate than SCTP) may be of interest to
some researchers.

5.2.2. Protocol Payload Types

The AR-AR interface MUST insert the CARD Request option and CARD
Reply option directly into the body of the SCTP User Data field. The
sequence number for the CARD Request on the AR-AR interface MUST be
initialized to zero when the AR reboots, and MUST be incremented
every time a CARD Request message is sent. The replying AR MUST
include a sequence number from the CARD Request in the CARD Reply.
If an unsolicited CARD Reply is sent, the sending AR MUST increment
the sequence number. Sequentially increasing sequence numbers allows
the receiving AR to determine whether the information has already
been received.

On the AR-AR interface, the Capability Container parameter is used to
convey capabilities between ARs. Optionally, the Preferences
parameter can be used for capability pre-filtering during the inter-
AR capability discovery procedure. Payload types and encoding rules
are the same as those described for the respective sub-option types
in Section 5.1 for the MN-AR interface. The same TLV-encoded format
is used to attach the options as payload to the protocol main header.
Additionally, an AR can set the T flag in the CARD Request header in

order to obtain the certificates for the CAR. The description of
sub-options in Section 5.1.3 includes information on what flag
settings are prohibited on the AR-AR interface.

6. Security Considerations

6.1. Veracity of CARD Information

The veracity of the CARD protocol depends on the ability of an AR to
obtain accurate information about geographically neighboring ARs, and
to provide accurate information about its own APs and capabilities to
other ARs. The CARD protocol described in the body of this document
does not contain any support for determining the AR-to-AP mapping or
capabilities, either for a specific AR or for a CAR. Therefore,
methods for determining the accuracy of the information exchanged
between ARs are out of scope for the base CARD protocol. The
appendices of this document describe procedures for discovering the
identities of the geographically adjacent ARs and APs (including
capabilities) and discuss relevant security considerations.
Alternatively, this information could be statically configured into
the AR.

6.2. Security Association between AR and AR

CARD contains support allowing ARs to exchange capability
information. If this protocol is not protected from modification, a
malicious attacker can modify the information. Also, if the
information is delivered in plain text, a third party can read it.

To prevent the information from being compromised, the CARD messages
between ARs MUST be authenticated. The messages also SHOULD be
encrypted for privacy of the information, if required.
Confidentiality might be required if the traffic between two ARs in
an operator's network traversed the public Internet, for example.

Two ARs engaging in the CARD protocol MUST use IKE [HarCar98] to
negotiate an IPsec ESP security association for message
authentication. If confidentiality is desired, the two ARs MUST
additionally negotiate an ESP security association for encryption.
Replay protection SHOULD also be enabled with IKE. To protect CARD
protocol messages between ARs, IPsec ESP [AtKe98] MUST be used with a
non-null integrity protection and origin authentication algorithm and
SHOULD be used with a non-null encryption algorithm for protecting
the confidentiality of the CARD information.

An AR can provide the certificates for its CARs if the certificates
are available. The AR requests certificates from its CARs by setting
the T flag in the CARD Request message. All certificates are sent.

   If CARD is used to exchange information between different
   administrative domains, additional security policy issues may apply.
   Such issues are out of the scope of this document. Use of CARD
   between administrative domains is not recommended at this time, until
   the policy issues involved are more thoroughly understood.

6.3. Security Association between AR and MN

   A malicious node can send bogus CARD Reply messages to MNs by
   masquerading as the AR. The MN MUST authenticate the CARD Reply
   messages from the AR. Since establishing an IPSec security
   association between the MN and AR is likely to be a performance
   issue, IKE is not an appropriate mechanism for setting up the
   security association. Instead, the SEND security association is used
   [Arkko04]. ARs MUST include a SEND Signature Option on CARD Reply
   messages. The format of the signature option is the same for both
   IPv4 and IPv6 CARD, though SEND itself is only defined for IPv6. A
   Mobile IPv4 ICMP Foreign Agent Advertisement option type code for the
   SEND signature option [Ke04] has been allocated.

   No authentication is required for CARD Requests since CARD
   information is provided by the AR to optimize link access. In
   contrast, CARD Reply authentication is required because a bogus AR
   could provide the MN with CARD information that would lead the MN to
   handover to a bogus router, which could steal traffic or propagate a
   denial of service attack on the MN. The asymmetry of the
   authentication requirement is the same as that involving Router
   Advertisements in IPv6 router discovery [Arkko04].

   Since CARD is a discovery protocol, confidentiality is not generally
   necessary on the MN-AR interface. In specific cases where different
   network operators share the same access network infrastructure,
   network operators may want to hide information about operator-
   specific capabilities for business reasons. The base CARD protocol
   contains no support for such cases. However, should such a case
   arise in the future, an AVP for an encrypted capability can be
   defined at that time.

6.4. Router Certificate Exchange

   Because SEND is only available in IPv6, the procedures for obtaining
   certificates differ depending on whether CARD is used with IPv4 or
   IPv6. In IPv6, when the MN receives a CARD reply with signature from
   an AR for which it does not have a certificate, it SHOULD use SEND
   DCS/DCA to obtain the AR's certificate chain. ARs MUST be configured
   with a certification path for this purpose, and MNs MUST be
   configured with a set of certificates for shared trusted anchors to
   allow verification of the AR certificates. An MN may not necessarily

need to use Cryptographically Generated Addresses (CGAs) with CARD,
so CGA support is OPTIONAL for CARD. A certificate profile for ARs
is described in the SEND specification [Arkko04].

In IPv4, there is no DCS/DCA message for obtaining the certificate.
If the MN does not have a certificate for the AR, the MN sends a CARD
Request message containing the L2 ID of its current AP and one
Trusted Anchor sub-option (Section 5.1.3.6) for each shared trusted
anchor for which the MN has a certificate, to obtain the
certification path for the current AR. The Component field of the
Trusted Anchor sub-option is set to 65535 to indicate that the entire
certification path is needed. No other options should be included in
the request. The AR replies by sending a CARD Reply containing the
L2 ID sub-option sent in the request, an Address sub-option for
itself, and a Router Certificate sub-option (Section 5.1.3.7)
containing one certificate in its certification path that matches one
of the requested trust anchors, and no other sub-options, setting the
Context-ID of all sub-options to match. The All Components field is
set to the path length, and the Component field is set to the number
of this component in the path. If the path is longer than one
certificate, the AR sends the L2 ID sub-option and the Address sub-
option in the first certificate and the other certificates in
separate ICMP messages, due to the limitation on ICMP message length,
with the same Context-ID set on each Route Certificate sub-option,
and with the Component field properly set. The router SHOULD NOT
send the trusted anchor's certificate and SHOULD send certificates in
order from the certificate after the trusted anchor. If the trusted
anchor option does not match any certificate, the AR returns the
Trusted Anchor sub-options in the reply. The MN SHOULD immediately
conduct a Certificate Revocation List (CRL) check on any certificates
obtained through CARD certificate exchange, to make sure that the
certificates are still valid.

Certification paths for CARs may be fetched in advance of handover by
requesting them as part of the CARD protocol. In that case, the MN
includes Trusted Anchor sub-options in the CARD request along with
the L2 ID sub-option for the AP for which the CAR certificate is
desired, and the AR replies as above, except that the L2 ID, address,
and certificates are for the CAR instead of for the AR itself. This
allows the MN to skip the DCS/DCA or CARD certificate exchange when
it moves to a new router.

Because the amount of space in an ICMP message is limited, the router
certification paths SHOULD be kept short.

6.5. DoS Attack

   An AR can be overwhelmed with CARD Request messages. The AR SHOULD
   implement a rate-limiting policy so that it does not send or process
   more than a certain number of messages per period. The following is
   a suggested rate limiting policy. If the number of CARD messages
   exceeds CARD_REQUEST_RATE, the AR SHOULD begin to drop messages
   randomly until the rate is reduced. MNs SHOULD avoid sending
   messages more frequently than CARD_REQUEST_RATE. ARs SHOULD also
   avoid sending unsolicited CARD Replies or CARD Requests more
   frequently than CARD_MIN_UPDATE_INTERVAL, but, in this case, the
   existence of an IPsec security association ensures that messages from
   unknown entities will be discarded immediately during IPsec
   processing.

   MNs MUST discard CARD Replies for which there is no outstanding CARD
   Request, as indicated by the sequence number.

6.6. Replay Attacks

   To protect against replay attacks on the AR-AR interface, ARs SHOULD
   enable replay protection when negotiating the IPsec security
   association using IKE.

   On the MN-AR interface, the MN MUST discard any CARD Replies for
   which there is no outstanding request, as determined by the sequence
   number. For ARs, an attacker can replay a previous request from an
   MN, but the attack is without serious consequence because the MN
   ignores the reply in any case.

7. Protocol Constants

Constant	Section	Default Value	Meaning
CARD_REQUEST_RETRY	5.1.1	2 seconds	Wait interval before initial retransmit on MN-AR interface.
CARD_RETRY_MAX	5.1.1	15 seconds	Give up on retry on MN-AR interface.
CARD_STARTUP_WAIT	5.2.1	1-3 seconds	Maximum startup wait for an AR before performing AR-AR CARD.
CARD_MIN_UPDATE_INTERVAL	5.2.1	60 seconds	Minimum AR-AR update interval.

```
     CARD_REQUEST_RATE           6.5      2 requests/  Maximum number of
                                          sec.         messages before
                                                       AR institutes rate
                                                       limiting.
```

8. IANA Considerations

    See [Ke04] for instructions on IANA allocation.

9. Normative References

    [Brad97]   Bradner, S., "Key words for use in RFCs to Indicate
               Requirement Levels", BCP 14, RFC 2119, March 1997.

    [Stew00]   Stewart, R., Xie, Q., Morneault, K., Sharp, C.,
               Schwarzbauer, H., Taylor, T., Rytina, I., Kalla, M.,
               Zhang, L., and V. Paxson, "Stream Control Transmission
               Protocol", RFC 2960, October 2000.

    [AtKe98]   Kent, S. and R. Atkinson, "IP Encapsulating Security
               Payload (ESP)", RFC 2406, November 1998.

    [HarCar98] Harkins, D. and D. Carrel, "The Internet Key Exchange
               (IKE)", RFC 2409, November 1998.

    [Arkko04] Arkko, J., Kempf, J., Zill, B., and P. Nikander, "SEcure
               Neighbor Discovery (SEND)", RFC 3971, March 2005.

    [Ke04]   Kempf, J., "Instructions for Seamoby and Experimental
               Mobility Protocol IANA Allocations", RFC 4065, July 2005.

10. Informative References

    [TKCK02]   Trossen, D., Krishanmurthi, G. Chaskar, H., Kempf, J.,
               "Issues in candidate access router discovery for seamless
               IP-level handoffs", Work in Progress.

    [MaKo03]   Manner, J. and M. Kojo, "Mobility Related Terminology",
               RFC 3753, June 2004.

    [Kood03]   Koodli, R., Ed., "Fast Handovers for Mobile IPv6", RFC
               4068, July 2005.

    [Funa02]   Funato, D., et al., "Geographically Adjacent Access Router
               Discovery Protocol", Work in Progress.

   [Tros03]   Trossen, D., et al., "A Dynamic Protocol for Candidate
              Access-Router Discovery", Work in Progress.

   [ShGi00]   Shim, E. and R. Gitlin, "Fast Handoff Using Neighbor
              Information", Work in Progress.

   [Malk03]   El Malki, K., et al., "Low Latency Handoffs in Mobile
              IPv4", Work in Progress.

11. Contributors

   The authors would like to thank Vijay Devarapalli (Nokia) and Henrik
   Petander (Helsinki University of Technology) for formally reviewing
   the protocol specification document and providing valuable comments
   and input for technical discussions. The authors would also like to
   thank James Kempf for reviewing and for providing a lot of valuable
   comments and editing help.

12. Acknowledgements

   The authors would like to thank (in alphabetical order) Dirk Trossen,
   Govind Krishnamurthi, James Kempf, Madjid Nakhjiri, Pete McCann,
   Rajeev Koodli, Robert C. Chalmers, and other members of the Seamoby
   WG for their valuable comments on the previous versions of the
   document, as well as for the general CARD-related discussion and
   feedback. In addition, the authors would like to thank Erik Nordmark
   for providing valuable insight about the piggybacking of CARD options
   upon Fast Mobile IPv6 messages.

Appendix A.   Maintenance of Address Mapping Tables in Access Routers

   This appendix provides information on two optional CAR table
   maintenance schemes for reverse address mapping in access routers.
   These schemes replace static configuration of the AP L2 ID-to-CAR IP
   address mapping in the CAR table. Details on these mechanisms are
   out of the scope of this document. The intention of this appendix is
   to provide only a basic idea on flexible extensions to the CARD
   protocol, as described in this document.

Appendix A.1.   Centralized Approach Using a Server Functional Entity

   The centralized approach performs CARD over the MN-AR interface as
   described in Section 4 of this document. Additionally, the
   centralized approach introduces a new entity, the CARD server, to
   assist the current AR in performing reverse address translation. The
   centralized approach requires that neighboring ARs register with the
   CARD server to populate the reverse address translation table. The
   registration of AR addresses with the CARD server is performed prior
   to initiation of any reverse address translation request.

   Figure A.1 illustrates a typical scenario of the centralized CARD
   operation. In this example, ARs have registered their address
   information with a CARD server in advance. When an MN discovers the
   L2 ID of APs during L2 scanning, it passes one or more L2 IDs to its
   current AR, and the AR resolves them to the IP address of the AR.
   For this, the AR first checks whether the mapping information is
   locally available in its CAR table. If it is not, the MN's current
   AR queries a CARD server with the L2 ID. In response, the CARD
   server returns the IP address of the CAR to the current AR. Then,
   the current AR directly contacts the respective CAR and performs
   capability discovery with it. The current AR then passes the IP
   address of the CAR and associated capabilities to the MN. The
   current AR then stores the resolved IP address within its local CAR
   table. The centralized CARD protocol operation introduces additional
   signaling messages, which are exchanged between the MN's current AR
   and the CARD server. The signaling messages between an AR and the
   CARD server function are shown with the preceding identifier "AR-
   Server", referring to the associated interface.

   An initial idea of performing reverse address translation using a
   centralized server is described in [Funa02].

```
                              +----------+
            +------------>|   CARD   |<------------+
            |+-----------|  Server  |------------+|
            ||              +----------+           ||
            ||                                      ||
            ||            ~~~~~~~~~~~~               ||
            ||  (4)AR-Server{          }            ||
  (3)AR-Server||          {          }            ||(0) CARD
     CARD   ||     CARD   {          }            ||Reg Req/
   Request  ||     Reply  {  IP Cloud }           | Reply
            ||              {          }            ||
            ||              {          }            ||
            ||              {_____}            ||
            |V            ~~~~~~~~~~~~             V|
     +---------+   (5)AR-AR CARD Request   +-----+-----+
     | Current |------------------------->| CAR | CAR |
     |   AR    |<-------------------------| 1   | 2   |
     +---------+   (6)AR-AR CARD Reply     +-----+-----+
        ^  |                                 |     |
  (2)MN-AR |  |(7)MN-AR                       |     |
     CARD  |  |  CARD                         |     |
   Request|  V  Reply                    +---+ +---+
     +--------------+   (1) AP1 L2 ID  +--|AP1| |AP2|
     |    Mobile    |<------------------+  +---+ +---+
     |    Node      |<------------------------------+
     +--------------+   (1) AP2 L2 ID
```
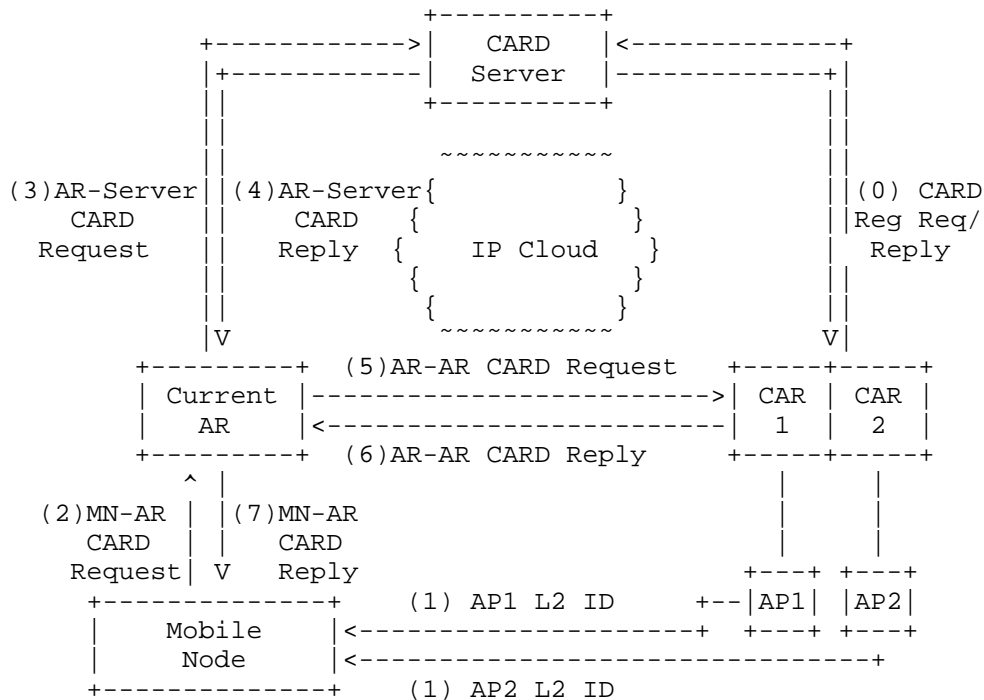
              Figure A.1: Centralized Approach for L2-L3 Mapping

Appendix A.2.   Decentralized Approach Using Mobile Terminals'
                Handover

   This approach performs CARD over the MN-AR interface as described in
   Section 4. However, it employs one additional message, called the
   Router Identity message, over the MN-AR interface to enable ARs to
   learn about the reverse address translation tables of their
   neighboring ARs, without being dependent on any centralized server.

   In this approach, CAR identities in the CAR table of an AR are
   maintained as soft state. The entries for CARs are removed from the
   CAR table if they are not refreshed before the timeout period expires
   and are created or refreshed according to the following mechanism.

   The key idea behind the decentralized approach is to bootstrap and
   maintain the association between two ARs as neighbors of each other
   using the actual handover of MNs occurring between them as input.
   The first handover between any two neighboring ARs serves as the
   bootstrap handover to invoke the discovery procedure, and the
   subsequent handover serves to refresh the association between the
   neighboring ARs. After the bootstrap handover, the MNs can perform

CARD and thus seamless handover using the CAR information. This idea
was presented in [ShGi00] and [Tros03].

Maintenance of the CAR table is done by using an additional option
for the CARD protocol operation performed between an MN and its
current AR. This message serves as Router Identity message.

Upon the completion of an inter-AR handover, the MN SHOULD send a
Router Identity message to its current AR. This message contains the
identity (IP address) of the previous AR (pAR), and can be sent as a
specific sub-option in the MN-AR CARD Request message. It SHOULD be
acknowledged with the MN-AR CARD Reply. The Router Identity message
enables the MN's current AR to learn that the pAR (still) has an AP
whose coverage overlaps with one of the APs of the current AR, and
vice versa. With this information, the MN's current AR can create or
refresh an entry for the pAR as its neighbor. If handover is no
longer possible between two ARs, the associated entries eventually
timeout and are removed from each AR's CAR table.

Prior to trusting the MN's report, however, the current AR may
perform a number of checks to ensure the validity of the received
information. One simple method is to verify the accuracy of the
Router Identity message by sending an AR-AR CARD Request message to
the pAR. The AR-AR CARD Request includes the identity of the MN.
Upon receiving this message, the pAR verifies that the MN was indeed
attached to it during a reasonable past interval and responds to the
current AR. In this way, each handover of a MN results in a bi-
directional discovery process between the two participating ARs.

Upon receiving a positive verification response, the current AR
creates or refreshes, as applicable, the entry for the pAR in its
local CAR table. In the former case, the current AR and the pAR
exchange capabilities using the AR-AR CARD Request and AR-AR CARD
Reply protocol messages. When a new entry is created, the ARs MUST
exchange their reverse address translation tables. They may exchange
other capabilities at this time or may defer exchange to a later time
when some MN undergoing handover between them performs CARD as
described in Section 4. In the latter (refresh) case, ARs may
exchange capabilities or defer exchanges until a later time when
another MN undergoes handover.

Finally, note that in a handover-based protocol, a first handover
between a pAR and an MN's current AR cannot use CARD, as this
handover bootstraps the CAR table. However, in the long term, such a
handover will only amount to a small fraction of total successful
handover between the two ARs. Also, if the MN engaging in such a
first handover is running a non-delay sensitive application at the
time of handover, the user may not even realize its impact.

Appendix B. Application Scenarios

   This section provides two examples of application scenarios for CARD
   protocol operation. One scenario describes a CARD protocol operation
   in a Mobile IPv6 (MIPv6) network, providing access to the
   infrastructure via wireless LAN Access Points and associated Access
   Routers. A second scenario describes CARD protocol operation in a
   Mobile IPv6-enabled network, which has enhanced support for fast
   handover integrated (Fast Mobile IPv6), also providing wireless LAN
   access to the infrastructure.

   This application scenario assumes a moving MN having access to the
   infrastructure through wireless LAN (IEEE802.11) APs. Mobility
   management is performed using the Mobile IPv6 protocol. The
   following figure illustrates the assumed access network design.

Appendix B.1. CARD Operation in a Mobile IPv6-Enabled Wireless LAN
               Network

```
            ----------------------------
          /                              \   +----+
          |           NETWORK            |---| HA |
          \                              /   +----+
            ----------------------------
              |                    |
          +-----+              +-----+
          | AR1 |---------+     | AR2 |
          +-----+         |     +-----+
             |  subnet 1  |        |subnet 2
          +-----+    +-----+    +-----+
          | AP1 |    | AP2 |    | AP3 |
          +-----+    +-----+    +-----+
             ^          ^          ^
              \
               \
                \
                 v
            +-----+
            | MN  | - - ->>>- - - ->>>
            +-----+
```
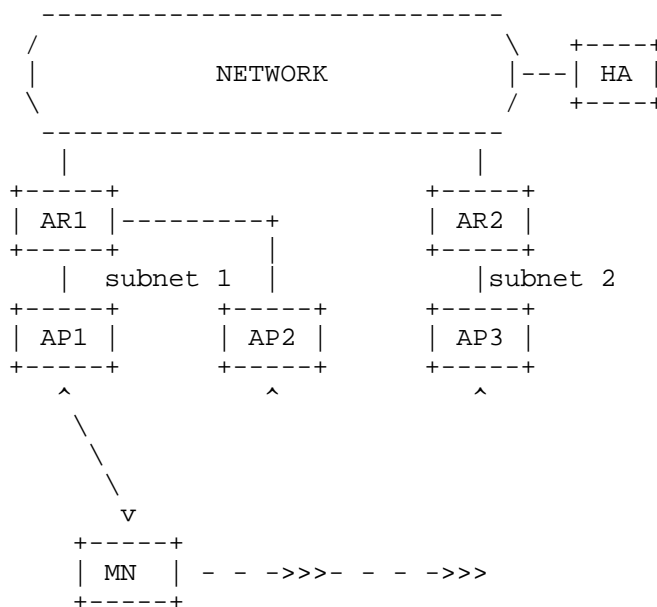
                Figure B.1: Assumed Network Topology

   A Mobile IPv6 Home Agent (HA) maintains location information for the
   MN in its binding cache. In Figure B.1, the MN holds a care-of
   address for the subnet 1, supported by AR1. As the MN moves, the
   MN's current environment offers two further wireless LAN APs with
   increasing link-quality as candidate APs for a handover. To

facilitate decision making, parameters associated with ARs are taken
into account during the decision process. The AR-related parameters
can be, for example, available QoS resources or the type of access
technologies supported from an AR. To learn about these candidate
ARs' capabilities and associated IP address information, the MN
performs CARD. This requires retrieving information about candidate
APs' L2 IDs. Furthermore, associated link-quality parameters are
retrieved to ascertain whether approaching APs are eligible
candidates for a handover. If AP2 and AP3 are suitable candidate
APs, the MN encapsulates both L2 IDs (AP2 and AP3) into a CARD
Request message, using the L2 ID sub-option, and sends the message to
its current AR (AR1).

AR1 resolves each L2 ID listed in L2 ID options to the associated IP
address of the respective CAR, making use of its local CAR table.
According to the environment illustrated in Figure B.1, the
associated AR IP address of the candidate AP2 will be the same as the
MN is currently attached to, which is AR1. The corresponding IP
address of the candidate AR, to which AP3 is connected, is the
address of AR2. IP addresses of the MN's CARs are now known to AR1,
which retrieves the CARs' capabilities from the CAR table. Assuming
that it has valid entries for respective capability parameters to
refresh dynamic capabilities, whose associated lifetimes in AR1's CAR
table have expired, AR1 performs Inter-AR CARD for capability
discovery. Since capability information for AR1 is known to AR1, a
respective Inter-AR CARD Request is sent only to AR2. In response,
AR2 sends a CARD Reply message back to AR1, encapsulating the
requested capability parameters with the signaling message in a
Capability Container sub-option.

Next, AR1 sends its own capabilities and the dynamically discovered
ones of AR2 back to the MN via a CARD Reply message. Furthermore,
AR1 stores the capability parameters of AR2 with the associated
lifetimes in its local CAR table.

Upon receipt of the CARD Reply message, the MN performs target AR
selection, taking AR1's and AR2's capability parameters and
associated APs' link-quality parameters into account. If the
selected AP is AP2, no IP handover needs to be performed. If AP3 and
the associated AR2 are selected, the MN needs to perform an IP
handover according to the Mobile IPv6 protocol operation.

Figure B.2 illustrates the signaling flow of the previously described
application scenario of CARD within a Mobile IPv6-enabled network.

```
        MN            AP1          AR1       AP2           AP3                    AR2
        |             |            |          |             |                      |
        |  connected  |            |          |             |                      |
        0-------------0-------0    |          |             |                      |
        |             |            |          |             |                      |
        |             |            |          |             |                      |
        |             |            |          |             |                      |
        |  <~~~~~~~~~L2-SCAN (AP2)~~~~~|       |             |                      |
        |  <~~~~~~~~~L2-SCAN (AP3)~~~~~~~~~~~~~~~~~~~~|       |                      |
        |             |            |          |             |                      |
        | (MN-AR) CARD Req         |          |             |                      |
        |-------------------->|               |             |                      |
        |             |            |   (AR-AR) CARD Req      |                      |
        |             |            |----------------------------------------------->
        |             |            |   (AR-AR) CARD Repl     |                      |
        | (MN-AR) CARD Repl        |<-----------------------------------------------|
        |<-------------------|     |          |             |                      |
        |             |            |          |             |                      |
[target AR           |            |          |             |                      |
selection]           |            |          |             |                      |
        |             |            |          |             |                      |
       //            //           //         //            //                     //
[either...]          |            |          |             |                      |
        |             |            |          |             |                      |
        |-------- L2 attach -------->|         |             |                      |
        |             |            |          |             |                      |
        |        connected         |          |             |                      |
        0--------------------0-------0         |             |                      |
        |             |            |          |             |                      |
       //            //           //         //            //                     //
[... or]             |            |          |             |                      |
        |             |            |          |             |                      |
        |-------------- L2 attach -------------->|           |                      |
        |             |            |          |             |                      |
        |        connected         |          |             |                      |
        0------------------------------------------0--------------------0
        |             |            |          |             |                      |
        |             |            |          |             |                      |
        |    MIPv6 Binding Update to the HA    |             |                      |
        |------------------------------------------------ - - - >                  |
        |             |            |          |             |                      |
```
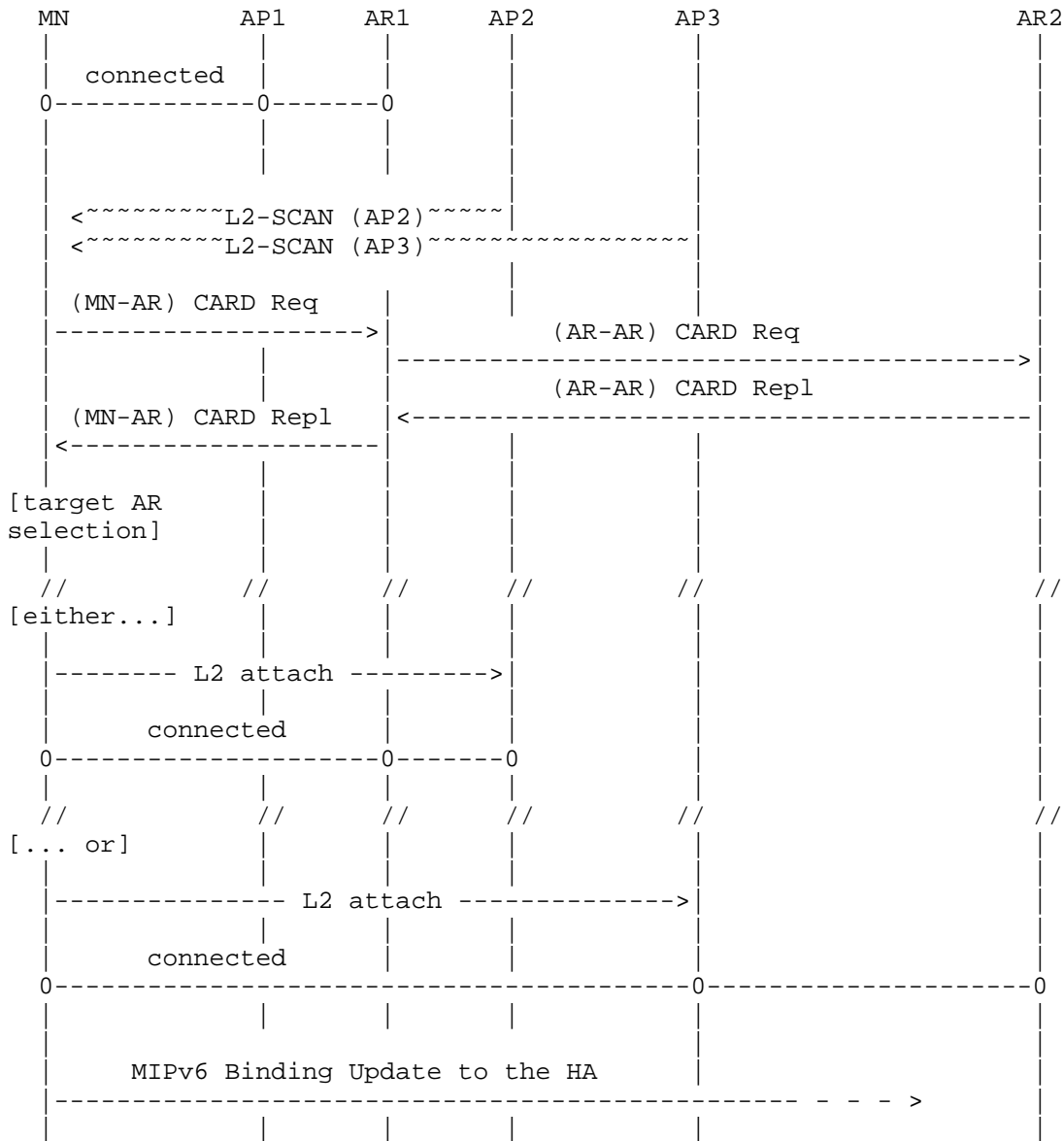
   Figure B.2. CARD Protocol Operation within a Mobile IPv6-Enabled
              Wireless LAN Network

Appendix B.2. CARD Operation in a Fast Mobile IPv6 Network

   This application scenario assumes that ARs can perform the fast
   handover protocol sequence for Mobile IPv6 [Kood03]. The MN scans
   for new APs for handover, similar to Figure B.1. To discover the ARs
   (CARs), the MN attaches a MN-AR CARD Request option to the ICMP-type
   Fast Mobile IPv6 RtSolPr message, which is sent to the MN's current
   AR (pAR, previous AR).

   Candidate APs' L2 IDs are encapsulated using the CARD protocol's L2
   ID sub-options, which allow the MN to send multiple L2 IDs of
   candidate APs to its current AR. (This potentially replaces the "New
   Attachment Point Link-Layer Address" option of the Fast Mobile IPv6
   protocol.)

   The pAR resolves the received list of candidate APs' L2 IDs to the IP
   addresses of associated CARs. The pAR checks its local CAR table to
   retrieve information about the CARs' capabilities. If any table
   entries have expired, the pAR acquires this CAR's capabilities by
   sending an AR-AR CARD Request to the respective CAR. The CAR replies
   with an AR-AR CARD Reply message, encapsulating all capabilities in a
   Capability Container sub-option and attaching them to the CARD Reply
   option. On receipt of the CARs' capability information, the pAR
   updates its local CAR table and forwards the address and capability
   information to the MN by attaching a MN-AR CARD Reply option to the
   Fast Mobile IPv6 PrRtAdv message. When the MN's handover is
   imminent, the MN selects its new AR and the associated new AP from
   the discovered list of CARs. According to the Fast Mobile IPv6
   protocol, the MN notifies the pAR of the selected new AR with the
   Fast Binding Update (F-BU) message, allowing the pAR to perform a
   fast handover according to the Fast Mobile IPv6 protocol.

   Optionally, the pAR could perform selection of an appropriate new AR
   on behalf of the MN after the pAR has the MN's CARs' addresses and
   associated capabilities available. The MN must send its requirements
   for the selection process to its pAR together with the MN-AR CARD
   Request message After the pAR has selected the MN's new AR, the
   address and associated capabilities of the chosen new AR are sent to
   the MN with the CARD Reply option in the Fast Mobile IPv6 PrRtAdv
   message.

Figure B.3 illustrates how CARD protocol messages and functions work
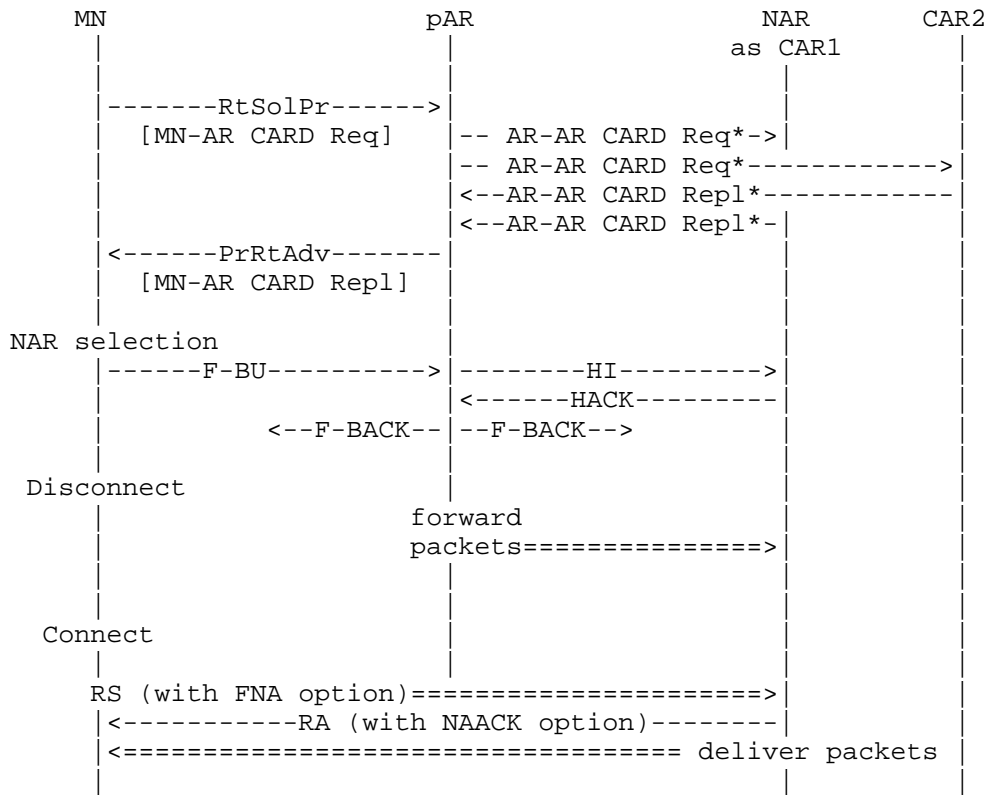with the Fast Mobile IPv6 protocol.

```
         MN                      pAR                 NAR        CAR2
          |                       |                as CAR1       |
          |                       |                   |         |
          |-------RtSolPr------->|                   |         |
          |    [MN-AR CARD Req]  |-- AR-AR CARD Req*->|         |
          |                      |-- AR-AR CARD Req*------------>|
          |                      |<--AR-AR CARD Repl*------------|
          |                      |<--AR-AR CARD Repl*-|         |
          |<------PrRtAdv-------|                   |         |
          |    [MN-AR CARD Repl] |                   |         |
          |                      |                   |         |
      NAR selection              |                   |         |
          |------F-BU---------->|--------HI--------->|         |
          |                      |<------HACK---------|         |
          |          <--F-BACK--|--F-BACK-->         |         |
          |                      |                   |         |
     Disconnect                  |                   |         |
          |               forward |                   |         |
          |               packets===============>|         |
          |                      |                   |         |
          |                      |                   |         |
      Connect                    |                   |         |
          |                      |                   |         |
       RS (with FNA option)=====================>|         |
          |<-----------RA (with NAACK option)--------|         |
          |<=================================== deliver packets |
          |                      |                   |         |
```

           Figure B.3. Fast Handover Protocol Sequence with
                       CARD Protocol Options

    * In Figure B.3, the CARD protocol interaction between the pAR and
      CARs is only required if the lifetime of any capability entries in
      the pAR's CAR table have expired. Otherwise, the pAR can respond
      to the requesting MN immediately after retrieving the CARs'
      addresses and capability information from its CAR table.

Authors' Addresses

   Hemant Chaskar
   AirTight Networks
   339 N. Bernardo Avenue
   Mountain View, CA 94043, USA

   EMail: hemant.chaskar@airtightnetworks.net

   Daichi Funato
   NTT DoCoMo, Inc.
   Communication Systems Laboratory
   Wireless Laboratories
   3-5, Hikarinooka, Yokosuka,
   Kanagawa 239-8536, Japan

   Phone: +81-46-840-3921
   EMail: funato@mlab.yrp.nttdocomo.co.jp

   Marco Liebsch
   NEC Network Laboratories
   Kurfuersten-Anlage 36,
   69115 Heidelberg, Germany

   Phone: +49 6221-90511-46
   EMail: marco.liebsch@netlab.nec.de

   Eunsoo Shim
   Panasonic Digital Networking Laboratory
   Panasonic Corporation
   Two Research Way
   Princeton, NJ 08540

   Phone: +1-609-734-7354
   EMail: eunsoo@research.panasonic.com

   Ajoy Singh
   Motorola Inc
   2G11, 1501 West Shure Dr.
   Arlington Heights, IL 60004, USA

   Phone: +1 847-632-6941
   EMail: asingh1@email.mot.com