IDNS: A SIMPLE APPROACH TO INTERNET HOST PORTABILITY

Yu Chen, Terrance Boult
Lehigh University
27 Memorial Drive West, Bethlehem, PA. 18015 USA
Tel: +1 610 758 4081, Fax: +1 610 758 6279, E-mail: {yuc3, tboult}@eecs.lehigh.edu

Abstract: *Current mobile networking applications can be classified into two categories --- mobility and portability. Mobility means the capability of transacting continuous network traffic and requires seamless handoff during the migration while portability means network connections be reinitialized after a migration. Although mobility support implies portability support, all existing approaches to mobile networking applications are designed mainly for mobility support and are unnecessarily complex if only portability support is required. In this paper, a simple DNS-DHCP combined protocol is proposed to support Internet host portability. It is named Incremental Domain Name System (IDNS) protocol. IDNS has significant benefits in comparison to existing approaches for Internet host portability.*

KEYWORDS: IDNS, DHCP, FQDN, Portability

INTRODUCTION

Mobile networking applications have become more and more popular due to the increased popularity of portable computer and wireless communication devices. As a result, the demand to provide Internet access for users as they move locations is growing rapidly. According to the different characteristics of the demand, the mobile networking applications can be classified into two categories [11]:

Mobility:

Requires the provision of a continuous connection between mobile hosts and their communication peers. The characteristics of this category are:

- Mobile hosts tend to migrate frequently.
- Seamless handoff and upper-layer transparency is needed.

Wireless/cellular industry will be the largest community which needs mobility support in the future.

Portability:

Requires the provision of available (but not continuous during migration) Internet access service to mobile hosts. In this category, it is expected that following characteristics present:

- Mobile hosts typically will disconnect from the Internet during its migration.
- Upper-layer transparency is not needed.

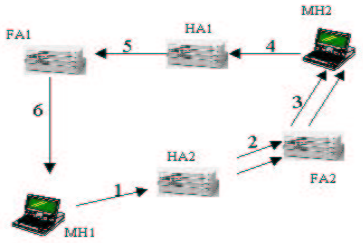
Mobile users using laptop computers with wired network connection is an example of this category. However, note that portability does bot imply wired network connection necessarily --- the host could be wireless, as long as upper layer transparency is not required by the applications.

Although existing approaches to mobility support such as IETF Mobile IP [2] can also support "portability", they are unnecessarily heavyweight for the inherently simple requirements of portability.

In this paper, we propose a simple DNS-DHCP combined approach called Incremental Domain Name System (IDNS) to provide the "portability" service in the second category. As we will outline in later section, the approach we propose in this paper allows a mobile host to change its IP address dynamically while it migrates. No IP tunneling and data encapsulation/de-capsulation is needed. Although IDNS is not likely to provide continuous connection without additional mechanism, it is good enough to portability support. Figure 1 presents the typical Mobile Host (MH) communications under Mobile IP in contrast with Figure 2 under IDNS.

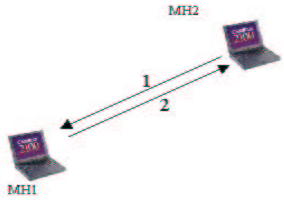
We believe in the current period and near future, most mobile hosts are likely to disconnect from the Internet when migrating due to the power limitation of mobile hosts or a lack of network connectivity. It

is expected that portability demands are considerable and outweigh mobility demands. Hence, it is necessary to separate portability service from mobility service. The remainder of the paper is organized as follows. In section 1, we give the overview of the IDNS approach and discuss relative works. In section 2, the mechanism of IDNS is presented in detail. Finally, section 3 gives conclusions.



1. MH1 performs DNS query of MH2 and get the **home** IP address of MH2, and sends packets to MH2
2. HA2 of MH2 intercepts the packets destined to MH2, encapsulates the packet and forwards it to FA2 of MH2 via IP tunneling.
3. FA2 of MH2 decapsulates the packets and forwards it to MH2.
4. MH2 sends the data packet to MH1 either according to the source IP address of packets it received or makes DNS query to get the **home** IP address of MH1.
5. HA1 of MH1 intercepts the data packet, encapsulates the packet and forwards it to FA1 of MH1 in IP tunneling.
6. FA1 of MH1 decapsulates the data packet and forwards it to MH1.

Figure 1 ***MHs' Communication under Mobile IP***



1. MH1 performs DNS query of MH2 and get the **current** IP address of MH2, and sends packets to MH2
2. MH2 sends the data packet to MH1 either according to the source IP address of packets it received or makes DNS query to get the **current** IP address of MH1.

Figure 2 ***MHs' Communication under IDNS***

The following acronyms are used in this paper. They are presented here for easy reference.

MH: Mobile Host	CH: Correspondent Host
DS: Domain Name Server	ADS: Authoritative Domain Name Server
RA: Registration Agent	NA: Negotiation Agent
RR: DNS Resource Record	RRset: Resource Record Set
FQDN: Full Qualified Domain Name	

1. OVERVIEW

We build IDNS mechanism on existing protocols: DNS [6][7] and DHCP [8]. The system is designed so that a MH can get a suitable IP address whenever needed instead of having a fixed home IP address. The IDNS mechanism has two main components: Registration and Dynamic Address Binding. Mobile users use Registration process of IDNS to register/de-register their presence and to get automatically configured. DHCP, with its dynamic configuration characteristic, is used in IDNS to allocate IP address dynamically. However, dynamic IP configuration does not enable the Mobile host (MH) to be found by Correspondent Hosts (CHs) automatically. Thus, Dynamic Address Binding is needed to ensure that CHs can get the up-to-date effective IP address of the MH. Since IP address is tightly associated with Full Qualified Domain Name (FQDN) and DNS query is typically the first step to find a route to the Internet host, the process of Dynamic Address Binding in INDS is to make the DNS mappings of MHs consistent to current IP configurations. Together with additional mechanisms, IDNS Dynamic Address Binding ensures CHs to be able to locate mobile hosts via normal DNS naming queries.

1.1 Background and Relative Works

Dynamic Host Configuration Protocol (DHCP) [8] is used to enable individual computers on an IP network to extract their configuration from a DHCP server. DHCP allows for dynamic allocation of network addresses and configurations to newly attached hosts. Additionally, DHCP allows for recovery and reallocation of network addresses through a leasing mechanism. A DHCP lease is the amount of time that the DHCP server grants to the DHCP client permission to use a particular IP address. Usually the

DHCP lease time ranges from 15 minutes to a year. If DHCP is used to support mobile users, typically the amount of users will be more than the addresses available. Thus in this situation, the DHCP lease time tends to be short in order to keep the maximum number of addresses for distribution.

Although DHCP can also be used to allocate care-of IP address in Mobile IP, Perkins recommends in [11] to use DHCP to allocate the MH a home address instead of a care-of address while still letting Mobile IP to allocate a care-of address and employing foreign agents on each subnet. Such approach can achieve better transport protocol performance during the handoff period, however, the cost is very high. The local DHCP server in the foreign network must reserve a pool of addresses for each mobile home network. It means that all the foreign networks where mobile hosts could possibly visit have to configure their DHCP server a reserved pool in advance. Such requirement may not be realistic. Trying to tunnel the DHCP request to the home DHCP server of mobile hosts turns out inefficient either. The recently proposed NAI draft [12] allows the home address to be allocated by the home network as part of the Mobile IP registration. The premise is that the foreign agent should be able to handle the NAI.

In general, although the combination of DHCP with Mobile IP could provide a scalable solution to mobility support, the cost of Mobile IP is too high for portability support when seamless handoff is not needed. Even though the home address could be efficiently allocated on the home network, employing foreign agents with DHCP server is still inefficient. Foreign agents are inherently unnecessary to the network which already deploys DHCP, especially to applications that only need portability support. The upper layer transparency provided by Mobile IP is not needed for many applications only require portability support (e.g. for web browsing).

Another problem is the management of the multiple address of the mobile host. When DHCP is used in combination with Mobile IP, the mobile host must handle packets with at least two different IP addresses. The minimum requirement is that the mobile host should receive packets delivered to its DHCP allocated care-of IP address while issue packets from its home address. Furthermore, for smooth handoff, a mobile host should continue to accept packets to its previous DHCP allocated care-of IP address while it migrates to a new point of attachment and acquires a new DHCP allocated care-of IP address. In the worst case, a mobile host whose home IP address is also DHCP allocated probably have to handle four IP addresses simultaneously: two different home addresses (one current, one previous) and two different care-of addresses. Such complexity can cause unexpected haywire.

Mobile IPv6 [9] provides more efficient mobility support. No foreign agents are needed in Mobile IPv6. IP tunneling is still needed but avoided whenever possible. It alleviates the “triangle routing” problem by enabling any IPv6 host to learn and cache the care-of address associated with a mobile host’s home address, and then to send packets destined for the mobile host directly to it at the care-of address using an IPv6 routing header. The binding cache could greatly minimize the network overhead in Mobile IPv4 and hence increases the scalability. However, the whole Mobile IPv6 approach is based on the global deployment of IPv6, which is still under development and has not been widely deployed. Furthermore, it requires each mobile host keep track of the correspondent hosts in order to send Binding Updates if necessary, which introduces extra overhead in mobile host.

To those mobile hosts that needs dynamic DHCP address allocation, the IDNS approach is a better approach which only needs allocate one IP address once. It mitigates the problem presented above since there is no difference between home address and care-of address. No data forwarding is needed and the mobile host can always obtain allocated IP address dynamically from the local DHCP server.

By conjunction DHCP to Dynamic DNS Update [3] [4], IDNS enables a mobile host to update the relevant DNS database each time it obtains a new home address and maintain its name in its new place. [15] specifies how DHCP clients and servers should use the Dynamic DNS updating mechanism to keep the DHCP allocated address consistent with the DNS mappings. A DHCID Resource Record (RR) defined in [14] can be used to associate client identification information with a DNS name and the A RR [6] associated with that name.

DHCPv6 [13] allocates IP addresses in a message extension instead of the main header. Dynamic Updates to DNS are also supported in the IPv6 address extension by setting suitable bits. DHCPv6 client-server authentication extension can provide authentication when necessary.

DRCP [17], a recently developed protocol which is an enhanced version of DHCP, is designed specifically for mobile users. DRCP minimizes the setup delay after a MH migration. Thus it is more powerful than DHCP to mobility support. However, since DHCP is enough to portability support and DRCP is still under development, IDNS uses DHCP in current design and in this paper we will focus IDNS mechanism based on DHCP only.

IDNS extends current DNS to achieve better portability support. In order to keep as small a revision

to DNS as possible, IDNS does not modify the message format of DNS but rather it extends some parameter fields. For simplicity, in this paper we only present the parts extended by IDNS. Readers should refer to DNS standards [3][4][5][6][7] for background on the underlying DNS system.

1.2 General requirements

IDNS aims to make the actual data transmission as simple as possible. IDNS assumes that each time a Correspondent Host (CH) tries to set up a connection with a Mobile Host (MH), it performs the normal DNS query and obtains the current effective IP address of the MH. Such an assumption is not suitable for mobility but is acceptable for portability. Thus a main design issue of IDNS focuses on the mechanism which keeps the Authoritative Domain Name Server (ADS) of the MH aware of the up-to-date IP address of the MH. As we have seen in Figure 2, no data forwarding or tunneling mechanism is needed in the IDNS. In the following we present the general requirements of IDNS on various network entities. Some concepts presented in this section will be clarified in detail in later sections.

Note that for mobility support, Dynamic Address Binding should be transparent to upper layers (transport layer) and applications while portability support does not have such requirement. Hence in IDNS that previous Internet connections will be closed when the MH migrates. The MH has to resume its Internet connection after it obtains a new leased IP address from the NA of the new location.

Bit 0	QR (Operation): 0 Query 1 Response
Bit 1-4	OPCODE (Operation Code): 0 QUERY (Standard Query) 1 IQUERY (Inverse Query) 2 STATUS (A server status request) 5 UPDATE (Dynamic update) **6 FORCE (Authoritative Answer Required)** **7 SETAAL (Set ADS Alert)**
Bit 5	AA (Set if answer authoritative)
Bit 6	TC (Set if message truncated)
Bit 7	RD (Set if recursion desired)
Bit 8	RA (Set if recursion available)
Bit 9-11	Z (Reserved)
Bit 12-15	RCODE (Response Code): 0 NOERROR (No error) 1 FORMERR (Format error in query) 2 SERVFAIL (Server failure) 3 NXDOMAIN (Name error) 4 NOTIMP (Not implemented) 5 REFUSED (Operation refused by server) **6 ALNOERR (ADS Alert waiting status set successfully)** **7 ALNOTIF (Alert notification)**

Table 1 *Bits of the parameter field*

- MH (Mobile Host)

IDNS does not require each MH to have a fixed permanent IP address. Instead, it requires each MH to have a FQDN. Hence, all applications employing IDNS must use FQDN instead of IP address as the unique identifier of the MH. Additionally, each MH should have a RRset (possibly empty) in its ADS database. It is expected that each MH should be provided appropriate private and public keys to present to the NA in order to generate SIG(0) RR [5] to authenticate DNS updates.

- DS (Domain Name Server)

IDNS extends the parameter field of the DNS message. To keep consistent with other recently emerging standards of DNS such as Dynamic Update of DNS [3] [4] and DNS security extensions [5], IDNS defines new OPCODEs and RCODEs based on these standards. Note the RCODE definitions of IDNS do not overlap with the RCODE definition of Dynamic Update of DNS [3]. Table 1 gives the detailed specification of parameter field of the IDNS message, with the new OPCODE and RCODE definitions in

bold.

IDNS requires that the DSs should support Dynamic DNS update standards specified in [3][4]. It is also required by INDS that if the server does not support the extensions defined by IDNS upon receiving an IDNS request message, it should signal RCODE of NOTIMP to the requestor if the OPCODE specified is not recognized or it is recognized but has not been implemented.

- RA (Registration Agent) and NA (Negotiation Agent)

A RA is an AAA server located in the home network of the MH. An AAA server is an Internet node which provides Authentication, Authorization and Accounting service to Internet hosts. IDNS requires the RA to authenticate MHs. RAs do not participate in data transmission.

In IDNS, a MH should request a Negotiation Agent (NA) to update the A RR and/or PTR RR for itself. A NA is a local DHCPv4 server which supports DHCP-DNS interaction specified in [15] and DHCP authentication [16] or a local DHCPv6 server. NA can dynamically assign IP address to MH. By building on DHCP, which can temporally multiplex IP address, the IDNS approach makes efficient utilization of IP addresses.

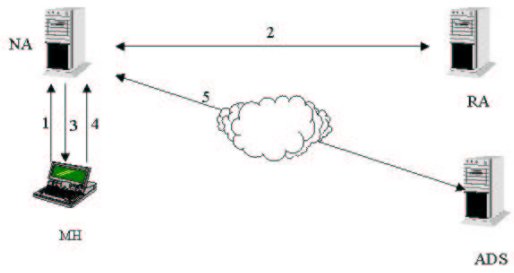
To avoid any possible inconsistency, IDNS requires all DNS updates must be performed by the NA. MHs should not perform DNS update themselves by directly issuing Dynamic DNS update messages. It is also required that NA should support secure dynamic DNS update.

- CH (Correspondent Host)

No particular requirements are needed on CH for normal DNS queries. But if CH wants to use IDNS exclusively defined mechanisms such as "FORCE" and "ALERT", which will be presented next section, it must comply with the relative IDNS specifications. Note all CHs can still use normal DNS queries to communicate with MH in IDNS even they do not support those IDNS extensions.

2. IDNS MECHANISM

The main mechanisms for IDNS include Registering/De-registering, "FORCE" query and "ALERT" mechanism, which will be described in turn. In essence, IDNS is totally FQDN driven. In Contrast to Mobile IP, IDNS also supports portable hosts where the home site is totally DHCP driven and the MH does not have a fixed IP address. In this case the following registration and de-registration processes we describe hold both at home and on the road.



1. MH sends DHCPEXTDISCOVER message to NA.
2. NA authenticates MH with RA.
3. NA sends DHCPOFFER message to MH.
4. MH sends DHCPREQUEST message to NA with "S" bit set in the FLAGS filed to request NA to perform dynamic DNS update for MH.
5. NA performs secure DNS UPDATE on ADS.

Figure 3 ***MH Registration Process***

2.1 MH Registering

MH registers in a foreign domain into which it migrates by contacting the NA. After authentication, the MH requests the NA to update the RR in its ADS. The whole process of registration is illustrated in Figure 3 and is discussed in more detail in the following sections.

2.2.1 MH Arrival.
IDNS does not require the NA to periodically broadcast service advertisement messages; the registration procedure is always originated from MH. When a MH arrives at a foreign domain and starts to connect the Internet, it broadcasts a DHCPDISCOVER message with the FQDN option (code 15 [11]) included. While all DHCP servers including the NA on the local net receives the message, only the NA will include the FQDN option in the responding DHCPOFFER message. If due to some unexpected error or policy concerns, the NA fails to allocate a suitable IP address, it should not send any messages to the MH. After a timeout period, the MH will give up on this attempt to register. After a few failed attempts the registration procedure will terminate.

2.2.2 MH authentication.
After the NA receives DHCPDISCOVER message, it is up to the administrative policy of the NA whether it should authenticate the MH before it allocates an IP address to the MH. The NA cooperates with the RA of the MH to proceed the authentication. Here the NA and the RA should comply with a same AAA protocol. Since today AAA servers identify clients by using the Network Access Identifier (NAI) [14], the MH should also include a NAI extension in DHCPDISCOVER message in case the NA requires authentication unless the FQDN could be used as a substitute of the NAI. Note the authentication here is independent to the dynamic DNS update authentication described later.

2.2.3 IP address allocation
If and when the MH receives the DHCPOFFER message from the NA, the MH proceeds to send the NA a DHCPREQUEST message to request the committed IP address. (If multiple DHCPOFFER messages are received, all but one from the NA with the FQDN option included is ignored). The MH may include the IP address lease time option in the DHCPREQUEST message. This option allows the MH to request a non-default lease time for the IP address and indicates how long the MH expects itself to remain in the foreign domain. Note such IP address allocation could be same as the normal DHCP IP address allocation or it may use a separate pool. The separate pool has the advantage of simplicity in handling those IP addresses that are released early.

2.2.4 IP address updating
In order to update the FQDN to IP address mapping, the MH should include the FQDN option in its DHCPREQUEST message. It should also set the rightmost (or “S”) bit in the FLAGS field in the option to 1 if the NA is a DHCPv4 server or set the “A” bit in the IP address extension if the NA is a DHCPv6 server. The NA then sends to the ADS of the MH an “dynamic update” DNS message with the operation set to “Query” and the query type set to "UPDATE". Such message should abide the RFC2136 [3]. Strict security can be achieved by using secure DNS UPDATE message defined in RFC2137 [4]. The MH should include the allocated IP address and the designated TTL or default TTL that is equal to ½ the DHCP lease time of the allocated IP address in the “Update Section” field of the dynamic update DNS message. The default TTL is set smaller than the lease time in order to alleviate any possible stale cache problems. The procedure of DNS updating conforms to the specification in [15]. Note here Dynamic DNS updating refers only to A RR and PTR RR in IPv4 or AAAA RR and PTR RR in IPv6. Other types of DNS RR updating are not involved in IDNS.

2.2 MH De-registering
MH (Mobile Host) de-registers with current domain whenever it migrates and updates the entry of RR (Resource Record) in its ADS (Authority domain name server). There is no different behavior of a MH in a home domain or in a foreign domain. IDNS requires a MH to perform de-registration when it releases the IP address before the lease expires. The TTL provides extra protection from stale cache problem in case MHs do not de-register consciously.

2.2.1 MH Migration
When a MH leaves its current domain before the lease of its DHCP allocated IP address expires, the MH should send an early lease termination message to the NA. DHCP already defines a DHCPRELEASE message for such early lease termination. When the NA receives the DHCPRELEASE request, the NA should send a secure dynamic update DNS message to the ADS of the MH to remove the entries that it creates before in Registration process if no inconsistency is detected. To help alleviate the stale cache

problem the released IP address should not be reused immediately until all message traffic using that IP address has cleared the network. If the MH does not send any early lease termination request, the NA is responsible to send a DNS update message to reset the TTL of the relative RRs in the ADS of the MH before it expires. The new TTL should be set as the smaller of the previous TTL and the remaining lease time. Finally, if the NA does not receive any DHCPREQUEST messages from the MH before the lease of the allocated IP address expires, it assumes the MH has left, hence frees the allocated IP address and removes any relative information of the MH in its record.

2.2.2 TTL function

The NA is NOT required to poll periodically to see if the MH is still staying in the network domain. Rather, the MH specified or the default DHCP lease time is used to ensure the expiration of its allocated IP address. Since the ADS should also have the TTL value equal to ½ the DHCP lease time of the allocated IP address in the RR of the MH, failing to receive a corresponding "UPDATE" DNS message that indicates early DHCP lease termination does not have great disadvantage.

2.3 "FORCE" Query

For IDNS to be most effective, it is important that, whenever possible, network access uses FQDN rather than IP addresses directly. The CH uses the IDNS mechanism to obtain the current IP address of MH. After that, the remaining procedure of data transmission for portable hosts is exactly same as the normal data communication between stationary hosts. Unlike Mobile IP, the IDNS mechanism does not require IP tunneling during the data transmission.

The stale cache problem occurs when a MH returns its allocated IP address early, and local DNS server of the CH has a cached copy of it. One of two possibilities exists. If the MH is now offline, the CH has no one to contact and its cached address is useless. Holding the allocated IP address until the TTL of DNS mappings expires can solve the problem. Otherwise it could be even harmful if the address was reused before the TTL expires. If the MH has already registered in a new domain before the original TTL expires, the cached address would be wrong, even though the MH is available. To handle these issues IDNS provides an option to allow the CH to ensure its address is correct for any new "connections".

2.3.1 "FORCE" Query

A DNS query with OPCODE set to "QUERY" allows the resolver to get the current IP address of the MH together with a TTL to specify how long the current IP address is effective. It is usually hard to tell whether the result of such query is up to date. A "cached" result retrieved from the cache of the resolver itself or one of the intermediate DS is probably stale. So we add an optional OPCODE of "FORCE" to indicate that the result can not be extracted from the cache, instead the result must be an authoritative answer (the AA bit is set in response). Thus the CH can set the query type field to "FORCE" in order to get the latest IP address from the mobile host's ADS if the information that it wants to transmit is highly confidential. Although the CH can directly make the "FORCE" query, usually taking such strategy indicates that the CH is aware of the mobility of its peer. Hence in general CH should not perform "FORCE" query directly. It should be used after the CH receives any returned ICMP errors. If a CH receives persistent ICMP "Host Unreachable" or "Network Unreachable" messages after sending packets to a MH using DNS mappings retrieved from a normal DNS query, it could assume the DNS mappings are probably stale and invoke the "FORCE" Query to get the most up-to-date DNS matching.

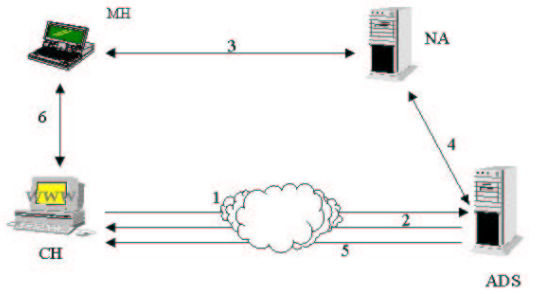
2.3.2 TTL function

The TTL is used to outdate any old DNS mappings in IDNS. As we stated previously, in the anticipated application of IDNS, the MH typically will disconnect with Internet while migrating. Thus such turn-off period should be enough to outdate any stale cache in any intermediate Domain Name Servers. Currently IDNS is not suitable to those mobile applications in which MHs tend to migrate in a high frequency. Otherwise, the "FORCE" queries will be invoked too often and make the whole performance of the DNS system drop significantly.

2.4 "ALERT" Mechanism

When a CH gets the query result, the query result could be "positive" or "negative". "Positive" means DNS has successfully retrieved the current IP address of MH while "negative" means the result is not available. There could be various reasons of getting "negative" result. One possibility is that the mobile

host has not registered in any domain at that time. If any CH performs DNS query at that time, it will get "negative" result. Under normal situation, the CH has to give up the transmission request and tries again later. IDNS gives a "ALERT" mechanism described below available to the CH when the information is very important and the CH wants the MH to receive it as soon as possible. Note such mechanism requires the CH to be able to send and receive particular IDNS messages as normal DNS messages. Figure 4 may help to clarify the "ALERT" mechanism.



1. CH performs SETAAL DNS query.
2. In case MH does not have an effective IP address, ADS appends the IP of CH to the waiting queue of MH and responds MH with ALNOERR DNS message.
3. MH registers with NA
4. NA performs secure DNS UPDATE on ADS.
5. ADS sends ALNOTIF DNS message to CH .
6. CH begins data communication with MH

Figure 4 ***Alert Mechanism***

2.4.1 Set ADS (Authoritative Domain Name Server) Alert

A CH should issue an IDNS query message with OPCODE set to "SETAAL" instead of normal DNS query message when it wants to employ the "ALERT" mechanism. The message should contain the IP address of the CH in "additional section". If an effective IP address is available in the ADS database, the IP address will appear in "Your IP address" field of DNS "Response" message. Otherwise the ADS should append the IP address of the CH to the tail of a waiting queue of the MH. The ADS should return an IDNS response message with RCODE set to "ALNOERR" if the ADS alert waiting status is properly set or return an "SERVFAIL" DNS message if it fails setting the alert waiting status. Note "SETAAL" implies the same requirement of "FORCE", that is, such query result must be authoritative, any intermediate DSs should forward the "SETAAL" message only.

All intermediate DSs are recommended not to create negative caches according to "ALNOERR" response message since any such negative caches will be soon outdated by the DNS mapping returned by the ADS of the MH in a "ALNOTIF" response message. Thus any negative cache is meaningless and even harmful.

2.4.2 Alert Notification

When a MH registers with a NA successfully and updates the DNS mappings with its ADS, the ADS checks the waiting queue of the MH. If there are any waiting CHs, it constructs a DNS "Response" message whose RCODE is "ALNOTIF" and with the AA bit set, sends to the CHs to inform them the updated IP address of MH. The CHs should be able to understand such "Response" IDNS message. The "ALNOTIF" message is sent only once even some CHs could fail to receive it. The entries of the waiting queue are deleted immediately.

2.4.3 Extension

A potential extension would be to have an ADS create a multicast group for each MH to handle the alerts. When a CH makes query of the IP address of a MH, the ADS adds the IP address of the CH into the multicast group. Thus later each time the MH updates its IP address with the ADS, the ADS can send multicast notification to each member of the multicast group. Whether it is valuable or not depends on the

scale of users supported by the ADS.

3. CONCLUSIONS

In this paper, we have presented a new protocol called IDNS to provide portability support for mobile hosts. Compared to solutions for general mobility, IDNS has the following benefits for Internet host portability support:

- It is more scalable because MHs does not rely on NA or RA to perform data communication while other approaches such as Mobile IP rely on HA and/or FA to forward datagrams to MHs. As long as a MH successfully finishes registering and obtains an effective IP address, it can perform data communication independently just like a static Internet host.
- It does not add extra network entity for exclusive IDNS mechanism. NA is an enhanced DHCP server. RA is an AAA server. DHCP servers and AAA servers are already widely used in many applications.
- It uses the FQDN to locate the MH instead of the IP address. IP addresses are used for routing purpose only. Keeping a permanent host name (FQDN) is convenient because the computer can always be reached via one name, independent of the computer's network attach point. Obviously, FQDN is more natural and efficient than a fixed home IP address to identify an Internet host.
- It does not have the problem of "triangle route optimization" which impairs the performance of the Mobile IP since no IP tunneling and data forwarding is needed in IDNS.
- It is easier to provide multiple IP addresses to multi-homed MH since NA need not be responsible for forwarding datagrams to each IP address of the MH.

A major limitation of IDNS is that, not all mobile hosts have a FQDN. We believe however, that is more of an administrative than a technical problem. It arises because ISPs often use dynamic IP allocation and there are no suitable standards for DNS to cooperate with DHCP to maintain the RRsets of these computers until recent years.

There are two possible problems with IDNS, both of which are related to DNS mapping caches in name servers. First, if the MH returns its IP address by early lease termination, the NA can reclaim that IP address. However, it should not reallocate that IP address immediately. Instead, it should keep such IP address unallocated until the TTL of DNS mappings expires. This can partly help solve the "stale cache" problem, by keeping the address out of use until the cache is purged automatically, but does cause some address depletion. Carefully choosing the TTL of the DNS RR will help optimize the utilization rate. The default TTL specified in IDNS is ½ the DHCP lease time. The second problem occurs if the CH uses a stale IP address of the MH to communicate with the MH directly instead of performing DNS query. For example this type of stale access would occur if it had an "open" TCP connection to a mobile host that changes its IP address. However, providing the continuous connection transparent to the CH requires the full power of Mobile-IP wherein the previous NA (FA in mobile IP) of the MH to forward packets destined to the MH. While it is possible to extend IDNS to have these abilities or to add IDNS functionality to mobile-IP, it is beyond the scope of this paper. The goal of current IDNS design is to provide a simpler solution, to a smaller, but very significant part portability problem. If, as is generally the case the MH is expected to disconnect from the Internet while moving, this second problem does not occur, and IDNS works without caching problems.

For the truly paranoid CHs, IDNS also provides two additional features. The first is a special "ALERT" mechanism aims to help improve the communication between CHs and any MH that is currently unavailable. The second added feature is a "FORCE" mechanism to ensure that the retrieved DNS mapping is up to date.

The power of IDNS is that it builds on existing protocols, and is based on the use of fully qualified domain names (FQDN). Most human operations use these domain names and hence IDNS is quite natural for interaction. Using the FQDN IDNS provides a simple and effective solution to Internet host portability, especially when the communication peers are both mobile hosts.

REFERENCES

[1] Charles E. Perkins, Andrew Myles, and David Johnson, "The Internet Mobile Host Protocol (IMHP)," *Proceedings of INET'94/JENC5*, pp.642, June 1994

[2] Charles E. Perkins, "Mobile IP," *IEEE Communications*, pp.84-99, May 1997

[3] P. Vixie, et. al., "Dynamic updates in the Domain Name System," *RFC2136*, April 1997

[4] D. Eastlake, "Secure Domain Name System Dynamic Update," *RFC2137*, April 1997

[5] D. Eastlake, et. al., "Domain Name System Security Extensions," *RFC2535*, March 1999

[6] P. Mockapetris, "Domain Names - Concepts and Facilities," *RFC1034*, November 1987

[7] P. Mockapetris, "Domain Names - Implementation and Specification," *RFC1035*, November 1987

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

[9] Charles E. Perkins, David B. Johnson, "Mobility Support in IPv6," *Proceedings of the Second Annual International Conference on Mobile Computing and Networking (MobiCom'96)*, November 1996

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

[11] Charles E. Perkins and Tangirala Jagannadh, "DHCP for Mobile Networking with TCP/IP," *Proceedings of IEEE Symposium on Computers and Communications 95*, 1995

[12] P. Calhoun, C. Perkins, "Mobile IP Network Access Identifier Extension for IPv4," *RFC2794*, March 2000

[13] Charles E. Perkins, Jim Bound, "DHCP for IPv6," *Proceedings of the Third IEEE Symposium on Computers and Communications*, June 1998.

[14] A. Gustafsson, "A DNS RR for encoding DHCP client identity," *Internet Draft (draft-ietf-dnsext-dhcid-rr-*)*, October 1999

[15] M. Stapp, Y,Rekhter, "Interaction between DHCP and DNS," *Internet Draft (draft-ietf-dhc-dhcp-dns-*)*, work in progress, March 2000

[16] R. Droms, W. Arbaugh, "Authentication for DHCP Messages," *Internet Draft (draft-ietf-dhc-authentication-*)* , June 1999

[17] Anthony McAuley, et. al., "Dynamic Registration and Configuration Protocol (DRCP*)," Internet Draft (draft-itsumo-drcp-*)*, October 1999