Network Working Group                                    Yakov Rekhter
INTERNET-DRAFT                                              Mark Stapp
                                                         Cisco Systems

                                                            June 1999
                                                              Expires
December 1999

## Interaction between DHCP and DNS
### <draft-ietf-dhc-dhcp-dns-10.txt>

Status of this Memo

   This document is an Internet-Draft and is in full conformance with
   all provisions of Section 10 of RFC2026.

   Internet-Drafts are working documents of the Internet Engineering
   Task Force (IETF), its areas, and its working groups.  Note that
   other groups may also distribute working documents as Internet-
   Drafts.

   Internet-Drafts are draft documents valid for a maximum of six months
   and may be updated, replaced, or obsoleted by other documents at any
   time.  It is inappropriate to use Internet-Drafts as reference
   material or to cite them other than as "work in progress."

   The list of current Internet-Drafts can be accessed at
   http://www.ietf.org/ietf/1id-abstracts.txt

   The list of Internet-Draft Shadow Directories can be accessed at
   http://www.ietf.org/shadow.html.

Abstract

   DHCP provides a powerful mechanism for IP host configuration.
   However, the configuration capability provided by DHCP does not
   include updating DNS, and specifically updating the name to address
   and address to name mappings maintained in the DNS.

   This document specifies how DHCP clients and servers should use the
   Dynamic DNS Updates mechanism in [RFC2136] to update the DNS name to
   address and address to name mappings so that the mappings for DHCP
   clients will be consistent with the IP addresses that the clients

acquire via DHCP.

## 1.  Terminology

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in RFC 2119 [RFC2119].

## 2.  Interaction between DHCP and DNS

DNS [RFC1034, RFC1035] maintains (among other things) the information about mapping between hosts' Fully Qualified Domain Names (FQDNs) [RFC1594] and IP addresses assigned to the hosts. The information is maintained in two types of Resource Records (RRs): A and PTR. The A RR contains a mapping from an FQDN to an IP address; the PTR RR contains a mapping from an IP address to a FQDN.  The Dynamic DNS Updates specification [RFC2136] describes a mechanism that enables DNS information to be updated over a network.

DHCP [RFC2131] provides a mechanism by which a host (a DHCP client) could acquire certain configuration information, and specifically its IP address(es). However, DHCP does not provide any mechanisms to update the DNS RRs that contain the information about mapping between the host's FQDN and its IP address(es) (A and PTR RRs). Thus the information maintained by DNS for a DHCP client may be incorrect - a host (the client) could acquire its address by using DHCP, but the A RR for the host's FQDN wouldn't reflect the address that the host acquired, and the PTR RR for the acquired address wouldn't reflect the host's FQDN.

The Dynamic DNS Update protocol can be used to maintain consistency between the information stored in the A and PTR RRs and the actual address assignment done via DHCP. When a host with a particular FQDN acquires its IP address via DHCP, the A RR associated with the host's FQDN would be updated (by using the Dynamic DNS Updates protocol) to reflect the new address. Likewise, when an IP address gets assigned to a host with a particular FQDN, the PTR RR associated with this address would be updated (using the Dynamic DNS Updates protocol) to reflect the new FQDN.

Although this document refers to the A and PTR DNS record types and to DHCP assignment of IPv4 addresses, the same procedures and requirements should apply for updates to the analogous RR types that are used when clients are assigned IPv6 addresses via DHCPv6.

## [3](). Models of operation

When a DHCP client acquires a new address, both the A RR (for the client's FQDN) and the PTR RR (for the acquired address) have to be updated. Therefore, two separate Dynamic DNS Update transactions occur. Acquiring an address via DHCP involves two entities: a DHCP client and a DHCP server. In principle each of these entities could perform none, one, or both of the transactions. However, upon reflection one could realize that not all permutations make sense.  This document covers the possible design permutations:

    (1) DHCP client updates the A RR, DHCP server updates the PTR RR

    (2) DHCP server updates both the A and the PTR RRs

One could observe that the only difference between these two cases is whether the FQDN to IP address mapping is updated by a DHCP client or by a DHCP server. The IP address to FQDN mapping is updated by a DHCP server in both cases.

The reason these two are important, while others are unlikely, has to do with authority over the respective DNS domain names. A client may be given authority over mapping its own A RRs, or that authority may be restricted to a server to prevent the client from listing arbi- trary addresses or associating its address with arbitrary domain names. In all cases, the only reasonable place for the authority over the PTR RRs associated with the address is in the DHCP server that allocates them.

In any case, whether a site permits all, some, or no DHCP servers and clients to perform DNS updates into the zones which it controls is entirely a matter of local administrative policy. This document does not require any specific administrative policy, and does not propose one. The range of possible policies is very broad, from sites where only the DHCP servers have been given credentials that the DNS servers will accept, to sites where each individual DHCP client has been configured with credentials which allow the client to modify its own domain name. Compliant implementations will support some or all of these possibilities.

This document describes a new DHCP option which a client can use to convey all or part of its domain name to a DHCP server. Site-specific policy determines whether DHCP servers use the names that clients offer or not, and what DHCP servers should do in cases where clients do not supply domain names.

**3.1**.  **Client FQDN Option**

   To update the IP address to FQDN mapping a DHCP server needs to know
   the FQDN of the client to which the server leases the address. To
   allow the client to convey its FQDN to the server this document
   defines a new DHCP option, called "Client FQDN". The FQDN Option also
   contains Flags and RCode fields which DHCP servers can use to convey
   information about DNS updates to clients.

   Clients MAY send the FQDN option, setting appropriate Flags values,
   in both their DISCOVER and REQUEST messages. If a client sends the
   FQDN option in its DISCOVER message, it MUST send the option in sub-
   sequent REQUEST messages.

   The code for this option is 81. Its minimum length is 4.

```
    Code   Len    Flags  RCODE1 RCODE2   Domain Name
   +------+------+------+------+------+------+--
   |  81  |  n   |      |      |      |      ...
   +------+------+------+------+------+------+--
```

**3.1.1**.  **The Flags Field**

   This figure presents the format of the Flags field, which is one byte
   long:

```
    0 1 2 3 4 5 6 7
   +-+-+-+-+-+-+-+-+
   |   MBZ   |E|O|S|
   +-+-+-+-+-+-+-+-+
```

   When a client sends the FQDN option in its DHCPDISCOVER and/or
   DHCPREQUEST messages, it sets the right-most bit (labelled "S") to
   indicate that it will not perform any Dynamic DNS updates, and that
   it expects the DHCP server to perform any FQDN-to-IP (the A RR) DNS
   update on its behalf. If this bit is clear, the client indicates that
   it intends to perform its own FQDN-to-IP mapping update.

   If a DHCP server intends to take responsibility for the A RR update
   whether or not the client sending the FQDN option has set the "S"
   bit, it sets both the "O" bit and the "S" bit, and sends the FQDN

   option in its corresponding DHCPOFFER and/or DHCPACK messages.

   The data in the Domain Name field may appear in one of two formats:
   ASCII, or DNS-style binary encoding (without compression, of course).
   A client which sends the FQDN option sets the "E" bit to indicate
   that the data in the Domain Name field is DNS-encoded, as described
   in [RFC1035].

   The remaining bits in the Flags field are reserved for future assign-
   ment. DHCP clients and servers which send the FQDN option MUST set
   the MBZ bits to 0, and they MUST ignore values in the part of the
   field labelled "MBZ".

### 3.1.2.  The RCODE Fields

   The RCODE1 and RCODE2 fields are used by a DHCP server to indicate to
   a DHCP client the Response Code from the an A RR Dynamic DNS Update
   performed on the client's behalf. The server also uses these fields
   to indicate whether it has attempted such an update before sending
   the DHCPACK message. Each of these fields is one byte long.

### 3.1.3.  The Domain Name Field

   The Domain Name part of the option carries the FQDN of a DHCP client.
   A client may be configured with a fully-qualified domain name, or
   with a partial name that is not fully-qualified. If a client knows
   only part of its name, it MAY send a single label, indicating that it
   knows part of the name but does not necessarily know the zone in
   which the name is to be embedded. The data in the Domain Name field
   may appear in one of two formats: ASCII (with no terminating NULL),
   or DNS encoding as specified in [RFC1035]. If the DHCP client wishes
   to use DNS encoding, it MUST set the third-from-rightmost bit in the
   Flags field (the "E" bit); if it uses ASCII encoding, it must clear
   that Flags bit.

   A DHCP client that can only send a single label using ASCII encoding
   includes a series of ASCII characters in the Domain Name field,
   excluding the "." (dot) character. The client SHOULD follow the
   character-set recommendations of [RFC1034] and [RFC1035]. A client
   using DNS encoding sends a single label as a single byte count, fol-
   lowed by that number of bytes of data, without a terminal reference
   to the root.

### 3.2.  DHCP Client behavior

   The following describes the behavior of a DHCP client that implements

the Client FQDN option.

If a client that owns/maintains its own FQDN wants to be responsible for updating the FQDN to IP address mapping for the FQDN and address(es) used by the client, then the client MUST include the Client FQDN option in the DHCPREQUEST message originated by the client. A DHCP client MAY choose to include the Client FQDN option in its DISCOVER messages as well as its REQUEST messages. The rightmost bit in the Flags field in the option MUST be set to 0. Once the client's DHCP configuration is completed (the client receives a DHCPACK message, and successfully completes a final check on the parameters passed in the message), the client SHOULD originate an update for the A RR (associated with the client's FQDN). The update MUST be originated following the procedures described in section 3.4. If the DHCP server from which the client is requesting a lease includes the FQDN option in its ACK message, and if the server sets both the "S" and the "O" bits in the option's Flags field, the client MUST NOT initiate an update for the name in the Domain Name field.

A client can choose to delegate the responsibility for updating the FQDN to IP address mapping for the FQDN and address(es) used by the client to the server.  In order to inform the server of this choice, the client SHOULD include the Client FQDN option in its DHCPREQUEST message. The rightmost (or "S") bit in the Flags field in the option MUST be set to 1. A client which delegates this responsibility MUST NOT attempt to perform a Dynamic DNS update for the name in the Domain Name field of the FQDN option. The client MAY supply an FQDN in the Client FQDN option, or it MAY supply a single label (the most-specific label), or it MAY leave that field empty as a signal to the server to generate an FQDN for the client in any manner the server chooses.

Since there is a possibility that the DHCP server may be configured to complete or replace a domain name that the client was configured to send, the client might find it useful to send the FQDN option in its DISCOVER messages. If the DHCP server returns different Domain Name data in its OFFER message, the client could use that data in performing its own eventual A RR update, or in forming the FQDN option that it sends in its REQUEST message. There is no requirement that the client send identical FQDN option data in its DISCOVER and REQUEST messages. In particular, if a client has sent the FQDN option to its server, and the configuration of the client changes so that its notion of its domain name changes, it should send the new data in an FQDN option when it communicates with the server again. This may allow the DHCP server to update the name associated with the PTR record, and, if the server updated the A record representing the client, to delete that record and attempt an update for the client's current domain name.

A client which delegates the responsibility for updating the FQDN to
IP address mapping to a server might not receive any indication
(either positive or negative) about the status of the update from the
server. The client MAY use a DNS query to check whether the mapping
is updated.

A client MUST set the RCODE1 and RCODE2 fields in the Client FQDN
option to 0 when sending the option.

If a client releases its lease prior to the lease expiration time and
the client is responsible for updating its A RR(s), the client SHOULD
delete the A RR (following the procedures described in [RFC2136])
associated with the leased address before sending a DHCPRELEASE mes-
sage. Similarly, if a client was responsible for updating its A RR,
but is unable to renew its lease, the client SHOULD attempt to delete
the A RR before its lease expires. A client which has not been able
to delete an A RR which it added (because it has lost its IP address)
SHOULD add an entry to its logfile and/or notify its administrator.

## 3.3. DHCP Server behavior

When a server receives a DHCPREQUEST message from a client, if the
message contains the Client FQDN option, and the server replies to
the message with a DHCPACK message, the server SHOULD originate an
update for the PTR RR associated with the address leased to the
client if the server is configured to perform DNS updates. The update
MUST be originated following the procedures described in Section 3.4.
The server MAY complete the update before the server sends the
DHCPACK message to the client. In this case the RCODE from the update
[RFC2136] MUST be carried to the client in the RCODE1 field of the
Client FQDN option in the DHCPACK message. Alternatively, the server
MAY send the DHCPACK message to the client without waiting for the
update to be completed.  In this case the RCODE1 field of the Client
FQDN option in the DHCPACK message MUST be set to 255.  The choice
between the two alternatives is entirely determined by the configura-
tion of the DHCP server. Servers SHOULD support both configuration
options.

In addition, if the Client FQDN option carried in the DHCPREQUEST
message has the "S" bit in its Flags field set, then the server MAY
originate an update for the A RR (associated with the FQDN carried in
the option) if it is configured to do so by the site's administrator,
and if it has the necessary credentials. The server MAY be configured
to use the name supplied by the client, or it MAY be configured to
modify the supplied name, or substitute a different name.

The update MUST be originated following the procedures described in
Section 3.4. The server MAY originate the update before the server

sends the DHCPACK message to the client. In this case the RCODE from
the update [RFC2136] MUST be carried to the client in the RCODE2
field of the Client FQDN option in the DHCPACK message.  Alterna-
tively the server MAY send the DHCPACK message to the client without
waiting for the update to be completed. In this case the RCODE2 field
of the Client FQDN option in the DHCPACK message MUST be set to 255.
The choice between the two alternatives is entirely up to the DHCP
server. In either case, if the server intends to perform the DNS
update and the client's REQUEST message included the FQDN option, the
server SHOULD include the FQDN option in its ACK message, and MUST
set the "S" bit in the option's Flags field.

Even if the Client FQDN option carried in the DHCPREQUEST message has
the "S" bit its Flags field clear (indicating that the client wants
to update the A RR), the server MAY, be configured byt the local
administrator to update the A RR on the client's behalf. A server
which is configured to override the client's preference SHOULD
include an FQDN option in its ACK message, and MUST set both the "O"
and "S" bits in the FQDN option's Flags field. The update MUST be
originated following the procedures described in Section 3.4. The
server MAY originate the update before the server sends the DHCPACK
message to the client. In this case the RCODE from the update
[RFC2136] MUST be carried to the client in the RCODE2 field of the
Client FQDN option in the DHCPACK message. Alternatively, the server
MAY send the DHCPACK message to the client without waiting for the
update to be completed. In this case the RCODE2 field of the Client
FQDN option in the DHCPACK message MUST be set to 255. Whether the
DNS update occurs before or after the DHCPACK is sent is entirely up
to the DHCP server's configuration.

When a server receives a DHCPREQUEST message from a client, and the
message contains the Client FQDN option, the server MUST ignore the
values carried in the RCODE1 and RCODE2 fields of the option.

When a DHCP server sends the Client FQDN option to a client in the
DHCPACK message, the server MUST copy the Domain Name field from the
Client FQDN option that the client sent to the server in the DHCPRE-
QUEST message. If, however, the client sent only a single label, or
if the DHCP server has been configured to assign the client a name
different from the one the client has sent, the server SHOULD send
its notion of the complete FQDN for the client. The server MUST use
the same encoding format (ASCII or DNS-encoding) that the client used
in the FQDN option in its DHCPREQUEST, and MUST set the "E" bit in
the option's Flags field accordingly.

If the DHCPREQUEST message received by a DHCP server from a DHCP
client doesn't carry the Client FQDN option (e.g., the client doesn't
implement the Client FQDN option), and the DHCP client acquires its

FQDN from a DHCP server (as part of a normal DHCP transaction), then
the server MAY be configured to update both A and PTR RRs. Any
updates MUST be originated following the procedures described in Sec-
tion 3.4. In this case, the server MAY NOT wish to return the FQDN
option to a client which may not be able to understand it. If it can,
the DHCP server MAY (optionally) return the host part of the domain
name that it will use for the client in the host-name DHCP option
(defined in [RFC2132]). Note that it may not be possible to con-
sistently encode some domain name data in the format specified by the
host-name option.

If a server detects that a lease on an address that the server leases
to a client has expired or has been released by the client, the
server SHOULD delete the PTR RR which it associated with the address
via DNS Dynamic Update. In addition, if the server added an A RR on
behalf of the client, the server SHOULD also delete the A RR. The
deletion MUST follow the procedures described in Section 3.4.

If a server terminates a lease on an address prior to the lease's
expiration time, for instance by sending a DHCPNAK to a client, the
server SHOULD delete the PTR RR which it associated with the address
via DNS Dynamic Update. In addition, if the server took responsibil-
ity for the client's A RR , the server SHOULD also delete that A RR.
The deletion MUST follow the procedures described in Section 3.4.

## 3.4.  Procedures for performing DNS updates

There are two principal issues that need to be addressed concerning A
RR DNS updates:

   o Name Collisions

      If the entity updating the A RR (either the DHCP client or DHCP
      server) attempts to perform a DNS update to a domain name that
      has an A RR which is already in use by a different DHCP client,
      what should be done?  Similarly, should a DHCP client or server
      update a domain name which has an A RR that has been configured
      by an administrator?  In either of these cases, the domain name
      in question would either have an additional A RR, or would have
      its original A RR replaced by the new record. Either of these
      effects may be considered undesirable in some sites. This
      specification describes behavior designed to prevent these
      undesirable effects, and requires that implementations make this
      behavior the default.  In some scenarios these name collisions
      are unlikely, in some scenarios they are very likely:

         1.  Client updates A RR, uses DNSSEC: Name collisions in this
         scenario are unlikely (though not impossible), since for the

client to use DNSSEC, it has already received credentials
specific to the name it desires to use.  This implies that
the name has already been allocated (through some
implementation- or organization-specific procedure, and
presumably uniquely) to that client.

2.  Client updates A RR, uses some form of TSIG: Name colli-
sions in this scenario are possible, since the credentials
necessary for the client to update DNS are not necessarily
name-specific.  Thus, for the client to be attempting to
update a unique name requires the existence of some adminis-
trative procedure to ensure client configuration with unique
names.

3.  Server updates the A RR, uses a name for the client which
is known to the server: Name collisions in this scenario are
likely unless prevented by the server's name configuration
procedures.  See Section 5 for security issues with this form
of deployment.

4.  Server updates the A RR, uses a name supplied by the
client: Name collisions in this scenario are highly likely,
even with administrative procedures designed to prevent them.
(This scenario is a popular one in real-world deployments in
many types of organizations.)  See Section 5 for security
issues with this type of deployment.

Scenarios 3 and 4 are much more attractive given some form of
DHCP Authentication, but the difficulties remain.

Scenarios 2, 3, and 4 rely on administrative procedures to
ensure name uniqueness for DNS updates, and these procedures may
break down.  Experience has shown that, in fact, these pro-
cedures will break down at least occasionally.  The question is
what to do when these procedures break down or, for example in
scenario #4, may not even exist.

In all cases of name collisions, the desire is to offer two
modes of operation to the administrator of the combined DHCP-DNS
capability:  first-update-wins (i.e., the first updating entity
gets the name) or most-recent-update-wins (i.e., the last updat-
ing entity for a name gets the name).

o Multiple DHCP servers

If multiple DHCP servers are able to update the same DNS zones,
or if DHCP servers are performing A RR updates on behalf of DHCP
clients, and more than one DHCP server may be able to serve

addresses to the same population of DHCP clients, the DHCP
servers should be able to provide reasonable DNS name update
behavior for DHCP clients.

The solution to both of these problems is for the updating entities
(either DHCP clients or DHCP servers) to be able to cooperate when
updating DNS A RRs.

Specifically, a KEY RR, described in [RFC2535] is used to associate
client ownership information with a DNS name and the A RR associated
with that name.  When either a client or server adds an A RR for a
client, it also adds a KEY RR which specifies a unique client iden-
tity (based on a "client specifier" created from the client's
client-id or MAC address: see Appendix A).  In this model, only one A
RR is associated with a given DNS name at a time.

By associating this ownership information with each A RR, cooperating
DNS updating entities may determine whether their client is the first
or last updater of the name (and implement the appropriately config-
ured administrative policy), and DHCP clients which currently have a
host name may move from one DHCP server to another without losing
their DNS name.

See Appendix A for the details of the use of the KEY RR for this pur-
pose.

The specific algorithms utilizing the KEY RR to signal client owner-
ship are explained below.  The algorithms only work in the case where
the updating entities all cooperate -- this approach is advisory only
and does not substitute for DNS security, nor is it replaced by DNS
security.

3.4.1.  Adding A RRs to DNS

When a DHCP client or server intends to update an A RR, it first
prepares a DNS UPDATE query which includes as a prerequisite the
assertion that the name does not exist.  The update section of the
query attempts to add the new name and its IP address mapping and the
KEY RR with its unique client-identity.

If this update operation succeeds, the updater can conclude that it
has added a new name whose only RRs are the A and KEY RR records.
The A RR update is now complete (and a client updater is finished,
while a server would then proceed to perform a PTR RR update).

If the first update operation fails with YXDOMAIN, the updater can
conclude that the intended name is in use.  The updater then attempts
to confirm that the DNS name is not being used by some other host.

The updater prepares a second UPDATE query in which the prerequisite
is that the desired name has attached to it a KEY RR whose contents
match the client identity (see Appendix A).  The update section of
this query deletes the existing A records on the name, and adds the A
record that matches the DHCP binding and the KEY RR with the client
identity.

If this query succeeds, the updater can conclude that the current
client was the last user of this name, and that the name now contains
the updated A RR. The A RR update is now complete (and a client
updater is finished, while a server would then proceed to perform a
PTR RR update).

If the second query fails with NXRRSET, the updater must conclude
that the client's desired name is in use by another host.  At this
juncture, the updater can decide (based on some administrative confi-
guration outside of the scope of this document) whether to let the
existing owner of the name keep that name, and to (possibly) perform
some name disambiguation operation on behalf of the current client,
or to 'take-over' the name on behalf of the current client.

DISCUSSION:

   The updating entity may be configured to allow the existing owner
   to keep the name, and to perform disambiguation on the name of the
   current client in order to attempt to generate a similar but
   unique name for the current client.  In this case, once such a
   similar name has been generated, the updating entity should res-
   tart the process of adding an A RR as specified in this section.

### 3.4.2.  Adding PTR RR Entries to DNS

The DHCP server submits a DNS query which deletes all of the PTR RRs
associated with the lease IP address, and adds a PTR RR whose data is
the client's (possibly disambiguated) host name. The server also adds
a KEY RR whose data is the client's client-identity as described in
Appendix A.

### 3.4.3.  Removing Entries from DNS

The first rule in removing DNS entries is be sure that an entity
removing a DNS entry is only removing an entry for which it is
responsible.

When a lease expires or a DHCP client issues a DHCPRELEASE request,
the DHCP server SHOULD delete the PTR RR that matches the DHCP bind-
ing, if one was successfully added. The server's update query SHOULD
assert that the name in the PTR record matches the name of the client

whose lease has expired or been released.

The entity chosen to handle the A record for this client (either the client or the server) SHOULD delete the A and KEY records that were added when the lease was made to the client.

In order to perform this delete, the updater prepares an UPDATE query which contains two prerequisites.  The first prerequisite asserts that the KEY RR exists whose data is the client identity described in Appendix A. The second prerequisite asserts that the data in the A RR contains the IP address of the lease that has expired or been released.

If the query's prerequisites fail, the updater MUST NOT delete the DNS name.  It may be that the host whose lease on the server has expired has moved to another network and obtained a lease from a dif-ferent server, which has caused the client's A RR to be replaced. It may also be that some other client has been configured with a name that matches the name of the DHCP client, and the administrative pol-icy at the site was that the last client to specify the name would get the name.  In this case, the KEY RR will no longer match the updater's notion of the client-identity of the host pointed to by the DNS name.

## 4.  Updating other RRs

The procedures described in this document only cover updates to the A and PTR RRs. Updating other types of RRs is outside the scope of this document.

## 5.  Security Considerations

Whether the client may be responsible for updating the FQDN to IP address mapping, or whether the this responsibility lies with the DHCP server is a site-local matter. The choice between the two alter-natives may be based on a particular security model that is used with the Dynamic DNS Update protocol (e.g., only a client may have suffi-cient credentials to perform updates to the FQDN to IP address map-ping for its FQDN).

Whether a DHCP server is always responsible for updating the FQDN to IP address mapping (in addition to updating the IP to FQDN mapping), regardless of the wishes of a DHCP client, is also a site-local matter. The choice between the two alternatives may be based on a particular security model.

The client SHOULD use some form of data origin authentication pro-
cedures (e.g. [TSIG], [DNSSEC]) when performing DNS updates.

While the DHCP client MAY be the one to update the DNS A record, in
certain specialized cases a DHCP server MAY do so instead.  In this
case, the DHCP server MUST be sure of both the name to use for the
client, as well as the identity of the client.

In the general case, both of these conditions are not satisfied --
one needs to be mindful of the possibility of MAC address spoofing in
a DHCP packet.  It is not difficult for a DHCP server to know unambi-
guously the DNS name to use for a client, but only in certain cir-
cumstances will the DHCP server know for sure the identity of the
client. If DHCP authentication [DHCPAUTH] becomes widely deployed
this may become more customary. An example of a situation which
offers some extra assurances is one where the DHCP client is con-
nected to a network through an MCNS cable modem, and the CMTS (head-
end) of the cable modem ensures that MAC address spoofing simply does
not occur.

Another example where the DHCP server would know the identity of the
client would be in a case where it was interacting with a remote
access server which encoded a client identification into the DHCP
client-id option.  In this case, the remote access server as well as
the DHCP server would be operating within a trusted environment, and
the DHCP server could trust that the user authentication and authori-
zation procedure of the remote access server was sufficient, and
would therefore trust the client identification encoded within the
DHCP client-id.

In either of these cases, a DHCP server would be able to correctly
enter the DNS A record on behalf of a client, since it would know the
name associated with a client (through some administrative procedure
outside the scope of this protocol), and it would also know the
client's identity in a secure manner.

**6**. **Appendix A - Use of the KEY RR**

The KEY RR used to hold the DHCP client's identity is formatted as
follows:

The name of the KEY RR is the name of the A or PTR RR which refers to
the client.

The flags field is set to 0x42 - that is, the 1 bit and the 6 bit are
set.

The protocol field is set to [TBD].

The algorithm field is set to [TBD].

The first byte in the key field contains the length of the client-
identity, and is followed by that number of bytes of client-identity
data. If a DHCP client sent the client-id option in its request mes-
sage, the client-identity MUST be identical to the data in the
client-id option. If a client did not send the client-id option, the
client-identity is constructed from the htype byte, the hlen byte,
and hlen bytes of the client's chaddr from its request message.

## 7.  References

[RFC1034]   P. Mockapetris, "Domain names - concepts and facilities",
            RFC1034, 11/01/1987

[RFC1035]   P. Mockapetris, "Domain names - implementation and specif-
            ication", RFC1035, 11/01/1987

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

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

[RFC1594]   A. Marine, J. Reynolds, G. Malkin, "FYI on Questions and
            Answer Answers to Commonly asked ``New Internet User''
            Questions", RFC1594, 03/11/1994

[RFC2136]   P. Vixie, S. Thomson, Y. Rekhter, J. Bound, "Dynamic
            Updates in the Domain Name System (DNS UPDATE)", RFC2136,
            April 1997

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

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

[TSIG]      P. Vixie, O. Gudmundsson, D. Eastlake, B. Wellington,
            "Secret Key Transaction Signatures for DNS", draft-ietf-
            dnsind-tsig-*, Work in Progress.

[DHCPAUTH] R. Droms, W. Arbaugh, "Authentication for DHCP Messages",
            draft-ietf-dhc-authentication-*, Work in Progress.

## 8.  Acknowledgements

Many thanks to Mark Beyer, Jim Bound, Ralph Droms, Robert Elz,
Peter Ford, Edie Gunter, R. Barr Hibbs, Kim Kinnear, Stuart Kwan,
Ted Lemon, Michael Lewis, Michael Patton, and Glenn Stump for
their review and comments.

## 9.  Author information

Yakov Rekhter
Cisco Systems, Inc.
170 Tasman Dr.
San Jose, CA 95134
Phone: (914) 235-2128
email: yakov@cisco.com

Mark Stapp
Cisco Systems, Inc.
250 Apollo Drive
Chelmsford, MA  01824
Phone: (978) 244-8498
email: mjs@cisco.com

## 10.  Full Copyright Statement

THE INFORMATION HEREIN WILL NOT INFRINGE ANY RIGHTS OR ANY IMPLIED
WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE.