
Ravi Pendse | Electrical Engineering and Computer Science

Quantitative Analysis of Enhanced Mobile IP

Patricia K. Best

Ravi Pendse

Affiliation:
Patricia K. Best, LSI Logic
Ravi Pendse, Wichita State University

Quantitative Analysis of Enhanced Mobile IP

Patricia K. Best, LSI Logic

Ravi Pendse, Wichita State University

Abstract

As the popularity of mobile computing grows, the associated protocols and their scalability are subject to much closer scrutiny. Mobile IP relies heavily on the use of IP-to-IP tunneling, requiring 20 bytes of overhead for every packet routed to or from a mobile node, assuming reverse tunneling is enabled. The goal of this research was to show that Enhanced Mobile IP (EMIP) eliminated the overhead by replacing tunneling with Network Address Port Translation (NAPT) without significantly impacting other performance factors. EMIP was implemented and benchmarks were used to compare EMIP and Mobile IP. The percentage of overhead with EMIP approached zero as the number of packets exchanged increased, while the percentage of overhead with Mobile IP remained constant. Once three or more round-trip packets were exchanged, the overhead of EMIP was less than Mobile IP, and when 1000 round-trip packets were exchanged, EMIP resulted in a bandwidth savings of almost 40,000 bytes. To achieve the bandwidth savings, EMIP introduced a one-time delay in session startup ranging from 160 to 260 ms when compared to Mobile IP, but it does not significantly impact the scalability or overall performance of the protocol. Therefore, the decrease in bandwidth consumed by the overhead of EMIP greatly outweighs the one-time delay and additional memory required.

Introduction

Mobile computing was envisioned to allow users to connect to networks from any location to exchange business and personal information. As the popularity of real-time traffic such as voice and video grows, mobile computing is also being used to facilitate these applications. Mobile IP was developed to allow nodes to change location while maintaining network connectivity [1]. Tunneling was an existing networking concept adopted as part of Mobile IP for packet redirection. Although reusing existing networking technologies has benefits, tunneling has several drawbacks. The added overhead required for tunneling packets reduces the bandwidth available in the wired network. Scalability becomes a concern as the number of Mobile Nodes (MNs) increases and the tunneling overhead consumes available network bandwidth.

Enhanced Mobile IP (EMIP) was developed to eliminate the need for tunneling when providing connectivity to mobile nodes [2]. The goal of this research was to analyze the scalability of EMIP in terms of the overhead of the mobility protocol and the impact it has on other aspects of performance and scalability in comparison with Mobile IP. The remainder of this article is organized as follows. We provide an overview of Mobile IP and Enhanced Mobile IP. Then we give a brief overview of the implementation of EMIP and describe the test scenarios. The results and analysis are presented, and the conclusions are discussed.

Overview of Mobile IP and Enhanced Mobile IP

Mobile IP

Mobile IP defines the Home Agent (HA) and the Foreign Agent (FA) in order to facilitate a Mobile Node (MN) that maintains connectivity as it changes location. The FA assigns the MN a care-of address (CoA) while it resides in the foreign network. The MN then registers its CoA with the HA. After registration takes place, a tunnel is created between the HA and the FA. When a packet is sent to the MN from a corresponding node (CN), the packet is routed to the home network of the MN. The HA intercepts the packet and sends it through the tunnel to the CoA, which is typically the FA. When the FA receives the packet, it forwards the packet to the MN. IP-in-IP Encapsulation must be supported by HAs and FAs for tunneling datagrams in Mobile IP and is used as the tunneling mechanism in this research [3]. For IP-in-IP Encapsulation, the encapsulating IP header adds 20 bytes to the size of each packet in the tunnel. Often packets sent from the MN to a CN are routed back through the tunnel to the HA before reaching their destination. This process is called reverse tunneling, and can be used to prevent packet filtering and provide accounting information to the home network.

Providing quality of service (QoS) with Mobile IP has been researched using both Integrated Services and Differentiated Services tech-

Step	Description	Source IP	Destination IP	Source port	Destination port	Payload
1	Original packet sent by the CN and intercepted by the HA	CN_{IP}	MN_{IP}	CN_{PORT}	MN_{PORT}	Data
2	Mapping request sent by the HA to the FA	HA_{IP}	FA_{IP} (CoA)	Any	434	MN_{IP}, MN_{PORT}, CN_{IP}, CN_{PORT}, CoA_{HA}, CoP_{HA}, ID
3	Mapping reply sent by the FA to the HA (mobility binding tables of the HA and the FA are updated)	FA_{IP} (CoA)	HA_{IP}	Any	434	ID, CoA_{FA}
4	Message sent from the HA to the FA	CoA_{HA}	CoA_{FA}	CoP_{HA}	MN_{PORT}	Data
5	Message sent from the FA to the MN	CN_{IP}	MN_{IP}	CN_{PORT}	MN_{PORT}	Data

Table 1. *Packet sent from the CN to the MN.*

niques. Since the Resource Reservation Protocol (RSVP), commonly used to provide QoS in wired networks, requires added overhead and complexity to work across tunnels, several modifications have been proposed for RSVP with Mobile IP [4–6]. Network Address Translation (NAT), when applied at the edge of foreign networks, also complicates Mobile IP requiring UDP tunneling since tunnels do not work well with NAT [7].

Enhanced Mobile IP

Enhanced Mobile IP (EMIP), developed to eliminate tunneling, is described in [2]. It uses the HA and the FA defined by Mobile IP, and the same mechanisms for discovering the CoA and registering with the HA. EMIP differs from Mobile IP in the way packets are redirected from the HA to the FA. A concept built on Network Address Port Translation (NAPT) is used in place of tunnels [8].

With EMIP, when the MN registers with the HA a tunnel is not created between the HA and the FA. Instead, a mapping is created between the HA and the FA when each connection the MN communicates across is established. Mappings are creating by intercepting packets to and from the MN at the HA and the FA. The HA and FA then exchange mapping request and mapping reply messages containing the source and destination IP and port address of the MN and the CN. The mobility agent that intercepts the packet also supplies a care-of port (CoP) that is used by the HA and the FA to identify the mapping. Once the mapping between the HA and the FA is established for a communication session, the mobility agents can redirect packets to and from the MN by modifying the IP and TCP or UDP packet headers instead of using a tunnel.

For example, when an MN moves to a foreign network, it is assigned a CoA and registers with its HA, just as with Mobile IP. When a CN sends a packet to the MN, it is routed to the home network of the MN and intercepted by the HA. The HA buffers the packet to the MN and sends a mapping request to the FA, which contains the IP and port address of both the CN and the MN. The HA assigns the session a CoP and includes the CoP in the mapping request. The HA also includes a CoA_{HA} to use for the mapping that can differ from the IP address of the HA used for registration. The HA and the FA store the mapping information in a mapping table. The FA sends a mapping reply when it receives the mapping request to inform the HA that it has stored the mapping information. The mapping reply can include an optional CoA_{FA} to be used by the FA for the mapping to increase the number of MNs it can support. The mapping request and mapping reply packets contain an identification number so that the mobility agents can match the mapping request and mapping reply packets.

Once the mapping is established, the HA modifies the buffered packet destined for the MN in order to redirect the packet to the CoA. The source and destination IP addresses are changed to that of the HA and the FA, and the source port address is replaced with the CoP. This allows the packet to be routed to the FA since its IP address is now the destination of the packet. When the FA receives the packet, it uses the source IP and port address, which are the CoA_{HA} and the CoP, to find the mapping in its mapping table. The mapping entry is used to replace the original header with the IP and port addresses of the MN and the CN so that the packet can be forwarded to the MN. Table 1 shows the packet sent from the CN to the MN and the modifications made to the header at the various stages. It also shows the information exchanged in the mapping request and mapping reply packets.

All future packets exchanged for this communication session can use the existing mapping and do not require packets to be buffered at the HA or mapping request and reply messages to be exchanged. If the MN chooses to communicate with a new CN or on a different port address with the same CN, a new mapping would have to be established. Often with Mobile IP, reverse tunneling is used where packets from the MN are redirected to the HA before being sent to the CN. This process is termed reverse routing with EMIP. To perform reverse routing, the FA intercepts packets from the MN and modifies the header to redirect them to the HA using

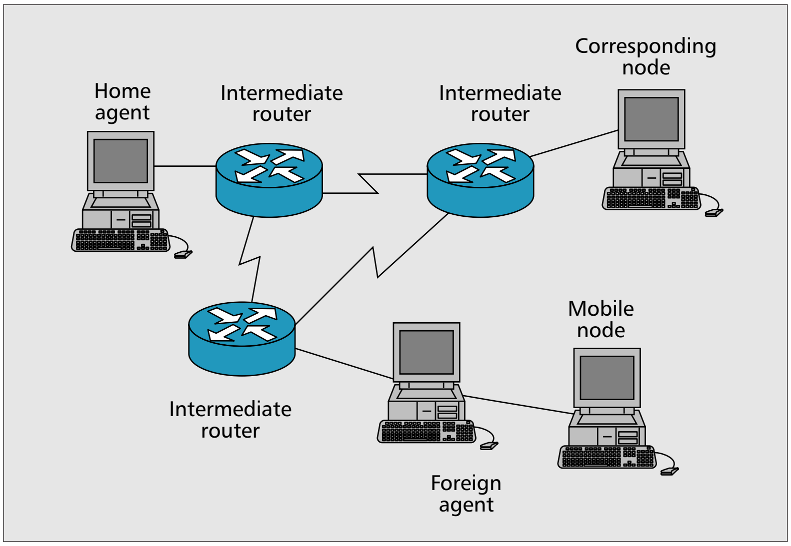

Figure 1. *Test network.*

the mapping information. If the MN sends the first packet of a new communication session, the FA must supply the CoP for the communication.

EMIP eliminated the tunneling overhead, replacing it with a one-time bandwidth overhead to exchange the mapping request and mapping reply packets. However, a delay was introduced with EMIP to buffer the first packet of each new communication in order to establish a mapping between the HA and the FA. The original version of EMIP described in [2], now termed EMIP pre-Performance Improvement (EMIPprePI), required the packet to be buffered until the mapping request is sent and the mapping reply is received. This is the order shown in Table 1. It was later determined that the buffered packet could be transmitted once the mapping request was sent with the assumption that it was received properly. This would result in moving step 3 in Table 1 to a time after the packet had been delivered to the MN. Since the mapping request packet is small and has a low probability of being dropped in the network, this assumption is typically valid. Retry mechanisms are added in the protocol to account for the possibility of the mapping request being lost. By sending the buffered packet immediately after sending the mapping request, the delay of buffering the packet is reduced. This technique is considered the standard for the EMIP protocol.

Implementation of Enhanced Mobile IP and Testing Procedures

Dynamics Mobile IP and EMIP Implementation

To provide a more complete performance evaluation of EMIPprePI and EMIP, the protocols were implemented in software, and tests were executed to measure specific parameters. The Dynamics Mobile IP software was used for the implementation. Dynamics Mobile IP relies on the Linux implementation of tunneling to perform tunneling between the FA and the HA. When an MN registers, the Dynamics Mobile IP application creates the tunnel using the Linux interface. The primary difference between Mobile IP and the newly developed protocols is that EMIPprePI and EMIP require new Network Address Translation (NAT) entries for each new connection and must continue to interact with the packets to detect new connections and establish mappings after registration is complete. IPTables is the firewall technology supplied with Linux, which also includes its NAT implementation. Connection tracking is another Linux module critical to the support of NAT. IPTables and connection tracking were used to intercept packets, obtain the mapping information from the headers, and redirect packets using NAT to modify the source and destination IP and port addresses as described by the protocol.

Memory and Bandwidth Overhead of EMIPprePI and EMIP

To compare the memory requirements for EMIPprePI and EMIP to Mobile IP, the added memory needed to store the mapping information in the implementation was analyzed. Each mapping contained all the information not present in the registration information, consisting of the CN IP address, the CoA_{FA}, the CoA_{HA}, the CN and MN port addresses, the CoP_{FA}, the CoP_{HA}, a timeout value, an identification value for the mapping, and a pointer to the next mapping. A 4-byte pointer to the linked-list of mapping entries was also added to the registration information. The total additional memory required would be 4 bytes/MN and 32 bytes/mapping.

The mapping request and mapping reply messages represent the total bandwidth overhead of EMIPprePI and EMIP. The mapping request packets are 52 bytes in size and contain the IP and UDP headers, the CN and MN IP and port addresses, the CoA and CoP, the identification value of the mapping, the mobility packet type, and one byte for options reserved for future use. The mapping reply packets are 40 bytes in size and contain the IP and UDP headers, the CoA and CoP, the mobility packet type, and one byte for options also reserved for future use.

Test Parameters and Procedures

EMIPprePI and EMIP were compared to Mobile IP in terms of round-trip delay between the MN and the CN and the percentage of overhead the protocol generates. Traffic parameters were varied during testing to sufficiently simulate large and small data transfers, and long and short communication sessions. Small packet sizes simulated applications such as Voice over IP (VoIP), while larger packet sizes simulated applications such as file transfers.

The test simulations were executed in the Student Routers Laboratory at Wichita State University. The network setup shown in Fig. 1 was used for each test scenario. The HA, FA, MN, and CN were personal computers (PCs) with Pentium III processors executing at 500 MHz. They all had 512 kbytes of integrated Level 2 cache, and the memory consisted of 128

Mbytes of SDRAM. The PCs all used the Redhat Linux Version 7.0 operating system. The Dynamics Mobile IP software, or the modified version for the EMIP implementation, was executed on the MN, FA, and HA. The remaining network entities were Cisco 2500 family routers running the Routing Information Protocol (RIP). Serial links existed between each Cisco router running with a clock speed of 4 Mb/s, while 10 Mb/s Ethernet links were used to connect the PCs to the routers and the MN to the FA. A wired link was used between the MN and the FA since mobility was not being tested.

Test Results and Analysis

The scalability of the mobility protocols was analyzed by evaluating how well the protocols would perform under various loads. The analyses of the results were used to determine if EMIPprePI and EMIP reduced the amount of bandwidth required by Mobile IP with minimal trade-offs in the other performance factors. Although a large combination of packet sizes and number of packets exchanged was tested for both Transmission Control Protocol (TCP) and User Datagram Protocol (UDP) traffic, the trends observed in the results were similar for all packet sizes and traffic types. Therefore, only a representative section of the results are presented.

Percentage of Mobility Bandwidth Overhead

The mobility overhead was determined to be the bandwidth the mobility protocol required to transmit data to and from the mobile node for a single communication session. Since registration overhead is identical for all the mobility protocols, it was not included in this analysis. The percentage of overhead was calculated as the amount of overhead divided by the combined overhead and payload transferred. The IP header and the TCP or UDP header were removed from the calculations, since they result in the same overhead for all protocols. The overhead of Mobile IP for each packet transferred was calculated as the size of the tunneling header, 20 bytes, multiplied by the number of packets exchanged. The overhead for each packet transferred using EMIPprePI and EMIP was the one-time overhead related to the mapping request and mapping reply packets required to establish each connection. The percentage of overhead required for each protocol to transmit 10-Byte UDP packets round trip with varying payloads is presented in Fig. 2.

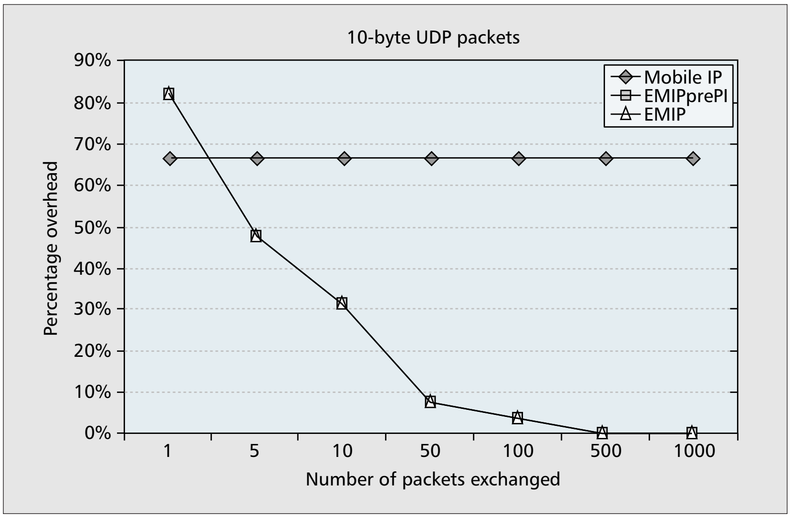

Figure 2. *Percentage of mobility overhead with 10-byte UDP packets transmitted.*

The overhead of Mobile IP was reduced in EMIPprePI and EMIP for all packet sizes and all but the smallest number of packets exchanged in a session. With Mobile IP the mobility overhead remains constant regardless of the number of packets exchanged in the session, while with EMIPprePI and EMIP the percentage of mobility overhead approaches zero as the number of packets exchanged increases.

When only one round-trip packet was exchanged, EMIPprePI and EMIP required 52 additional bytes of overhead relative to Mobile IP. However, EMIPprePI and EMIP saved 108 bytes of bandwidth when only five round-trip-packets were exchanged and almost 40,000 bytes of bandwidth when 1000 round-trip packets were exchanged. Therefore, the greatest savings would occur when transferring large files or when using real-time applications such as Voice over IP (VoIP) where a large number of packets are exchanged.

The overhead related to Mobile IP created scalability concerns for the protocol, since a large number of MNs would result in a large amount of bandwidth consumed by the 20 bytes of overhead in each packet. The results have shown that EMIPprePI and EMIP required less than 100 bytes of overhead per connection with each MN. Since the overhead is one-time per connection instead of being on a per-packet basis, EMIPprePI and EMIP eliminated the scalability concerns related to bandwidth.

Delay and Throughput

To compare the delays of the protocols, benchmarks were used to measure the round-trip time required to send a packet from the MN to the CN and receive a packet of the same length in return. The MN waited for the reply packet from the CN before sending another packet. The delay measurements for the test scenario involving 10-byte UDP packets are shown in Table 2, and the delays for the 1000-byte TCP packets are shown in Table 3.

Tables 2 and 3 show that EMIP has approximately a 160 ms increase in delay over Mobile IP when transmitting the first packet, and EMIPprePI has a 235 ms increase in delay over Mobile IP. Since the difference in delay does not increase as the number of packets transmitted increases, it can be concluded that the delay only occurs for the first packet transmitted. This is shown in Table 4 by subtracting 160 ms from all of the EMIP delays and 235 ms from all of the EMIPprePI delays.

When the delay of the first packet is removed, the delays of all the protocols were very similar. After transmitting a large number of packets, the EMIPprePI and EMIP protocols even outperform Mobile IP. This occurs because Linux is

The memory requirements of the mobility agents create a scalability concern for all mobility protocols. For Mobile IP, EMIPprePI, and EMIP, the Dynamics Mobile IP implementation required 110 bytes of memory to store the registration information for each MN.

UDP	Mobile IP	EMIPprePI	EMIP
No. packets	Delay (ms)	Delay (ms)	Delay (ms)
1	3	241	169
5	15	249	174
10	30	263	189
50	152	382	306
100	304	530	453
500	1518	1711	1632
1000	3035	3187	3103

Table 2. *Round-trip delay measured when transmitting 10-byte UDP packets.*

TCP	Mobile IP	EMIPprePI	EMIP
No. packets	Delay (ms)	Delay (ms)	Delay (ms)
0	4	237	164
1	24	258	185
5	127	340	342
10	242	444	369
50	1045	1271	1196
100	2087	2304	2227
500	10,419	10,565	10,481
1000	20,833	20,892	20,798

Table 3. *Round-trip delay measured when transmitting 1000-byte TCP packets.*

slightly more efficient performing NAT translation than it is at tunneling packets.

The increases in delay seen with EMIPprePI and EMIP were entirely due to the one-time delay sustained when the first packet was transmitted. The results also show that the added delays seen with EMIP were entirely CPU processing delays, required for intercepting the first packet in a connection and adding the appropriate entries to the IPTables. EMIPprePI had approximately the same CPU processing delay as EMIP since both protocols execute much of the same code, but EMIPprePI also has the added propagation delay of the mapping request and mapping reply packets when handling a new connection.

Since the delay was caused by CPU processing with EMIP, a faster processor would improve the results. The implementation itself could also be modified to reduce the one-time processing delay by moving the implementation from the application level to the kernel level. If EMIPprePI and EMIP were implemented within the Linux kernel itself or within an operating system designed specifically for networking, the delays of adding the IPTables entries would be reduced. Even though the one-time delay could be reduced, it will always exist since the protocol does require more processing to occur.

The delays of both protocols would be scalable with an increase in MNs. The only delay of Mobile IP that is dependent on the number of MNs would be the lookup of the proper tunnel to use to redirect packets to and from the MN. For EMIPprePI and EMIP, a large number of MNs would require additional lookup time to add a mapping for a certain MN. There would also be an added delay to determine the proper NAT translation to perform. This delay is comparable to the increase in the delay for Mobile IP to determine the appropriate tunnel to use. The delays would all be minor due to the efficiencies of these routing techniques and the current processing power of routers. Therefore, delay is not considered a scalability concern for either EMIP or EMIPprePI.

Redirection Using NAT

To eliminate tunneling in EMIPprePI and EMIP, NAPT was used to redirect packets between the HA and the FA. The use of NAT only creates a scalability concern if the number of unique combinations of IP addresses and port numbers available could limit the number of MNs supported by either the HA or the FA. To avoid using predefined port numbers with specific meanings, typically NAT only uses about 4,000 port numbers. Although it is unlikely that any mobility agent would be required to support more mappings than this at a given time, both the HA and the FA can also use an alternate CoA than the one used for registration. Therefore, NAT does not create any scalability issues with the EMIPprePI and EMIP protocols.

Mobility Agent Memory Requirements

The memory requirements of the mobility agents create a scalability concern for all mobility protocols. For Mobile IP, EMIPprePI, and EMIP, the Dynamics Mobile IP implementation required 110 bytes of memory to store the registration information for each MN. The information includes the IP addresses of the MN, HA, and FA, and timeout information, as well as security and accounting information. Since mobility agents have a limited amount of memory available to support Mobile IP, the memory requirements could limit the number of MNs supported.

EMIPprePI and EMIP require all the registration information stored with Mobile IP and an additional 4 bytes to reference the mappings for a given MN. Each mapping then requires 32 bytes of memory. An inactive MN will only require an additional 4 bytes of memory usage as compared to Mobile IP, while MNs that are currently transmitting data also require the additional 32 bytes per active communication session.

The number of mappings required for each MN depends on the behavior of each individual

UDP	Mobile IP	EMIPprePI	EMIP
No. packets	Delay (ms)	Delay (ms)	Delay (ms)
1	3	6	9
5	15	14	14
10	30	28	29
50	152	147	146
100	304	295	293
500	1518	1476	1472
1000	3035	2952	2943

Table 4. *Round-trip delay measured when transmitting 10-byte UDP packets, with the first packet delays removed from EMIPprePI and EMIP.*

user, the applications, and the mapping timeout values enforced by the protocol. For example, users that download information from the Internet will only require an active mapping during the download. They would not require an active mapping while they review the information they just received. Other users may frequently request information that requires a mapping to be maintained constantly. Another set of users may use one mapping for an application such as VoIP while requiring a second mapping for email or other applications. For the purpose of discussion in this research, it was assumed the average MN had one active mapping at any given time, since the amount of time MNs spend idle exceeds the amount of time a node would require multiple, simultaneous mappings. Based on this assumption, EMIPprePI and EMIP required 146 bytes of memory per MN while Mobile IP only required 110 bytes of memory.

The effect the additional 36 bytes per MN has on scalability depends on the amount of memory available on the mobility agent, and the expense of increasing the amount of available memory. A 2500 series Cisco router can have anywhere from 1 Mbyte to 16 Mbytes of Dynamic Random Access Memory (DRAM). For a router with only 1 Mbyte of DRAM, if the memory required to store mobility information was limited to 1 percent of the total DRAM, the router would be able to support 95 MNs with Mobile IP and only 71 MNs with EMIPprePI and EMIP. However, a router with 16 Mbytes of memory and the same 1 percent limit would support 1525 MNs with Mobile IP and 1149 MNs with EMIPprePI and EMIP. While Mobile IP will always be able to support more MNs than EMIPprePI and EMIP with the same amount of memory, a router can easily contain enough DRAM so that memory is not a scalability factor for EMIPprePI or EMIP. The cost of requiring ample memory on the mobility agents should be much lower than the cost required to support the extra bandwidth required with Mobile IP across the entire Internet and every network that supports mobility.

Conclusions

The primary goal of EMIPprePI and EMIP was to reduce the overhead of Mobile IP, without significantly reducing the scalability of the protocol based on other factors such as packet delay and the memory requirements of the mobility agents. Testing showed the percentage of overhead associated with EMIPprePI and EMIP was less than that of Mobile IP after only a few packets were transmitted to or from the MN via the mobility agents.

The bandwidth savings was much greater when a larger number of packets were exchanged. A single connection transmits a large number of packets when large files are transferred, which would occur with large email attachments, when downloading files from the Internet, or when loading Web pages containing large graphics files. Real-time voice and video applications can also generate a large number of packets. For example, most voice encoders only place about 20 to 30 ms of voice into a single packet to ensure a certain Quality of Service (QoS), requiring up to 50 packets/s. This means that a voice call could produce as many as 3,000 packets per minute. A three-minute voice call would use almost 180,000 bytes of additional bandwidth using Mobile IP instead of EMIPprePI or EMIP.

Although EMIPprePI and EMIP improved bandwidth scalability, they did not have a negative effect on delay. The delay is only seen when delivering the first packet in a communication session, and therefore does not affect the packets after the traffic flow begins. Since the delays occur before any packets are received by the destination, and since the delays were small and could be reduced further, the increase in delay seen with EMIP does not outweigh the benefits of the bandwidth saved.

The other scalability issue affected by the EMIPprePI and EMIP implementations were the memory requirements of the HA and the FA. The additional memory consumed with EMIPprePI and EMIP should be available on existing mobility agents and therefore does not impact scalability. Furthermore, the added memory required is confined to the HAs and FAs in the network, whereas the additional bandwidth required for Mobile IP is needed in all networks that are traversed between all HAs and FAs.

This research has shown that EMIPprePI and EMIP are more scalable mobility protocols than Mobile IP. The bandwidth savings achieved through the elimination of tunneling greatly outweighs any of the disadvantages. Along with the bandwidth savings, the elimination of tunneling may prevent fragmentation and eliminates the problems that tunneling creates for QoS in a Mobile IP environment. EMIPprePI and EMIP could be easily introduced into existing mobile networks since they can be hosted on current network equipment, and mobility agents could add EMIPprePI or EMIP functionality while still supporting Mobile IP for backwards compatibility.

Although EMIPprePI and EMIP improved bandwidth scalability, they did not have a negative effect on delay. The delay is only seen when delivering the first packet in a communication session, and therefore does not affect the packets after the traffic flow begins.

REFERENCES

[1] C. Perkins, "IP Mobility Support for IPv4," RFC 3344, Aug. 2002.

[2] P. Best and R. Pendse, "Enhanced Mobility Protocol for Wireless Networks," *IEEE Semiannual VTC*, Orlando, FL, Oct. 6–9, 2003.

[3] C. Perkins, "IP Encapsulation within IP," RFC 2003, Oct. 1996.

[4] A. K. Talukdar, B. R. Badrinath, and A. Acharya, "MRSVP: A Resource Reservation Protocol for an Integrated Services Network with Mobile Hosts," *Wireless Networks*, vol. 7, no. 1, Jan.–Feb. 2001, pp. 5–19.

[5] C. Tseng, G. Lee, and R. Liu, "A Hierarchical Mobile RSVP Protocol," *Wireless Networks*, vol. 9, no. 2, Mar.-Apr. 2003, pp. 95–102.

[6] S. Paskalis *et al.*, "An Efficient RSVP-Mobile IP Interworking Scheme," *Mobile Networks and Apps.*, vol. 8, no. 3, June 2003, pp. 197–207.

[7] H. Levkowetz and S. Vaarala, "Mobile IP Traversal of Network Address Translation (NAT) Devices," RFC 3519, Apr. 2003.

[8] P. Srisuresh and K. Egevang, "Traditional IP Network Address Translator (Traditional NAT)," RFC 3022, Jan. 2001.