DNS Extensions to Support Location Management in IP Networks

Alexandros Kaloxylos, Stathes Hadjiefthymiades, Lazaros Merakos

Communication Networks Laboratory, Department of Informatics, University of Athens
TYPA Building, Panepistimioupolis, Illisia, Athens 15784, Greece
{agk | shadj | merakos}@di.uoa.gr

Abstract

The explosive growth of the Internet and the rapid developments in the area of mobile communications necessitate the design of new location management mechanisms. Existing solutions, such as the Mobile-IP, offer an efficient solution for connectionless best-effort traffic. However, these solutions do not perform sufficiently well in cases where the applications require a stable QoS (i.e., reservation of resources along the data path). In this paper we propose a new approach for mobility management that aims to resolve the problem of misrouted paths between mobile users in an IP network.

1. Introduction

Nowadays, we are experiencing an impressive growth in the area of wireless/mobile communications with technologies like wireless ATM, GSM and the emerging Universal Mobile Telecommunications System standard (UMTS/IMT2000). Such platforms, apart from conventional, circuit switched telephony enable the use of desktop computer applications with multimedia capabilities by the nomadic subscriber. The applications' arena is currently dominated by software designed and tuned for IP networks (e.g., CuSeeMe, NetMeeting, and WWW). In the IP world, the IETF's Mobile-IP (MIP) proposal presents an effort to deal with the problem of terminal mobility. Quality of Service (QoS) is also an important issue in the IP community, since, until recently, the best-effort model of service provision did monopolize the Internet. IETF is dealing with QoS through the DiffServ framework and the Resource reSerVation Protocol (RSVP - IntServ). The combination of the above-mentioned IP-based technologies is considered of extreme importance for the evolving area of mobile-aware multimedia applications.

According to the mobility management scheme suggested by IETF (MIP specification), when some fixed IP node (FN) wants to contact a mobile terminal (MN), the request is initially addressed to the called terminal's home network (the network to which the terminal administratively belongs). If the called terminal is found there, communication is established normally. If not, inbound traffic is diverted (by means of the IP tunneling approach), by a special entity named Home Agent (HA), to the current location of the terminal. In the reverse direction (MN $\rightarrow$ FN), communication is also performed normally - through an optimal path. This triangular routing scheme has been identified as the most important problem of MIPv4. In the upcoming MIP version 6, traffic through the home network is only exchanged at the very first stage of communication. After this initial stage, communication is performed through an optimized path in both the uplink and the downlink directions. The requirement for this initial stage, though, causes considerable problems in resource reservation schemes like the RSVP protocol (for real-time traffic) since the relevant signaling flows through the sub-optimal path (FN $\rightarrow$ HA $\rightarrow$ MN) which, additionally, encompasses an IP-tunnel. The problem is aggravated by the path followed by reservation confirmations (RESV). The discrepancy between IP-based mobility management and resource reservation schemes has been considered in [15] but the initial communication stage through the HA is not avoided in the proposed solutions.

A possible solution to the aforementioned problem could be the design and implementation of a new mechanism that determines the current location of a MN prior to the reservation of the required resources. One way of discovering the required location information is to take advantage of the DNS (Domain Name System) functionality. More specifically, each time a MN moves to a new area, it receives a new address and notifies its primary name server. This new address will be available to a calling node through DNS queries for address resolution.

The rest of the paper is organized as follows. In Section 2 we describe in detail, the proposed solution. In Section 3 we discuss several performance and evaluation issues for the proposed mechanism. We conclude this paper in section 4.

2. Location Management using DNS functionality

As described in ([1], [16], [17]), the goal of domain names is to provide a mechanism for naming resources in such a way that the names are usable in different hosts, networks, protocol families, internets, and administrative organizations.

Domain names are passed as arguments by users to local agents, called resolvers, which retrieve information associated with the domain name (Figure 1). Thus, a user might ask for the host address or mail information associated with a particular domain name. To enable the user to request a particular type of information, an appropriate query type is passed to the resolver with the domain name. To the user, the domain tree is a single information space while the resolver is responsible for hiding the distribution of data among name servers from the user.

The database that makes up the domain space is distributed among various name servers. Different parts of the domain space are stored in different name servers, although a particular data item will be stored redundantly in two or more name servers. The resolver starts its operation with knowledge of at least one name server. When the resolver processes a user query it asks a known name server for the information and in return, the resolver either receives the desired information or a referral to another name server. Using such referrals, resolvers learn the identities and contents of other name servers. Resolvers are responsible for dealing with the distribution of the domain space and dealing with the effects of name server failure by consulting redundant databases in other servers.

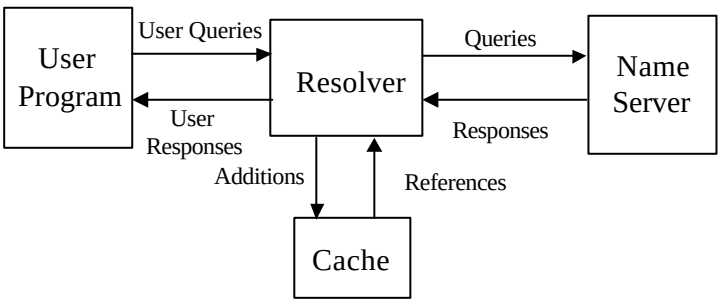

Figure 1: DNS Functionality

Name servers manage two kinds of data. The first category of information is held in sets called zones. Each zone is the complete database for a particular "pruned" subtree of the domain space. Data associated with a specific node is called Resource Record (RR). A name server periodically checks to make sure that its zones are up to date, and if not, obtains a new copy of updated zones from master files stored locally or in another name server. The second kind of data is cached data that was acquired by a local resolver. Such data may be incomplete, but improve the performance of the retrieval process when non-local data are repeatedly accessed. Cached data are

eventually discarded by a timeout mechanism.

In order to tackle the problem of location management in IP networks we propose the design and implementation of an extended DNS system that, besides the well-known *name-to-address mapping*, would also perform a *name-to-location mapping* (Figure 2). In such an approach, at least the primary name server of a domain would be responsible for keeping track of the current location of the mobile nodes that use this domain as their home area. This can be achieved by assigning two addresses to the mobile nodes. The first address is a permanent one (Mobile Home Address – MHA) and is used by all nodes (fixed or mobile) that want to establish communication with the mobile node. The prefix of the MHA denotes the address of the home domain of the mobile node. The second address is a temporary one (Mobile Foreign Address – MFA) and it is acquired each time the MT registers in a domain different from its home. Each time a mobile node acquires a MFA, the primary name server of its home domain is notified and stores this information for future use.

Although the assignment of MHA addresses can be done even manually, there is obviously a need for dynamically assigning MFAs addresses to mobile nodes that have recently arrived to a foreign domain. This can be done using the DHCP protocol ([10], [11]), or IPv6 stateless address auto-configuration ([12], [13]). Since the MFA address has to be transmitted back to the primary server of the mobile node, new signals and possibly entities are needed. The designers can take however advantage of new RFCs that extend the functionality of the DNS (e.g., [7], [9]).

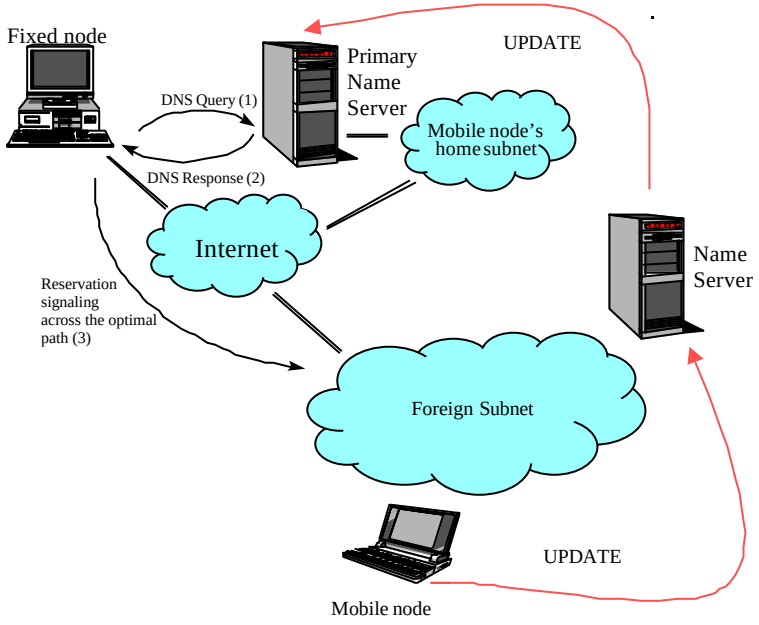

Figure 2: Communication involving enhanced DNS

More specifically, the UPDATE messages as defined in [7], could be used for notifying the name server of the current location as well as the primary name server of a mobile node. This message has the format shown in Figure 3:

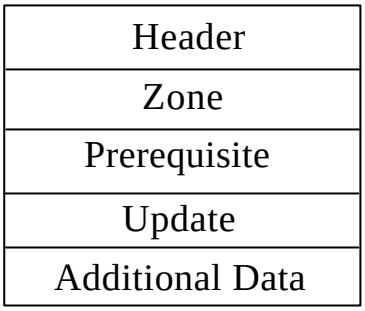

Figure 3: Format of the UPDATE message

The Header Section specifies that this message is an UPDATE, and describes the size of the other sections. The Zone Section names the zone that is to be updated by this message. The Prerequisite Section specifies the starting invariants (in terms of zone content) required for this update. The Update Section contains the edits to be made, and the Additional Data Section contains data that may be necessary to complete, but is not part of, this update. The Update section contains Resource Records to be added to or deleted from a specific zone. Note here that any duplicate Resource Records are silently ignored by the nameserver. This means only that the last update that contains the most recent information will be examined. The UPDATE signal could be issued by the mobile node or, alternatively by the Home Agent of the node. In this last scenario the binding update procedure could be used for triggering the transmission of the UPDATE signal towards the nameserver.

Location searching is performed each time a terminal wishes to communicate with a mobile node. In this case a name-to-address resolution phase is required before actual data are exchanged (steps 1-3 in Figure 2). At the end of this phase the terminal receives the MFA of the mobile node, and it is able to establish a flow along an optimum network path.

This approach takes advantage of the fact that the users of the IP applications use extensively the name of a node instead of its address. For the execution of this task, the name server of the calling terminal is contacted. If the called terminal is a mobile node, then the aforementioned name server will have to contact the primary name server of the called mobile node, since the later is aware of the current location of the called terminal.

3. Performance and implementation considerations

Since the phase of address resolution is a default operation in IP networks, we believe that resolving the location of a mobile node during the same phase is surely advantageous. The actual resolution of the current location of a mobile node can be simply performed by having the primary name server of the home domain of the mobile node, replying to requesting nodes not only with the MHA address of a mobile node but also with its MFA (if any). This mechanism presents two major advantages. The first one is that location-searching time is minimized, since the current location of a mobile node is discovered during the resolution of its address. Moreover, optimum flow of packets is achieved, since resources are reserved only along the path towards the current location of the mobile node and not through its Home Agent.

Storing location information in the DNS is not a new idea, although until now it has been targeted to fulfil different needs. For example, in [5], a mechanism is described to allow the DNS to carry location information about hosts, networks, and subnets. The information stored with this mechanism is aimed to be used in "visual traceroute" applications and network management applications that could use the location information to generate maps of hosts and routers being managed. Using the DNS to deal with location management issues is also mentioned in [2], although arguments, such as the estimated complexity and lack of security, are expressed. Although security problems also exist in the Mobile IP approach (when the mobile node registers in the network), these arguments

were quite true. However, during the past years, researchers world-wide have been working to extend existing protocols. This work has been mainly due to the required modifications imposed by IPv6. Thus, issues like addressing ([3]), DNS extensions ([4]), and security ([6], [8]) have been addressed again. What we propose in this paper is to take advantage of the current trend to re-define existing standards and design the required functionality to handle location management in an efficient way.

For the aforementioned mechanism to work, the standard DNS functionality has to be modified and new interfaces have to be specified. Firstly, we believe that using the standard caching functionality of DNS for mobile nodes is not a good strategy. This is because the probable frequent movement of nodes will render the cached location information inconsistent. For this reason, cached location information can either be forbidden or can be set to have minimum TTL (time-to-live). An alternative solution would be the separation of the address space to different classes of mobile terminals according to their estimated mobility pattern. More specifically, desktop computers would normally be assigned a low mobility profile, whereas handheld devices would be assigned with a high mobility profile. Obviously, this information would be used to determine whether the information about a mobile node should be cached or not.

Concerning the behavior of secondary DNS servers, we believe that their constant update server about the current location of mobile nodes would unnecessarily overload the network. The only functionality needed in the secondary name servers is the transfer of the, possibly distinct, mobile zone files from the primary name server in a different, predefined frequency. These zone files can include security and authentication information and can be used only when a primary name server is not functioning.

4. Conclusions

In this paper we have presented a new approach for performing location management in IP networks. The main goal of this approach is to tackle efficiently the problem of reserving resources along a misrouted path. This can be easily achieved by enhancing the functionality of the DNS to store and retrieve the current location of the mobile nodes when they move in foreign areas. The proposed approach can work supplementary to existing mechanisms since it can be used only for applications that require data flows of a stable QoS.
Although the new mechanism presents several advantages, such as the minimization of the location searching time and the establishment of flows along optimum paths, analysis is required to evaluate its performance, especially if caching cannot be easily used.

References:

1. P. Albitz and C. Liu, “DNS and BIND”, O’ Reilly & Associates Inc, ISBN 1-56592-010-4, July 1994.
2. C. E. Perkins, “Mobile IP”, IEEE Communications Magazine, May 1997.
3. R. Hinden et al., “IP Version 6 Addressing Architecture”, RFC 1884, December 1995.
4. S. Thomson and C. Huitema, “DNS Extensions to support IP version 6”, RFC 1886, December 1995.
5. C. Davis et al. “Means for Expressing Location Information in the Domain Name System”, RFC 1876, January 1996.
6. D. Eastlake and C. Kaufman, “Domain Name System Security Extensions”, RFC 2065, January 1997.
7. P. Vixie et al., “Dynamic Updates in the Domain Name System (DNS UPDATE)”, RFC 2136, April 1997.
8. D. Eastlake, “Secure Domain Name System Dynamic Update”, RFC 2137, April 1997.
9. P. Vixie, “Extensions to DNS (EDNS1)”, Internet Draft (draft-ietf-dnsind-edns1-01.txt), November 1998.
10. R. Droms, “Dynamic Host Configuration Protocol”, RFC 2131, March 1997.
11. J. Bound and C. Perkins, "Dynamic Host Configuration Protocol for IPv6 (DHCPv6)", Internet Draft (draft-ietf-dhc-dhcpv6-13.txt), July 1998.
12. T. Narten, E. Nordmark, and W. Simpson, “Neighbor Discovery for IP Version 6”, RFC 1970, August 1996.
13. S. Thomson and T. Narten, “IPv6 Stateless Address Auto-configuration”, RFC 1971, Network Working Group, August

1996.
14. G. Fankhauser, S. Hadjiefthymiades, N. Nikaein, and L. Stacey, "RSVP Support for Mobile IP Version 6 in Wireless Environments", Internet draft (draft-fhns-rsvp-support-in-mipv6-00), November 1998.
15. P. Mockapetris, “Domain Names – Concepts and Facilities”, RFC 1034, November 1987.
16. P. Mockapetris, “Domain Names – Implementation and Specification”, RFC 1035, November 1987.