PEER-TO-PEER BASED ARCHITECTURE FOR MOBILITY MANAGEMENT IN WIRELESS NETWORKS

Shou-Chih Lo[1] and Wen-Tsuen Chen[2]

[1]Dept. of Computer Science & Information Engineering, National Dong Hwa University, Hualien, Taiwan, R.O.C.; [2]Dept. of Computer Science, National Tsing Hua University, Hsinchu, Taiwan, R.O.C.

Abstract: Mobility management is an important task in wireless networks. The Mobile IP protocol provides a basic solution to the mobility management. However, Mobile IP suffers from several problems. In this paper, we propose an enhancing version of Mobile IP by using the Peer-to-Peer (P2P) network technology. We organize home agents into P2P networks and use the Domain Name System (DNS) to provide the universal telephone number that can uniquely identify one person regardless of the type of equipped device. We claim that our proposed version can provide the advantages of update locality, scalability, load balancing, fault tolerance, and self-administration.

Key words: Mobile IP; Domain Name System; Peer-to-Peer; Wireless Networks.

1. INTRODUCTION

Mobility management in wireless networks is an important task in order to keep connectivity with roaming users at anytime. Mobile IP[1], which is a standard proposed by the Internet Engineering Task Force (IETF), can serve as the global mobility management in the future heterogeneous wireless networks[2].

Mobile IP uses the *home agent* (HA) and *foreign agent* (FA) to maintain the mobility of a mobile node (MN). The HA maintains the address binding of an MN, and the address binding is a mapping between the permanent home address to the care-of address (CoA) temporally borrowed from an FA.

Mobile IP suffers from several problems[3-5] such as the triangular routing, frequent and long distant registration updates, and single point of failures.

In this paper, we propose some mechanisms to solve these problems experienced in Mobile IP, and most of importance, we introduce the emerging technique of *Peer-to-Peer* (P2P) networks[6-8] into Mobile IP. P2P networks are overlay networks whose topologies are fully independent of physical networks. P2P networks are mostly designed for the data sharing applications. One user can publish its shared data items such as songs or pictures into the P2P network. The developed P2P lookup mechanisms enable one user to efficiently locate the desired data item in logarithmic time. Also, P2P networks with self-organizing and self-configuring features can provide load balancing and fault tolerance.

We take the address binding as a shared data item, and organize a set of HAs into a P2P network. We develop a mechanism to distribute the address binding of an MN to a selected HA with low update cost from the P2P network. We allow each system operator to organize its own P2P network and use the *Domain Name System* (DNS) to provide access to the various P2P networks. Moreover, we provide a universal identifier converted from a typical telephone number to reach a user regardless of the type of equipped device.

The rest of this paper is organized as follows. In Section 2, we give a brief survey on mobility management using Mobile IP. In Section 3, we present the design of our proposed architecture. Section 4 compares the difference and performance of a variety of approaches. Finally, we give a conclusion in Section 5.

2. RELATED WORK

Mobile IP specified a mechanism to enable an MN to change its point of attachment without changing its IP address. Both Mobile IPv4 and Mobile IPv6 are discussed in the IETF. In this paper, we explain our main idea based on Mobile IPv4. The same idea can be deployed in the IPv6 framework.

Mobile IP has the problem of frequent registration updates particularly for an MN with high mobility. The regional registration[3-5] is commonly used to reduce the registration cost. When an MN moves within the same domain (or region), the registration update is locally handled by a domain-level agent (called Gateway FA, GFA). We called this approach region-based Mobile IP.

As an MN moves far away from its permanent HA, the long distant registration update to the HA would cost high. The dynamic HA assignment becomes the potential solution to this problem. In the approachs[9,10], the GFA is used to be a temporary HA. The association of a temporary HA to an MN is recorded in DNS. In the approach[11], an FA can select a near and light loaded HA to register for an MN, and a redirection link is created between the permanent and temporary HAs. However, how to select a proper HA for an MN is not discussed.

Another problem raised in Mobile IP is the triangular routing. The straightforward solution is to bypass the HA and directly establish the connection to the currently visited FA of the MN. In the approachs[9,10,12,13], the DNS is used to support the query of address binding of any MN. The address binding is stored in DNS as a resource record, and can be refreshed by the dynamic update[14]. We call this approach DNS-based Mobile IP.

In Mobile IP, the HA or FA is sensitive to the single point of failure. The fault-tolerant issue becomes important. In the approachs[15,16], an HA or FA has some other redundant ones as its backup set. Once the HA or FA is failed, another one would be dynamically selected from its backup set.

3. PEER-TO-PEER BASED ARCHITECTURE

In this paper, we propose a P2P-based Mobile IP architecture to efficiently manage the MN's mobility. We combine the advantages of region-based and DNS-based Mobile IPs in our proposed architecture. Moreover, our design takes advantage of the load-balancing and scalability characteristics of P2P networks.

3.1 System Overview

We use subnet-level granularity to explain the basic operations of our mobility management. The detailed descriptions will be given in Section 3.5. Suppose that each subnet is associated with an FA. Several subnets would constitute a domain which is associated with a GFA. The functional overview of our proposed architecture is depicted in Fig. 1.

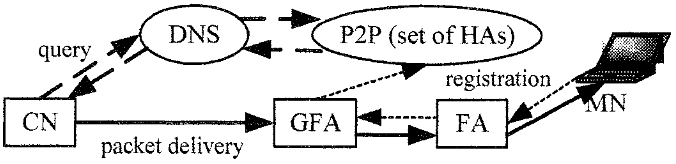

Figure 1. P2P-based Mobile IP.

To avoid the problems of single point of failure and overloaded traffic on the HA, we organize all the existing HAs into a P2P network. With the essential feature of P2P networks, one HA can freely join (when newly installed) and leave (when failed) the network. We would dynamically select an HA for an MN from the P2P network, which is close to the MN. If the selected HA is heavy loaded, we would seek another light loaded one in the neighborhood. With a little modification on the existing P2P lookup mechanism, we can efficiently locate the HA that is selected for a particular MN in the P2P network.

When an MN moves within the same domain, the registration update is locally performed to the GFA. Only whenever the MN moves to another domain, the registration update to the HA is performed. Meanwhile, we may select a new HA that is close to the MN for reducing the long distant registration update.

When a CN (Corresponding Node) would like to connect to an MN, it issues a query to DNS where the P2P lookup mechanism is triggered to locate the MN's HA. The found HA would return the location of GFA the MN is currently located in to the CN. As a result, the CN can directly establish a connection to the GFA and this connection would be further redirected to the MN.

We claim that this architecture can have the following advantages:

Update locality. The frequent registration updates due to the MN's movement of small scope will be partially localized by the regional registration technique. Moreover, the periodical registration updates to the HA during the binding renewal period would be cost saving, because we have selected a near HA to the MN.

Load balancing. The DNS does not perform the complex name resolution for an MN. Instead, the DNS only provides the entry point to the P2P network and triggers the P2P lookup mechanism to find the MN's HA. We put the burden of the complex name resolution on the P2P network where the actual execution would be distributed to the nodes involved in the P2P network. The set of HAs in the P2P network will work together and can migrate the workload with each other.

Self-administration. Each system operator can administer its own P2P network, which facilities the prevention of binding data from the revelation to other system operators. Also, a system operator can freely increase or decrease the number of HAs depending on the amount of users that are served.

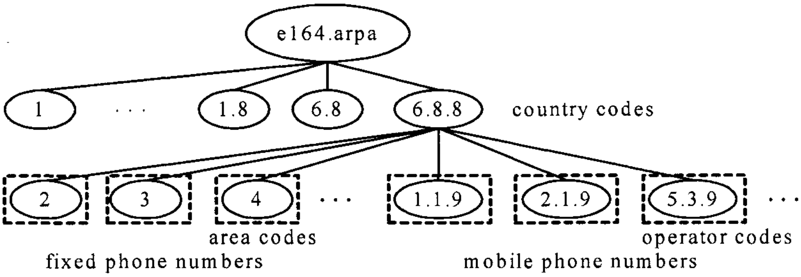

Figure 2. ENUM domain name space.

3.2 DNS Structure

Assume each MN is associated with and identified by a unique telephone number (called ENUM[17] throughout the paper). As illustrated in the IETF RFC 2916[18], a typical E.164 telephone number like +886-3-5741234 can be transformed to the domain name with format:

"4.3.2.1.4.7.5.3.6.8.8.e164.arpa".

"e164.arpa" is the suggested root of the ENUM domain names. A possible portion of the ENUM domain name space is shown in Fig. 2. The second-level domains include one entry for every country code, and the third-level domains include one entry for every area code or every operator code.

Assume the ENUM domain name space is divided into non-overlapping zones according to the ENUMs administered by different system operators. We indicate each zone in Fig. 2 by using a dotted rectangle. A zone will have one primary name server and several secondary name servers for the fault-tolerant reason.

A system operator can install a set of HAs for each of its service zones. These HAs are not necessary to be associated with network routers as done in Mobile IP, and can be artificially distributed into the service coverage of the corresponding zone. These HAs are responsible for storing the resource records of MNs having their ENUMs in that zone. The resource record might contain the IP address binding for Internet applications that need mobility support and the service binding which specifies the preferred means (e-mail, telephone, etc.) to be reached at a particular period of time.

The name server in the zone performs the name resolution, given an MN's ENUM domain name, by locating the HA which stores the MN's resource record in the P2P network. The system operator has the responsibility to keep the list of HAs for each of its service zones up to date in the corresponding name server. The recorded information might include the HA's IP address and the HA's location possibly gotten from the GPS (*Global Positioning System*). The HA list would be used in the following functions each name server would provide.

- get_host_name($ENUM\_NAME_{MN}$). This function returns the resource record of the MN with ENUM domain name $ENUM\_NAME_{MN}$. The name server when performing this function would randomly select one HA from its HA list as the entry point to the P2P network, and from there the P2P lookup mechanism is activated.
- get_neighbor_HA(IP_{target}, k). This function returns the k nearest HAs from the HA list that are close to the node with IP address IP_{target}. The actual measurement of locality is beyond the scope of the paper and GPS is one of the possibilities.

3.3 P2P Structure

The emerging P2P networks have potential to support large data sharing applications. Some system protocols, such as Pastry[6], CAN[7], and Chord[8], have been proposed for building large P2P networks. These protocols are based on a *distributed hash table* (DHT), which allows shared data items to be uniformly distributed into the nodes in the P2P network.

Each node participated in the P2P network has part of index information to shared data items. In Chord, for example, one user can issue a data lookup query to any node in the P2P network, and from there the query would be subsequently forwarded with at most log N hops till to the target node containing the desired data item. Also, Chord can efficiently support the node's join and leave with $\log^2 N$ messages.

Our goal is to construct an individual P2P network with the HAs in each zone. Within this network, the MN's address binding would be considered as a shared data item. An MN can publish its address binding (or service binding) identified by an ENUM to the P2P network. We call the node the binding data is hashed to a *destined* HA for an MN. A user can locate the destined HA of a particular MN by sending a lookup query, carrying the MN's ENUM, to the P2P network.

Note that the destined HA is not artificially selected but is determined by the DHT. The registration cost would be high if the destined HA is far away from the MN's current location. To support update locality, we artificially select a near HA called *assigned* HA to the MN, and the actual binding data is stored in the assigned HA. Since we can only locate the destined HA through the P2P lookup query, we establish a redirection link from the destined HA to the assigned HA.

Sometimes the near HA to the MN would be heavy loaded. In this case, we select another HA in the neighborhood, which is light loaded as an assigned HA. To support this function, we construct a neighbor list for each HA, which records the state (alive and heavy loaded or not) of its neighbors. We can get the *k* nearest neighbors to an HA *X* by asking the name server of the local zone via function get_neighbor_HA(IP_X, *k*).

The HA in our P2P network provides the following functions:

- locate_HA($ENUM\_NAME_{MN}$): This function locates the destined HA of the MN with ENUM domain name $ENUM\_NAME_{MN}$ by using the P2P lookup mechanism.
- redirect_HA(IP_{target}, $ENUM\_NAME_{MN}$): This function creates a redirection link from the destined HA of the MN with ENUM domain name $ENUM\_NAME_{MN}$ to the node with IP address IP_{target}.

3.4 Region Structure

We construct a GFA in each domain, and this GFA can provide a global CoA (GCoA) to a registered MN under the domain. By contrast, the FA can provide a local CoA (LCoA) to a registered MN under the subnet. Packets, which are sent from a CN and are destined to an MN, are tunneled to the GFA by the GCoA and then tunneled to the MN by the LCoA. During the Mobile IP session, the CN after querying the DNS would directly deliver packets to the GFA the MN is currently located in. If the MN makes a movement and changes to another GCoA and/or LCoA, the GFA known by the CN has the responsibility to redirect the packets to the MN.

If the movement is within the same domain, we can either establish a redirection path between the old FA and the new FA (path 1 in Fig. 3) or between the GFA and the new FA (path 2 in Fig. 3). The choice is depending on the update and packet delivery costs.

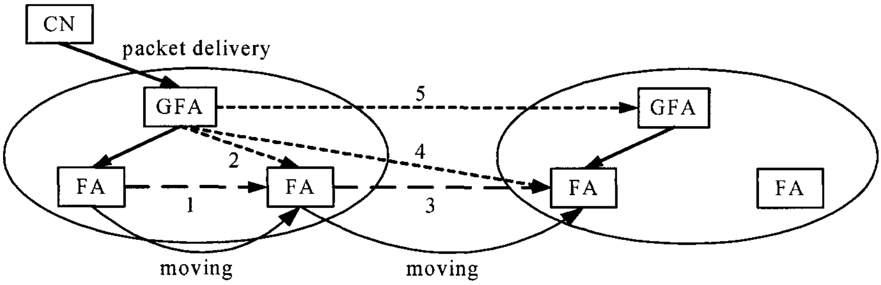

Figure 3. Possible redirection paths.

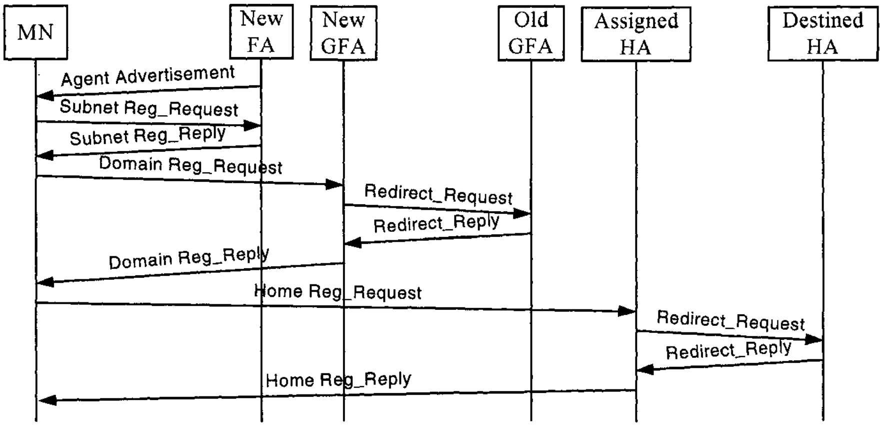

Figure 4. Registration flow during inter-domain movement.

If the movement is across different domains, we have three choices on the redirection paths. One is between the old FA and the new FA (path 3 in Fig. 3). Another is between the old GFA and the new FA (path 4 in Fig. 3). The other is between the old GFA and the new GFA (path 5 in Fig. 3) after the MN has registered to the new GFA. The choice is depending on the update and packet delivery costs and the IP address space used. The first two have to use the LCoA of global IP address space; while the last one can use the LCoA of private IP address space.

The GFA has another job of maintaining a list of available HAs, which are located in the same zone and are close to the GFA, for each of service zones. The GFA can select an HA from the corresponding list as an assigned HA for an MN coming from a certain service zone. The GFA can send the get_neighbor_HA(IP_{GFA}, k) request to the name server of a particular zone to construct this list. Those MNs with their ENUM domain names belonging to the same zone would share a common list of HAs in the GFA.

3.5 System Operations

3.5.1 Registration Update

In Fig. 4, we depict the signaling flow during registration update. Whenever the MN changes subnets within the same domain, the MN only communicates its new LCoA to the serving GFA (we use path 2 in Fig. 3). Whenever the MN changes domains, it first obtains an LCoA by performing a subnet-specific registration update to the serving FA. The serving FA assigns the MN a designated GFA. Then the MN performs a domain-specific

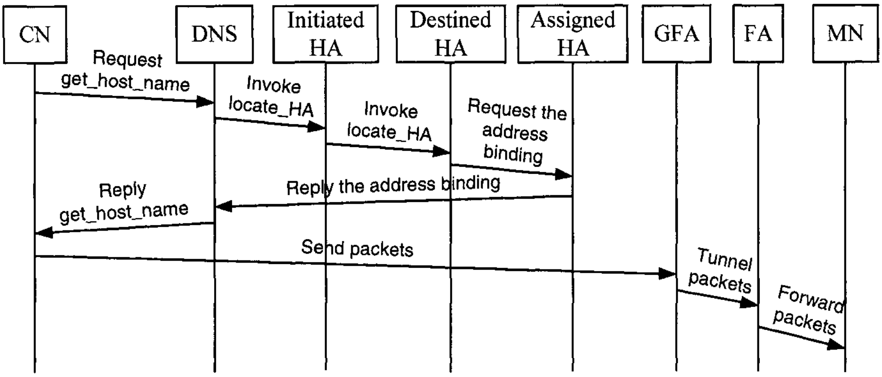

Figure 5. Packet delivery flow.

registration update by communicating its current LCoA to the designated GFA. The designated GFA replies to the registration with a GCoA and an assigned HA. If the MN has an active session, the redirection path (we use path 5 in Fig. 3) between new and old GFAs is created. Then the MN performs a home registration update by communicating its current GCoA to the assigned HA.

The assigned HA can either accept or reject the MN's registration according to its current capacity. If rejected, the assigned HA would reply the MN with another HA selected from its neighbor list. The MN would subsequently attempt to register to a different HA till accepted. If the accepted HA is different from the destined HA of the MN, the accepted HA would call the redirect_HA($IP_{assigned\_HA}$, $ENUM\_NAME_{MN}$) function. The subsequent binding renewal would be only performed to the assigned HA.

In our registration scheme, any new connection from a CN is directed to the new GFA since the address binding has been updated. Those old connections would be guaranteed to be deliverable via the redirection path.

3.5.2 Packet Delivery

In Fig. 5, we depict the signaling flow during packet delivery. The CN first sends the get_host_name($ENUM\_NAME_{MN}$) request to DNS. The DNS would contact the name server of the zone to which $ENUM\_NAME_{MN}$ belongs. The name server randomly selects an HA (called initiated HA) from the HA list to initiate the P2P lookup operation. The destined HA of the MN can be found by calling function locate_HA($ENUM\_NAME_{MN}$). Then, we follow the redirection link to reach the assigned HA and from there the address binding (i.e., MN's GCoA) is retuned to the CN. As a result, the CN establishes a connection and sends packets to the GFA, and these packets are further tunneled to the FA, and to the MN.

Table 1. Performance comparison

	Standard	Region_Based	DNS_Based	P2P_Based
Triangular Routing	Yes	Yes	No	No
Frequent Registration	Yes	No	Yes	No
Single Point of Failure	Yes	Yes	Yes	No
Load Balancing	No	No	No	Yes
Update Locality	No	Yes	No	Yes
Registration Update	2*2 hops	2*[3, 2] hops	2*2 hops	2*[3+log N, 2] hops
Connection Setup	3 hops	4 hops	2*d+2 hops	2*(d+log N+2)+3 hops
Packet Delivery	3 hops	4 hops	2 hops	3 hops

4. PERFORMANCE EVALUATION

We have mentioned three categories of enhancements to Mobile IP: region_based, DNS_based, and P2P_based ones. Here we make a comparison of these different enhancing mechanisms to the standard Mobile IP. In Table 1, we summarize the advantages/disadvantages and costs of these mechanisms. Our proposed P2P_based mechanism essentially inherits the advantages of region_based and DNS_based ones, so we have no triangular routing (due to DNS) and frequent registration update (due to regions) problems. The self-configuring characteristic of P2P networks makes our mechanism having no single point of failure on the HA.

Moreover, our proposed P2P_based mechanism has good load balancing due to the following reasons:

1. In the DNS: The domain name hierarchy of DNS can naturally distribute the workload to different name servers. Moreover, the random selection of the entry point to a P2P network from a name server can distribute the P2P lookup overhead to different nodes.
2. In the P2P network: The operations of a P2P lookup query are naturally distributed to the nodes involved. The neighbor list associated with each HA can be a reference to migrate the registration related jobs from a heavy loaded HA to a light loaded one.
3. In the GFA: The assigned HA is randomly selected from a list maintained by the GFA, which can avoid a certain HA to become heavy loaded.

Next, we analyze the registration update, connection setup, and packet delivery costs in terms of hop distances for these mechanisms. The standard Mobile IP follows the path MN-FA-HA during the registration update, hence the cost is twice (for round trip) the hops from the MN to the HA. During the connection setup and packet delivery, the path CN-HA-FA-MN is followed.

In the region_based one, the inter-domain registration update follows the path MN-FA-GFA-HA; while the intra-domain registration update follows the path MN-FA-GFA. We use the bracket in the table to denote the maximal and minimal values of the cost. The connection setup and packet delivery follow the path CN-HA-GFA-FA-MN.

In the DNS-based one, the path MN-FA-DNS is followed during the registration update, and the path CN-DNS-CN-FA-MN is followed during the connection setup. The path segment CN-DNS-CN is to iteratively locate the proper name server in the domain name hierarchy. Assume the average number of iterations is denoted by *d*. The packet delivery follows the path CN-FA-MN.

In the P2P_based one, the path MN-FA-GFA-Assigned HA-Destined HA is followed during the inter-domain registration update; while the path MN-FA-GFA is followed during the intra-domain registration update. The path CN-DNS-Initiated HA-Destined HA-Assigned HA-CN-GFA-FA-MN is followed during the connection setup. The P2P lookup from an initiated HA to a destined HA would take log *N* hops in a typical P2P network like Chord. The packet delivery follows the path CN-GFA-FA-MN. As can be seen, the connection setup is longer than that in other mechanisms. It is one of our future work to reduce this setup delay by using data replication in the P2P network.

5. CONCLUSION

In this paper, we introduce the emerging P2P network technology into Mobile IP to efficiently support mobility management. In our proposed architecture, we provide the dynamic HA assignment and the capability of load balancing and fault tolerance on the HA. We overcome the problems in Mobile IP such as triangular routing and frequent registration update at the expense of the delay on connection setup. In the future, we will incorporate the AAA (Authentication, Authorization, and Accounting) server into our architecture to enhance the capability of security.

ACKNOWLEDGMENTS

The authors would like to thank the anonymous referees for their helpful suggestions. This research was partially supported by the National Science Council of the Republic of China under Contract No. NSC 93-2752-E-007-003-PAE.

REFERENCES

1. C. Perkins, "IP Mobility Support for IPv4, Revised," *RFC3220*, IETF, Jan. 2002.
2. L. Morand and S. Tessier, "Global Mobility Approach with Mobile IP in All IP Networks", *IEEE Int'l Conf. on Communications* (ICC), pp. 2075-2079, May 2002.
3. Campbell, J. Gomez, S. Kim, Z. Turanyi, C-Y. Wan, and A. Valko, "Design, Implementation and Evaluation of Cellular IP," *IEEE Personal Communications Magazine*, vol. 7, no. 4, pp. 42-49, Aug. 2000.
4. S. Das, A. Mcauley, A. Dutta, A. Misra, K. Chakraborty, and S. K. Das, "IDMP: An Intradomain Mobility Management Protocol for Next-Generation Wireless Networks," *IEEE Wireless Communications*, vol. 9, no. 3, pp.38-45, Jun. 2002.
5. R. Ramjee, K. Varadhan, L. Salgarelli, S. R. Thuel, S. Y. Wang, and T. L. Porta, "HAWAII: a Domain-Based Approach for Supporting Mobility in Wide-Area Wireless Networks," *IEEE/ACM Trans. on Networking*, vol. 10, No. 3, pp. 396-410, Jun. 2002.
6. Rowstron and P. Druschel, "Pastry: Scalable Distributed Object Location and Routing for Large-Scale peer-to-Peer Systems," *Proc. IFIP/ACM Int'l Conf. on Distributed Systems Platforms* (*Middleware*), Nov. 2001.
7. S. Ratnasamy, P. Francis, M. Handley, R. Karp, and S. Shenker, "A Scalable Content-Addressable Network," *ACM SIGCOMM*, pp. 161-172, Aug. 2001.
8. Stoica, R. Morris, D. Karger, M. F. Kaashoek, H. Balakrishnan, "Chord: A Scalable Peer-to-Peer Lookup Service for Internet Applications," *IEEE/ACM Trans. on Networking*, vol.11, no. 1, pp. 17-32, 2003.
9. Y. Chen and T. Boult, "Dynamic Home Agent Reassignment in Mobile IP," *IEEE Wireless Communications and Networking Conference*, pp.44-48, 2002.
10. R. Zheng, Y. Ge, J. C. Hou, and S. R. Thuel, "A Case for Mobility Support With Temporary Home Agents," *ACM SIGMOBILE Mobile Computing and Communications Review*, vol. 6, no. 1, pp. 32-46, Jan. 2002.
11. M. Kulkami, A. Patel, and K. Leung, "Mobile IPv4 Dynamic Home Agent Assignment," *IETF*, draft-ietf-mip4-dynamic-assignment-00.txt, Jan. 2004, Work in Progress.
12. M. Conti, E. Gregori, and S. Martelli, "DNS-based Architecture for an Efficient Management of Mobile Users in Internet," *15th Int'l Symposium on Parallel and Distributed Processing*, pp. 1957-1964, Apr. 2001.
13. C. Snoeren and H. Balakrishnan, "An End-to-End Approach to Host Mobility," *6th Int'l Conf. on Mobile Computing and Networking* (*MOBICOM*), pp. 155-166, Aug. 2000.
14. P. Vixie, S. Thomson, Y. Rekhter, J. Bound, "Dynamic Updates in the Domain Name System (DNS UPDATE)", *RFC2136*, IETF, Apr. 1997.
15. J. W. Lin and J. Arul, "An Efficient Fault-Tolerant Approach for Mobile IP in Wireless Systems," *IEEE Trans. on Mobile Computing*, vol. 2, no. 3, Jul.-Sept. 2003.
16. J. H. Ahn and C. S. Hwang, "Efficient Fault-Tolerant Protocol for Mobility Agents in Mobile IP," *15th Int'l Symposium on Parallel and Distributed Processing*, pp. 1273-1280, 2001.
17. C. Mctaggart, "Telephone Numbers, Domain Names, and ENUMbers," *IEEE Communications Magazine*, pp. 26, Sept. 2002.
18. P. Faltstrom, "E.164 Number and DNS," *RFC2916*, IETF, Sept. 2000.