



HEBIS  
2  
2002

**LIBRARY  
Michigan State  
University**

This is to certify that the

dissertation entitled

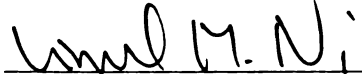
SUPPORTING REAL-TIME AND MULTIMEDIA APPLICATIONS  
IN WIRELESS NETWORKS

presented by

FAN DU

has been accepted towards fulfillment  
of the requirements for

Ph.D. degree in Computer Science

  
Major professor

Date September 24, 2001

**PLACE IN RETURN BOX** to remove this checkout from your record.  
**TO AVOID FINES** return on or before date due.  
**MAY BE RECALLED** with earlier due date if requested.

DATE DUE	DATE DUE	DATE DUE
OCT 07 8 2004		

**SUPPORTING REAL-TIME AND MULTIMEDIA APPLICATIONS IN WIRELESS  
NETWORKS**

**By**

**Fan Du**

**A DISSERTATION**

**Submitted to**

**Michigan State University**

**in partial fulfillment of the requirements**

**for the degree of**

**DOCTOR OF PHILOSOPHY**

**Department of Computer Science and Engineering**

**2001**



## **ABSTRACT**

### **SUPPORTING REAL-TIME AND MULTIMEDIA APPLICATIONS IN WIRELESS NETWORKS**

By

Fan Du

Wireless technology is one of the areas receiving most attention and investment. Wireless services are quickly becoming part of people's daily life. This dissertation presents a framework for supporting real-time and multimedia applications in wireless networks. Mobile applications require backbone network support from multiple aspects, including QoS, secure multicast, seamless handoff and others. Included in the framework presented in this dissertation is a set of new mechanisms addressing those requirements.

There are four components in the framework. Different needs of mobile applications are addressed by different component. The first component, Wireless Resource Reservation Protocol (WRSVP), enables wireless devices to reserve necessary resources in wireless networks, which is a necessary condition for QoS. With help of WRSVP, resources can be successfully reserved and refreshed without being affected by poor links and handoffs.

The second component, Handoff Protocol for Overlay Network (HOPOVER), is a handoff protocol designed specifically for overlay networks where multiple wireless networks coexist. With increasing number of high-speed moving wireless devices, smaller cell sizes, and more and more wireless networks being built, handoffs happen at increasingly higher frequency both horizontally and vertically. HOPOVER. enables

wireless devices to move freely and smoothly intra- and inter- network.

The third component, Smart Server Selection (S3) helps wireless devices in choosing the best server when they use replicated services. Traditional server selection solutions are hard to use in wireless networks because network topology is unstable from the viewpoint of both the servers and the clients. S3 enables accurate, fast and low overhead server selection by utilizing the client side DNS server and up-to-data routing information from routers.

The last component, Secure Transmission Backbone (STB) enables wireless devices to participate in secure multicast sessions. STB effectively solves the multicast key management problem and addresses the problems caused by handoffs and limited resources in wireless environment.

In summary, with all these components, we have solid support for mobile applications in overlay networks, thus help users to utilize their favorite applications anytime, anywhere.

## **ACKNOWLEDGMENTS**

First, I would like to thank my advisors, Dr. Lionel M. Ni and Dr. Abdol-Hossein Esfahanian, for their kind guidance, encouragement and enormous help. This dissertation would not be possible without them. I can always count on Dr. Ni and Dr. Esfahanian for advice and guidance whenever I need. They helped me so much in reviewing papers, brainstorming sessions and discussions. They also have contributed greatly to my maturity as a researcher. I was very fortunate to have had the opportunity to work with them in the past 5 years, and they will always be models to me.

I would like to thank my Ph.D. committee members, Dr. Tien-Yien Li and Dr. Matt W. Mutka, and other faculty and staff members in MSU. The past 5 years in MSU was definitely a rewarding experience and I thank them for their teaching, advice and help.

I also wish to thank my colleagues in The Advanced Technology (TEC) group with Deutsche Bank: Erik Oldekop, Levent Karaoglan, Mohamed Jamalooden and Sven Maerz. They helped me a lot to get a broad view of the industry. I also benefited greatly from their comments and discussions. Especially, I would like to thank Erik. His support made my working experience at TEC much smoother.

My time at MSU wouldn't have been half as enjoyable if not for my friends here. Thanks for all the good times. I would especially like to thank Dai Wan, Wenjian Qiao, Lin Hong, Yonghong Li and Manqing Huang.

I am very fortunate to have a family that has always been encouraging and supportive. My parents have always given me more than I could ever ask for. My

brothers have always contributed and supported whatever I set out to accomplish. I will always be grateful to them.

Finally, I would also like to thank my wife, Xiaojie Dong, for all her love, support and encouragement. She is always my motivation to go further and do better. Her excellence pushed and motivated me enormously to work hard to avoid being too much inferior to her.

# TABLE OF CONTENTS

<b>1</b>	<b><u>MOBILE TECHNOLOGY OVERVIEW AND TRENDS</u></b> .....	<b>1</b>
1.1	<u>OVERVIEW OF MOBILE TECHNOLOGIES</u> .....	1
1.1.1	<i>Cellular Telephone Networks</i> .....	1
1.1.1.1	<u>First-generation cellular networks</u> .....	2
1.1.1.2	<u>Second-generation cellular networks</u> .....	3
1.1.1.3	<u>Data transmission in 1G and 2G cellular networks</u> .....	4
1.1.1.4	<u>2.5 G cellular networks</u> .....	4
1.1.1.5	<u>3G cellular networks</u> .....	6
1.1.2	<i>Satellite systems</i> .....	6
1.1.3	<i>Wireless local area systems</i> .....	7
1.2	<u>MOBILE DEVICES AND APPLICATIONS</u> .....	9
1.3	<u>TRENDS OF MOBILE TECHNOLOGIES AND APPLICATIONS</u> .....	10
1.3.1	<i>Wireless becomes preferred method</i> .....	10
1.3.2	<i>Real-time and multimedia applications become popular</i> .....	11
1.3.3	<i>Multiple networks coexist</i> .....	11
1.3.4	<i>Universal cooperation eventually but not in short term</i> .....	12
<b>2</b>	<b><u>INTRODUCTION: SUPPORTING MOBILE APPLICATIONS</u></b> .....	<b>13</b>
2.1	<u>SUPPORT REQUIRED BY MOBILE APPLICATIONS</u> .....	13
2.2	<u>OVERLAY NETWORKS</u> .....	14
2.3	<u>SUPPORTING INFRASTRUCTURE FOR MOBILE APPLICATIONS (SIMA)</u> .....	15
2.3.1	<i>Handoff support in overlay networks</i> .....	16

2.3.2	<i>Resource reservation</i> .....	18
2.3.3	<i>Server selection for replicated service</i> .....	19
2.3.4	<i>Secure multicast support</i> .....	20
2.3.5	<i>Relationship of SIMA Components</i> .....	21
2.4	<u>ORGANIZATION OF THIS DISSERTATION</u> .....	22
2.5	<u>SUMMARY OF CONTRIBUTIONS</u> .....	22
<b>3</b>	<b><u>QOS IN WIRELESS NETWORKS</u></b> .....	<b>24</b>
3.1	<u>BACKGROUND</u> .....	24
3.2	<u>RELATED WORK</u> .....	26
3.2.1	<i>Intserv and Diffserv</i> .....	26
3.2.2	<i>RSVP: Resource ReSerVation Protocol</i> .....	28
3.2.3	<i>Mobile IP</i> .....	29
3.2.4	<i>Cellular IP</i> .....	31
3.2.5	<i>BARWAN</i> .....	32
3.2.6	<i>Other Related work</i> .....	34
3.3	<u>WRSVP: WIRELESS RESOURCE RESERVATION PROTOCOL</u> .....	34
3.3.1	<i>Solving poor link problem</i> .....	35
3.3.2	<i>Solving handoff problems</i> .....	38
3.3.3	<i>Improving wireless bandwidth utilization</i> .....	40
3.3.4	<i>Deployment Issues of WRSVP</i> .....	41
3.3.4.1	<i>Cooperating with regular RSVP components</i> .....	41
3.3.4.2	<i>Using base stations to help transmitting handoff control packets</i> .....	43
3.3.5	<i>Discussion</i> .....	44

3.3.5.1	<u>Querying MH position from routing daemon</u> .....	44
3.3.5.2	<u>Properties of WRSVP</u> .....	45
3.3.6	<u>Required support from handoff protocol</u> .....	46
3.4	<u>HOPOVER: HANDOFF PROTOCOL FOR OVERLAY NETWORKS</u> .....	47
3.4.1	<u>System model and overall approaches</u> .....	48
3.4.1.1	<u>Beacons</u> .....	49
3.4.1.2	<u>Authentication</u> .....	50
3.4.1.3	<u>Pre-resource reservation using WRSVP</u> .....	51
3.4.1.4	<u>Cooperating with Mobile IP and packet forwarding</u> .....	51
3.4.2	<u>Handoff process</u> .....	52
3.4.2.1	<u>Handoff prepare</u> .....	52
3.4.2.2	<u>Handoff</u> .....	54
3.4.2.3	<u>Updating Mobile IP information</u> .....	56
3.4.3	<u>Utilizing HOPOVER for horizontal handoffs</u> .....	57
3.5	<u>PERFORMANCE EVALUATION</u> .....	58
3.5.1	<u>Simulation environment</u> .....	58
3.5.2	<u>Soft and hard handoffs</u> .....	60
3.5.3	<u>Sound file and sound player</u> .....	61
3.5.4	<u>Metrics and parameters</u> .....	62
3.5.5	<u>Simulation results and analysis</u> .....	63
3.5.5.1	<u>Soft handoff simulation results</u> .....	63
3.5.5.2	<u>Hard handoff simulation results</u> .....	66
3.5.6	<u>Overhead analysis</u> .....	66

3.6	<u>DISCUSSION</u> .....	68
3.6.1	<u>Handoff among more than two networks</u> .....	68
3.6.2	<u>Using separate HOPOVER and WRSVP component</u> .....	69
3.7	<u>CONCLUSION</u> .....	70
<b>4</b>	<b><u>SERVER SELECTION IN WIRELESS NETWORKS</u></b> .....	<b>72</b>
4.1	<u>BACKGROUND</u> .....	72
4.2	<u>RELATED WORK</u> .....	74
4.2.1	<u>Server side approaches</u> .....	74
4.2.2	<u>Client side approaches</u> .....	74
4.3	<u>S3 - SMART SERVER SELECTION</u> .....	76
4.3.1	<u>Utilizing DNS server and routing metrics in server selection</u> .....	76
4.3.2	<u>Collecting routing metrics</u> .....	79
4.3.2.1	<u>Query the gateway router</u> .....	79
4.3.2.2	<u>Query the directly attached router</u> .....	80
4.3.2.3	<u>Comparison of the two approaches</u> .....	81
4.3.3	<u>DNS extensions</u> .....	82
4.4	<u>SUPPORTING SERVER SELECTION IN SIMA</u> .....	82
4.4.1	<u>Backbone part</u> .....	83
4.4.2	<u>Mobile host part</u> .....	83
4.5	<u>DISCUSSION</u> .....	84
4.5.1	<u>S3 is both static and dynamic</u> .....	84
4.5.2	<u>S3 is suitable for mobile clients</u> .....	84
4.5.3	<u>Server load consideration</u> .....	85



4.5.4	<i>S3 is scalable</i> .....	86
4.5.5	<i>S3 enhances fault tolerance</i> .....	86
4.6	<b>PERFORMANCE EVALUATION</b> .....	86
4.6.1	<i>Simulation setup</i> .....	87
4.6.2	<i>Simulation results</i> .....	88
4.6.3	<i>Overhead analysis</i> .....	92
4.7	<b>CONCLUSION</b> .....	93
<b>5</b>	<b><u>WIRELESS SECURE MULTICAST</u></b> .....	<b>95</b>
5.1	<b>BACKGROUND</b> .....	95
5.1.1	<i>Multicast routing protocols</i> .....	96
5.1.2	<i>Issues in secure multicast</i> .....	98
5.1.3	<i>Secure Transmission Backbone</i> .....	100
5.1.4	<i>Supporting secure multicast in wireless networks</i> .....	100
5.2	<b>CURRENT SECURE MULTICAST SOLUTIONS</b> .....	101
5.2.1	<i>Iolus</i> .....	102
5.2.2	<i>Hierarchical Tree Approach</i> .....	104
5.2.3	<i>Core Based Tree</i> .....	105
5.3	<b>SECURE TRANSMISSION BACKBONE</b> .....	107
5.3.1	<i>Public STB</i> .....	108
5.3.2	<i>Private STB</i> .....	111
5.4	<b>APPLYING STB IN WIRELESS NETWORKS</b> .....	112
5.4.1	<i>Supporting Public STB</i> .....	112
5.4.2	<i>Supporting Private STB</i> .....	114

5.5	<u>HOW SIMA QoS COMPONENTS HELP STB</u> .....	116
5.6	<u>DISCUSSION</u> .....	116
5.6.1	<i><u>Attributes of STB</u></i> .....	116
5.6.2	<i><u>The benefits of STB in wireless networks</u></i> .....	119
5.7	<u>CONCLUSION</u> .....	120
<b>6</b>	<b><u>CONCLUSION AND FUTURE WORKS</u></b> .....	<b>123</b>
6.1	<u>RESOURCE RESERVATION</u> .....	123
6.2	<u>HANDOFF SUPPORT</u> .....	124
6.3	<u>SERVER SELECTION</u> .....	126
6.4	<u>SECURE MULTICAST</u> .....	127
6.5	<u>FINAL COMMENTS AND FUTURE WORKS</u> .....	128
<b>7</b>	<b><u>REFERENCES</u></b> .....	<b>131</b>

## LIST OF TABLES

<u>Table 1 WRSVP Options</u> .....	45
<u>Table 2 WRSVP Parameters</u> .....	45
<u>Table 3.HOPOVER Metrics and Parameters</u> .....	62
<u>Table 4 Route Metrics of Yahoo Replicated Services</u> .....	88
<u>Table 5 Metrics for Evaluating Key Management Problem Solutions</u> .....	102
<u>Table 6. Comparison of Key Management Solutions</u> .....	107

## LIST OF FIGURES

<u>Figure 1. Cellular Phone Networks</u> .....	2
<u>Figure 2. HomeRF Protocol Stack</u> .....	8
<u>Figure 3. Overlay Network</u> .....	15
<u>Figure 4. Horizontal and Vertical Handoff</u> .....	17
<u>Figure 5. Mobile IP Process</u> .....	30
<u>Figure 6. Cellular IP</u> .....	32
<u>Figure 7. Handoff Process in HOPOVER (part1)</u> .....	55
<u>Figure 8. Handoff Process in HOPOVER (part2)</u> .....	56
<u>Figure 9. Simulation Topology</u> .....	59
<u>Figure 10. Effect of Playback Buffer in soft handoffs</u> .....	64
<u>Figure 11. Effect of Remote host distance in soft handoffs</u> .....	64
<u>Figure 12. Effect of Playback Buffer in hard handoffs</u> .....	65
<u>Figure 13. Effect of Remote host distance in hard handoffs</u> .....	65
<u>Figure 14. Hops vs Simulation Length</u> .....	90
<u>Figure 15. Hops vs <math>TTL_{DR} (\lambda_H = 1/60)</math></u> .....	90
<u>Figure 16. Hops vs <math>TTL_H (\lambda_H = 1/60)</math></u> .....	91
<u>Figure 17. Latency vs Simulation Length</u> .....	91
<u>Figure 22. Iolus Structure</u> .....	102
<u>Figure 23. Hierarchical Tree Approach</u> .....	105
<u>Figure 24. CBT Structure</u> .....	106
<u>Figure 25. STB Data Packet Format</u> .....	109
<u>Figure 26. STB Transmission Sequence</u> .....	110
<u>Figure 27. Reduce Number of Encryptions/decryptions</u> .....	118

# **1 Mobile Technology Overview and Trends**

This dissertation is about providing better help for mobile applications. It is necessary for us to have a good understanding of the current status and trends of mobile services. In this chapter, we first review the current technologies, standards and proposals in this area. Then, based on our observation and analyses, we forecast the trends in future mobile services and applications.

## **1.1 Overview of Mobile Technologies**

Current wireless networks can be categorized into two types. The first type operates over the distance of hundreds of meters, kilometers or even more. Examples of this category include cellular telephone voice and data networks and satellite systems. The other type of networks are designed to provide LAN connectivity over much shorter distances, typically in meters and usually within a building or even a room. Examples include IEEE 802.11 wireless LAN, HomeRF and Bluetooth.

### **1.1.1 Cellular Telephone Networks**

In the wide area side, cellular telephone networks have evolved through 1G (analog systems), 2G (digital systems), and 2.5G (transitional systems) generations. For each generation, there are many networks in use. The new generation 3G has not been defined completely and only one network of this category are in operation by the time of this writing.

In cellular telephone networks, networks are organized into cells to enable radio frequency reuse. Any neighboring cells cannot shared the same frequency, but same

frequency can be reused in two cells which are not next to each other. The word cellular in the name comes directly from this design. Figure 1 represents this idea.

### 1.1.1.1 First-generation cellular networks

First-generation cellular networks are analog systems. When a conversation is started, the system allocates an unused frequency channel for it. During the whole conversation process, the allocated channel is dedicated to that conversation only; no other signal is transmitted on it. If there is no channel available, the users have to wait and try later.

Advance Mobile Phone System (AMPS) [26] is the representative solution, and is the only unified standard in North America. Any telephone that needs to support US nationwide roaming must have AMPS capability. Therefore, in many cases, the phone has two modes, one to be used for primary digital service and the other for AMPS.

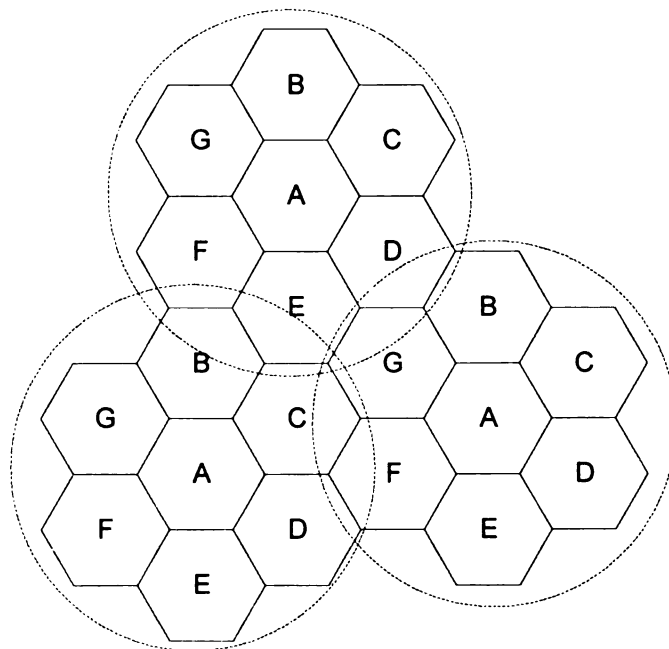


Figure 1. Cellular Phone Networks

AMPS uses frequency division multiple access (FDMA) as the transmission method. An AMPS system consists of two sets of channel pairs, each contains 416 30KHz-wide receive/transmit channel pairs. Control information is transmitted on the lowest 21 channels in each block, at the rate of 10Kbps. The control information includes specifics such as telephone number called, system ID, and the transmission power setting.

### **1.1.1.2 Second-generation cellular networks**

Second-generation cellular systems use digital transmission. Conversations are sampled into digital signals, compressed, and then sent. By taking advantage of digital technologies such as compression, second-generation systems greatly enhance the efficiency of frequency channel use. Frequency channels can now be shared among multiple conversations, thus more conversations can be carried on the system.

Major 2G systems use TDMA (Time Division Multiple Access) or CDMA (Code Division Multiple Access). TDMA divides each frequency channel into time slots, and assigns multiple digital signals onto different time slots. This way, each frequency channel can carry multiple conversations/data transmissions at the same time. Global System for Mobil Communication (GSM) is a representative TDMA solution and the dominant standard in Europe, Asia(except Japan and Korea), Australia, Africa and the middle East. GSM divides each channel into eight time slots, allowing four simultaneous calls, each using one slot for reception and one slot for transmission. [79]

In CDMA, different codes are used to identify different sessions (calls or data transmissions). All sessions within a cell are sent at the same time on the same range of frequencies. Each session occupies nearly the entire bandwidth. This results in many

overlapping signals. The receiver needs to apply the correct code to the signal to retrieve the real data. The representative CDMA solution is IS-95, also called cdmaOne.

### **1.1.1.3 Data transmission in 1G and 2G cellular networks**

1G and 2G networks were designed primarily for voice conversations, but they can also be used for data transmissions. All 1G and 2G networks allow data to be sent using a modem. To the network, such data transmissions are of no difference with voice conversations. Despite the simplicity, sending data in such way causes inefficient use of spectrum compared with a packet data system. Also, the user is not online all the time and must dial in to gain access.

Cellular Digital Packet Data (CDPD) is a technology designed to support packet data transmission over AMPS cellular systems. [78] CDPD detects idle channels and transmits data over them. If voice activity is detected anytime during data transmission, CDPD interrupts the data transmission, scans for another channel and continues the data transmission over the new channel. The data rate supported by CDPD is up to 19.2 Kbps.

For 2G systems, no equivalent system was designed. Packet data on the digital cellular networks begins directly with 2.5G.

### **1.1.1.4 2.5 G cellular networks**

The data rates supported in 2G systems are around 14.4 Kbps. To achieve higher data rates, several technologies have been proposed. They are called 2.5 systems because they extend the existing 2G standards. Important examples include IS-95B and IS-95C on the CDMA side, and General Packet Radio Service (GPRS) on the TDMA side.



IS-95B and IS-95C extends the current CDMA systems (IS-95A) to achieve data rates of 64Kbps and 144Kbps respectively. The higher data rates are achieved by replacing the overhead of voice channel with a packet-based system and by using more efficient coding schemes. Unlike CDPD, IS-95B and IS-95C are not overlay mechanisms. Instead, they fit into the system and use available bandwidth for transmission. That means, cooperation between existing CDMA system and IS-95B/IS-95C is required in bandwidth division and transmission scheduling. In the case of CDPD, no such requirement is necessary, because CDPD detects and uses unused channels and it is transparent to the original AMPS system.

GPRS was designed to enable current GSM systems to provide packet-based applications and services. [68] To obtain higher data rates, GPRS uses multiple time slots of GSM channels, with a target data rate of 115.2Kbps. GPRS requires modification to the GSM protocol stack to accept, interpret and reassemble data fragments. In the infrastructure part, a new, purely IP based backbone is introduced. The most important parts of the new backbone are Serving GPRS Support Node (SGSN) and Gateway GPRS Support Node (GGSN). SGSN is used to connect the GPRS backbone to the base stations, and it handles mobile device mobility and authentication. GGSN connects the GPRS backbone to the external data networks (the Internet), and it also performs accounting of users' resource usage. In addition to the backbone change, GPRS also requires more computation power on mobile units for the processing of additional time slots.

### **1.1.1.5 3G cellular networks**

3G standards have not been completely defined yet. [65][66][67][68] The two primary candidates are based on CDMA: Wideband CDMA (W-CDMA) and CDMA2000. Both of them introduce higher chip rates to obtain higher data rates. Chips are the smallest element of data in an encoded signal, even smaller than a bit. For CDMA2000, it allows chip rates from 1.2288 megachips per second (Mcps) to 14.7456Mcps. For W-CDMA the range is 4.096 Mcps to 16.384Mcps. The target data rates are 384Kbps to 2Mbps or even higher.

W-CDMA is planned to be backward compatible with GSM networks, while CDMA2000 is designed to be backward compatible with IS-95B. Both proposals are endorsed by International Telecommunication Union (ITU) as part of the vision definition standard for 3G cellular systems: International Mobile Communications-2000 (IMT-2000). It is likely that both proposals will be widely used in the future, and users may need dual-mode wireless units to ensure service in all locations.

### **1.1.2 Satellite systems**

Besides cellular systems, satellite systems provide another important way for wide area wireless voice and data connection. Many satellite systems were designed to provide global connection and saw cellular systems as territorial solutions. Due to the perception that global travelers will eventually be using a globally available 3G infrastructure supported by major wireless carriers, this assumption is no longer very valid. Still, the global mobile satellite data market is expected to grow rapidly.

An interesting satellite system is Global Positioning System (GPS), which is implemented and operated by the US Dept. of Defense. What makes it famous is its

widely use as a navigation system for commercial purposes. The system consists of 24 medium earth orbit satellites. By extracting orbit and timing information from multiple satellite signals in view (the arrangement of the satellites guarantees that 5-8 satellites are in view at any position of the earth), mobile units are able to calculate its precise location. It is forecasted that all future wireless phones will contain a GPS enabled positioning system.

### **1.1.3 Wireless local area systems**

As we mentioned earlier, cellular telephone voice and data networks and satellite systems are designed to operate over the distance of hundreds of meters, kilometers or even more. For short distances, such as the distance of meters and usually within a building or even a room, numerous solutions/products/standards have been proposed or deployed. We describe briefly the representative ones including IEEE 802.11 wireless LAN, Home Radio Frequency (HomeRF), and Bluetooth.

IEEE 802.11 wireless LAN standard was first approved in 1997. [69][74][75] Products providing data rates at 11Mbps are already widely available from companies such as Cisco (by acquiring Aironet), Lucent, Nortel and others. 802.11 products operate in the unlicensed 2.4GHz frequency band, using Direct-Sequence Spread Spectrum (DSSS). 802.11 was designed to work as a counterpart of 802.3 in the wireless environment to interconnect data devices such as PCs and printers. Compared with ISO 7-layer reference model, the standard includes specifications in both physical and data link layers. The default upper layer interface is IP.

HomeRF [70] is a competing technology supported by Compaq, Intel, Microsoft and other companies. The design goal of HomeRF is to interconnect home devices such as PCs, phones, and PDAs in short distance. Similar to 802.11, HomeRF consists of specifications for physical and data link layers. But HomeRF extends its data link layer to support DECT (Digital Enhance Cordless Telephone) standard. This way, HomeRF supports both data and phone connections through IP and DECT respectively. Figure 2 shows the protocol stack of HomeRF. Similar to 802.11, HomeRF can provide data rates up to 11Mbps. But in physical layer, HomeRF chooses to use Frequency-Hopping Spread Spectrum (FHSS).

Another big player in the short distance wireless communication area is Bluetooth [71], which is a specification for small-form factor, low-cost, short range radio links between mobile PCs, mobile phones and other portable devices. Unlike IEEE 802.11 and HomeRF, there is usually no infrastructure part in Bluetooth environment. Devices are expected to work in a purely ad hoc mode. Bluetooth simply enable devices to communicate without the need of wires. Comparing with the other two technologies, Bluetooth works in a much shorter distance (less than 10 meters compared to about 50

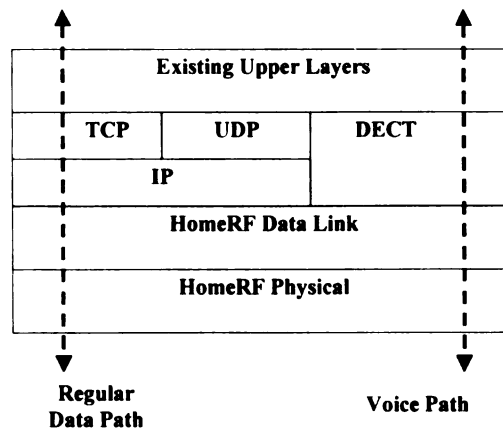


Figure 2. HomeRF Protocol Stack

meters for 802.11 and HomeRF) and supports fewer devices concurrently. Bluetooth also makes use of FHSS.

Currently, 802.11, HomeRF and Bluetooth cannot coexist with each other, thus makes it very difficult for customers to make decision in purchasing. IEEE formed the 802.15 committee in resolving the conflicting issues.

## **1.2 Mobile devices and applications**

Traditionally, wireless networks are used for precisely defined narrow applications. For example, cellular phone networks and most satellite systems are primarily used to carry voice traffic, and wireless LANs are used to connect home or office data devices such as PCs and printers in short range.

With the Internet's explosive growth, people are getting used to utilize the Internet for all kinds of personal and professional needs: shopping, banking, entertaining, reading news and books, searching for references, downloading tools, and certainly, gaming. The enthusiasm of being able to access the Internet anytime and anywhere increases exponentially. To meet such needs, the capacity of wireless networks continually to be enlarged and more and more Internet applications are being introduced. The result is, wireless networks are no longer used for special purposes such as voice conversation. Instead, they are used for all kinds of Internet applications.

The great popularity of mobile devices and applications can be proven by the following numbers. In European Union countries, the mobile penetration in 2004 is expected to be 3-4 times of the Internet penetration. Similar trends appear in USA and Asian pacific countries. The transaction amount performed on mobile devices is expected to be more than 1.4 trillion US dollars by the year 2003. And wireless service providers

already spent tens of billions of dollars in purchasing spectrum and upgrading their networks.

To satisfy the need of accessing information at any time and anywhere, a wide array of new mobile devices have been designed and introduced to the market. Also, many devices that previously used only analog technologies have adopted digital technologies in order to enhance capacity (for example, through digital compression), service quality and ease of storage. Current popular mobile devices include notebook PCs, Cell phone handsets, Personal Digital Assistants (PDAs) and other special purpose devices such as BlackBerry wireless Email. Two trends are common for almost all those mobile devices: 1) they are becoming in possession of greatly increased processing power; and 2) they are becoming increasingly capable to access the Internet, and at high speed (with advance both in mobile devices and network capacity). As the direct consequence, more demanding applications can be supported, especially those with real-time and multimedia requirements.

### **1.3 Trends of mobile technologies and applications**

With observation and analysis of mobile technologies, services and applications, we foresee a number of trends which will be reflected in the next several years.

#### **1.3.1 Wireless becomes preferred method**

Wireless technologies were viewed as expensive alternatives to wireline and used only when wireline services are not available. With advances in wireless networks and mobile device technologies, we begin to view wireless technologies as a cost-effective

alternative to wireline. For many cases, wireless solutions are becoming the preferred method even when wireline is available, due to the unmatched convenience.

Factors contributing to such transition include: enlarged network capacity, higher data rates, lower cost of network access, availability of inexpensive and powerful mobile devices and diversified mobile applications.

### **1.3.2 Real-time and multimedia applications become popular**

With the development of wireless networks and satellite technology, currently it is possible in most places to connect mobile devices to a local, regional or international network. In addition, higher data rates and easier access methods are continually introduced. At the same time, mobile devices are becoming increasingly powerful with advances in processor, display and accessories. Naturally, it becomes possible to support high-demanding applications with real-time and multimedia requirements, such as video conferencing, real-time stock purchasing, music and video streaming and multicast web seminar participating. In reverse, the high demand of such applications will drive the transition in networks and mobile devices to happen quicker.

### **1.3.3 Multiple networks coexist**

Due to decreasing cost and increasing demand and competition, it will be common that multiple networks will coexist to serve different purposes. Different networks will be created for local, regional and international coverage. The competition among vendors will result in the creation of multiple networks serving the same geographic areas. Also, multiple wireless LANs could coexist, serving different business entities or needs.

#### **1.3.4 Universal cooperation eventually but not in short term**

The most significant feature ensuring the success of the Internet and Web is that data and information can be exchanged effectively among computers, operating systems and applications by different manufactures. Currently, the cooperation among different wireless networks is still very limited, although several standardization efforts are in process. For wireless networks, due to the huge market and complicated factors such as government support, a single standard enabling effective information exchange is not likely to be available in the near future. However, in long term, a unified standard is unavoidable. With such a standard, mobile devices will be able to exchange information effortlessly and roam to other networks seamlessly.



## 2 Introduction: Supporting Mobile Applications

In concluding the trends of wireless networks, we found that diversified, highly demanding applications will be running on heterogeneous and geographically coexisting networks. The theme of this dissertation is to provide corresponding support. In this dissertation, we present a new structure that enables close cooperation among wireless networks and provides a set of backbone support for mobile applications.

### 2.1 Support Required by Mobile Applications

The support required by mobile applications is from multiple aspects. We discuss some of the most important ones here. First, real-time applications require packets to be transmitted within acceptable packet loss rate, latency and jitter. Such requirements are generally called *quality of service (QoS)* requirements. [4][57][61][62][65][66][67][68] Second, since many applications involve confidential information, there are security requirements. Security requirements can be further divided into several cases. Secrecy and authentication are the most important. [7][23][24][25][27] Third, many applications are not in the one to one mode. Instead, they relate to multiple parties. In such cases, multicast support is required. [17][18][2]

Many times, support from multiple aspects is desired at the same time. For example, confidential video conferencing requires support in all the three categories mentioned above. Data needs to be multicasted to multiple parties, securely, and within acceptable delay, jitter and packet loss rate.

Compared to wired networks, wireless networks possess some inherent properties that make it more difficult to meet such requirements. In wireless networks, usually, there

are no abundant resources, including processing power, buffer, bandwidth and others. Wireless links are subject to much higher *bit-error-rate (BER)* than wired links, so transmissions in wireless networks are more prone to packet losses and attacks. Wireless devices are often in movement and such movements cause changing network topology, lost packets, and sometimes even broken connections.

To overcome these difficulties and provide satisfactory support for real-time and multimedia applications in wireless networks, we propose a number of new mechanisms in this dissertation.

## **2.2 Overlay Networks**

One of the trends of wireless networks is that multiple networks will coexist geographically. This is due to the decreasing cost and increasing demand and competition. Different networks are created for local, regional and international level of coverage. The competition among vendors will result in the creation of multiple networks in each level, and serving the same geographic areas. Also, multiple wireless LANs could coexist, serving different business entities or needs.

We refer to such geographically collocating, multiple wireless network structures as *overlay wireless networks* (this term was first used in [5] and [6]). For example, a high bandwidth office/lab wireless LAN could be overlaid with a more moderate bandwidth building wireless LAN network. Or a building wireless LAN could be overlaid with a PCS system, corresponding to local and wide areas connectivity. Figure 3 illustrates this concept.

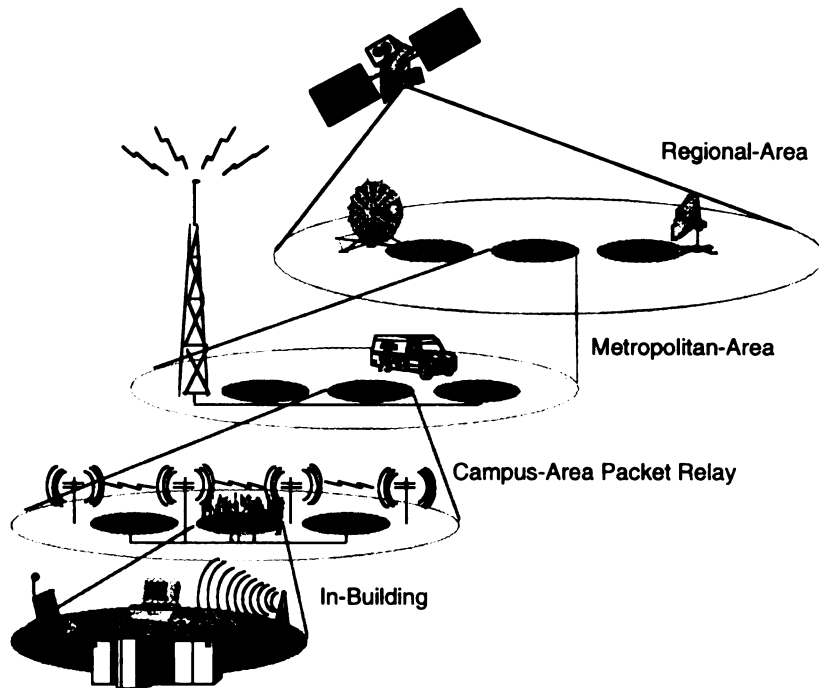


Figure 3. Overlay Network<sup>1</sup>

## 2.3 Supporting Infrastructure for Mobile Applications (SIMA)

Given the current and future status of wireless networks, there clearly is the need of cooperation between networks, especially in overlay networks. Such need is more urgent than that of wired networks. Wired networks can be considered as flat, because most machines participate only in one network and they never move into another network. The only contact points of different networks are at routers. In the case of wireless networks, each node can participate in multiple networks, and since they move, the contact points or common members of different networks could be any mobile nodes. In the case of wired networks, no cooperation between two networks means that the member of one network cannot access information from the other. Since they do not

---

<sup>1</sup> Figure adopted from [6]

move, they are still connected to and can work with the other members of the same network. However, in the case of wireless networks, two incompatible networks could result in total disconnection of mobile nodes which roam to the area of the other network.

In this dissertation, we present Supporting Infrastructure for Mobile Applications (SIMA). SIMA is IP-based and it enables the internetworking of different and heterogeneous wireless networks. The result is, mobile users could roam smoothly between networks and without worry about disconnected connections or services. As we discussed earlier, future mobile applications will be diversified. And many of them will be multimedia and real-time. Designed to support such highly demanding applications, SIMA include the following functions:

- Handoff support
- Resource reservation
- Secure multicast
- Application server selection

We briefly describe each of SIMA' functions in the next. The detailed discussion will be provided in later chapters.

### **2.3.1 Handoff support in overlay networks**

First and most importantly, SIMA enables wireless networks to be closely connected. This is represented by the support of overlay network handoffs. *Handoff* is the process in which a wireless device moves from current cell to a new one. The performance of handoff scheme directly affects the overall performance of mobile applications. The importance of handoff performance keeps going up as today's wireless networks have increasingly large user population and decreasing cell size.

Traditionally, the term handoff is used to refer to the case in which the two cells involved belong to the same wireless network, as shown in the left side of Figure 4. As we discussed earlier, the future wireless networks will be overlaid with each other to form hierarchies. In such structures, mobile devices could perform handoffs within the same networks as well as among different layers of networks. Handoffs of the former case are referred as *horizontal* and those of the later case are referred as *vertical*. Vertical handoff is shown in the right side of Figure 4

It is desirable that a handoff scheme enables *mobile hosts (MH)* to move among cells smoothly and causes little or no interference on mobile applications. A lot of research work has been done to improve the performance of horizontal handoffs, for example [32][19][33][34][35][36][37][38][39][40][41] Many solutions can provide satisfactory performance. However, much less effort has been spent on vertical handoff research. Up to now, most networks still support vertical handoff through Mobil IP [30], which was designed to provide macro-level and slow moving mobility. The scheme

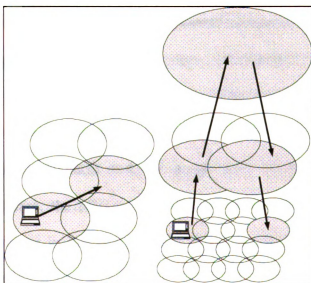


Figure 4. Horizontal and Vertical Handoff

suffers from high overhead when high frequency handoffs are needed. More importantly, handoffs cause big gaps in packet flows under this scheme. Besides mobile IP, there are only few other approaches and none of them got extensive use due to their limitations and shortcomings.

To address this problem, we designed a new overlay handoff protocol named *HOPOVER (HandOff Protocol for OVERlay networks)*, which supports both horizontal and vertical handoffs and is compatible with mobile IP. The main approaches used in this protocol include: facilitating WRSVP (discussed below) to pre-reserve resources in the new cell and along the path from the new cell to the flow transmission parties; authenticating in advance of the actual handoff; buffering in the new network for the MH and forwarding packets from the old network to the new network. With these measurements, HOPOVER significantly enhances handoffs performance.

### **2.3.2 Resource reservation**

To guarantee QoS of real-time flows, flows must be allowed to reserve network resources. Resource ReSerVation Protocol (RSVP) [57][58][59][60] is a widely used resource reservation protocol designed for wired networks. But in wireless networks, there are two big problems preventing the direct use of RSVP. One is poor link problem and the other is handoff problem. Poor link problem stems from the high bit error rate associated with wireless links. Such poor links cause difficulties in reserving and refreshing resources. Handoff problem occurs when the MH moves between cells. Without special care, the MH could suffer terminated flows due to un-established resource reservations in the new cell and along the path from the new cell to the transmission parties.

We integrated into SIMA structure a protocol named *Wireless RSVP* (WRSVP) [1]. WRSVP enable mobile devices to reserve resources, and the scheme is compatible with RSVP. We solve the poor link problem by separating resource reservation into wired and wireless parts and by adding Ack messages into wireless message exchanges. Using pre-reserving in neighbor cells, WRSVP protects flow reservations from handoffs, both in the cell and along the path. Resource reservation becomes a natural part of handoff.

### **2.3.3 Server selection for replicated service**

Replicated servers are often used for popular services with large amount of visits. By putting replicated servers at appropriate places, replicated servers increase network proximity and thus reduce access latency perceived by users. A fundamental issue for a global-scale replicated service is how the best server is selected based on the user's preference.

Nearly all the existing server selection schemes were designed with the assumption of a fixed topology. But in wireless networks, the network topology changes as clients move around. Naturally, such changing topology poses difficulties in server selection. Also, many existing solutions require the clients to participate in choosing the best server. For clients in wireless network, the lack of computing and bandwidth resources sometimes makes such participation a difficult task.

In [3], we presented a new server selection scheme, *Smart Server Selection* (S3), which utilizes client side DNS and routing information in choosing the best server. It includes extensions to routers and DNS, so that DNS can collect routing metrics from routers and select the best server for clients. S3 helps the mobile devices to select the best server according to metrics such as distance, delay and cost, thus effectively enhances the

quality of service perceived by the clients and reduces the overhead on both the network and application servers.

In SIMA, we extend S3 server selection mechanism into wireless networks. For static clients, in addition to regular S3 support, the clients' current session statuses are kept in SIMA servers. As a handoff happens, the status information is also transferred to the new SIMA server. Such information is used in the new cell to perform background server selection, then, the MH is notified about better server selections. This server selection scheme involves primarily backbone components, which saves valuable wires resources. What is more, the server selection process involves no remote information exchange, which makes the process fast and reliable.

#### **2.3.4 Secure multicast support**

*Secure multicast* is the inter-network service that securely delivers data from a source to multiple authorized receivers. Not only multicast has been proven to be resource efficient, but also many applications involve multiple parties by nature. Thus multicast is logically more natural for such applications. Example applications include audio and video conferencing, software update distribution, and stock market information services.

A key challenge of secure multicast lays in the handling of membership and encryption key information for large multicast groups which could have thousands or millions of members. Such groups have extremely high frequency membership-changes, and each time membership change occurs, the encryption key needs to be replaced to maintain security. This re-key process has to be scalable, reliable and highly secure.



In addition to such common issues found in both wired and wireless networks, In wireless networks, new issues arises due to problems imposed by wireless networks' inherent attributes: high bit error rate, frequent handoff and limited resources. High bit error rate limits the frequency of re-key process, because mobile devices need to be given extra time for possible packet re-delivery. Frequent handoff requires the scheme to be highly scalable, otherwise it will fail in front of large population of users.

SIMA includes a secure multicast solution we proposed in [2]. In this method, a *Secure Transmission Backbone (STB)* is constructed. With such a backbone, it is no longer necessary for each individual multicast group to maintain keys. Key management problem is solved/avoided automatically. STB also successfully addresses the special issues of wireless networks. It greatly reduces the number of re-key processes and it limits most processes locally.

### **2.3.5 Relationship of SIMA Components**

The above four components of SIMA are closely related. Each component is designed to provide a specific function, but usually they are used in a combined manner. For example, mobile hosts could choose the best multicast source using S3, establish resource reservation using WRSVP, then join the multicast session using STB.

The cooperation is reflected especially apparently when a handoff happens. The handoff process is handled by HOPOVER. And in this process, HOPOVER provides corresponding support to ensure that WRSVP, STB and S3 daemons are aware of the happening of the handoff and they can perform adjustment accordingly. The support includes providing information available only to the handoff daemon, broadcasting information for other daemons in beacons and others. For example, WRSVP daemon

requests mobile hosts location information, which is only available from the handoff daemon. Another example, STB requests the public key and key certificate authority information to be included into *base station (BS)* beacons. The detail of how SIMA components cooperate with each other will be explained in detail in corresponding chapters.

## **2.4 Organization of This Dissertation**

The remaining part of the dissertation is organized as follows. Chapter 3 presents the main QoS components of SIMA: WRSVP, a resource reservation protocol for wireless networks, and HOPOVER, an overlay handoff protocol. In Chapter 4, another SIMA components contributing QoS, smart server selection (S3), is presented. In Chapter 5, we discuss how secure multicast is enabled using secure transmission backbone (STB), and finally conclusion is given in Chapter 6.

## **2.5 Summary of Contributions**

In this dissertation, a set of new mechanisms for supporting mobile applications are presented. Our work mainly includes the following independent yet closely related components:

- WRSVP: A resource reservation protocol for wireless networks
- HOPOVER: An overlay handoff protocol
- S3: A server selecting scheme using routing metric aware DNS servers
- STB: A generic secure multicast support scheme

The first component enables wireless devices to reserve necessary resources in wireless networks, which is a necessary requirement for QoS. WRSVP cooperates with

the standard RSVP, and it supports multicast. The contribution of WRSVP lays in its solution to the problems of poor links and frequent handoffs.

The second component enables wireless devices to move freely and smoothly in overlay networks. Existing schemes incur long packet flow gaps and cannot meet the requirement of real-time and multimedia applications. HOPOVER provides a satisfactory solution through the use of resource pre-reserving, fast authentication, packet forwarding and buffering.

The third component helps wireless devices in choosing the best server when it uses a replicated service. Traditional server selection solutions are hard to use in wireless networks because network topology is unstable from the point of view of both the servers and the clients. S3 enables accurate, fast and low overhead server selection by uniquely utilizing routing information combined with DNS server.

The last component enables wireless devices to participate in secure multicast sessions. STB effectively solves the key management problem and addresses several special issues found in wireless networks.

Put all the components together, we have a wireless network environment with solid support for mobile applications. Different needs of mobile applications are addressed by different SIMA component. When multiple components are needed, they cooperate closely to deliver the required results.

### **3 QoS IN WIRELESS NETWORKS**

In this chapter, we present the way SIMA supports QoS in wireless networks. QoS is the most important goal of SIMA. We have three components contributing to QoS assurance: a wireless resource reservation protocol, WRSVP, an overlay handoff protocol, HOPOVER and an application server selection scheme S3. While each of them has different focus, they closely work together in the SIMA structure. The first two components are presented in this chapter. The relatively more independent component for server selection will be discussed in next chapter.

#### **3.1 Background**

Traditionally, the Internet only provides best effort service. Traffic is processed as quickly as possible, but there is no guarantee as to timeline or actual delivery. With the widespread use of World Wide Web, QoS has become a necessity in delivery of real-time content and services.

According to the level of QoS requirement, applications can be divided into the following four categories: conversational, streaming, interactive and background. Conversational traffic requires the time relation between packets to be preserved. In addition, the delay and jitter should be controlled at very low level. An example of this class is voice conversation, and actually this is why this class is named as conversational. Streaming traffic also requires the relation between packets to be preserved, but there is no very stringent requirement on delay. An example of this class is streaming video. The next class, interactive traffic is for applications such as web browsing. There are still delay and jitter requirements, but they are much relaxed. The last class, background

traffic only requires the data to be delivered correctly. The destination is not expecting the data to be delivered within certain time. The example here is background file downloading.

Same applications and QoS requirement appears in wireless networks as well. But due to the special environment such as changing topology, high BER, lower bandwidth and limited processing power, it is more challenging to meet such requirements.

In wired networks, no matter with Intserv or Diffserv, QoS is essentially provided by allocating abundant resources (bandwidth, buffer, processing power etc.) to higher priority traffic. If only such resources are available, usually packets can be delivered as required.

In wireless networks, having enough resources is still critical for QoS. But in addition to that, now there is an equally important factor, which is the performance of handoffs. Without enough resources, no QoS can be talked about. Without a good handoff scheme, no QoS can be sustained after a handoff takes place.

Correspondingly, in SIMA, QoS assurance is provided primarily by two components: a wireless resource reservation protocol, WRSVP, and an overlay handoff protocol, HOPOVER. The third QoS component, an application server selection scheme, is supplemental to the first two components.

There are certainly other supports needed to guarantee QoS of mobile applications. For example, the ISP backbone should support either Intserv or Diffserv schemes to give better treatment of higher priority traffic. Also, special purpose components such as proxy servers may be necessary for some applications. In our

following discussion, we assume such support is already available, because in this dissertation, we are focusing on issues unique to wireless networks.

In the remaining part of this chapter, we first review related work in section 3.2. Then we present the design of WRSVP and HOPOVER in 3.3 and 3.4 respectively. In 3.5, we present the simulation results. Section 3.6 discusses issues and attributes of WRSVP and HOPOVER. Finally, conclusion is given in 3.7.

## **3.2 Related Work**

A lot of research work has been done in the area of providing QoS on data packet networks, both for wired and wireless cases. We go through some of the most representative ones.

### **3.2.1 Intserv and Diffserv**

The two primary models for providing QoS in the Internet are *Integrated services* (Intserv) and *differentiated services* (Diffserv). The essence of Intserv is to reserve resources such as link bandwidth and buffer space for each individual flow so that QoS can be guaranteed. The essence of Diffserv is to divide traffic into different classes and give them differentiated treatment, resulting in better chance of meeting QoS requirement for higher priority traffic.

In Intserv [61], there are four components: the signaling protocol (e.g. RSVP), the admission control routine, the classifier and the packet scheduler. To receive QoS, applications must set up the paths and reserve resources before transmitting packets. The admission control routines decide whether such a request can be granted. The classifier

resides on routers, and performs packet classification (based on header fields) on incoming traffic. The packet scheduler then schedules the packets accordingly.

In Diffserv [62][63], the original IPv4's "type of service" field is used to differentiate traffic. Before sending out packets, applications should mark them according to the desired service level. Or, ISP edge routers can mark them based on header fields. Then inside the ISP networks, packets are treated differently. Since the way packets are handled inside the network is unknown to users, ISPs have much flexibility in assigning resources inside their networks.

Intserv leaves more control to customers while Diffserv gives it to ISPs. With Intserv, the network only accepts traffic up to its capacity. Users will be able to know immediately if the desired service is available or not. With Diffserv, they only receive statistics from ISP about delivered traffic in pre-specified interval (hourly, daily or even longer). The difference is most apparent when the chosen ISP suffers problems with their network and becomes unable to provide required QoS. With Intserv, the application will go to another ISP immediately due to failed resource reservation. With Diffserv, the application will only go to another ISP when the degraded QoS is perceived.

We foresee in the near future, both Intserv and Diffserv will be widely used. From the users' point of view, there is not much requirement for Diffserv, because most of time, the traffic is classified at the ingress routers of the ISP. But for Intserv, the required support for signaling protocol (or resource reservation protocol) must be available from the users' backbone network.

### **3.2.2 RSVP: Resource ReSerVation Protocol**

Resource ReSerVation Protocol (RSVP) is a widely used resource reservation protocol designed for wired networks. RSVP is useful in both Intserv and Diffserv. [64]

RSVP uses “flowspec” to describe the required resources. Admission control looks at a new flow’s “flowspec” to decide whether or not to admit the flow. RSVP requires senders to periodically send Path messages to their receivers. In a Path message, the sender specifies the traffic characteristics of the flows it intends to send. These characteristics are described in the form of token bucket filters. While the Path message is being delivered, the routers along the path add their information to it. So, upon received by a receiver, the Path message contains the route information from the sender to the receiver. The receiver constructs a Reservation message, puts the token bucket filter and its acceptable level of delay into the message, and then sends it along the path indicated by the Path message (in reverse direction). If the network can handle this new flow, it will be accepted and each router along the path reserves the required resources for this flow. After that, the receiver periodically sends Refresh messages (contains similar information as Reservation message) to the network to refresh the reservation. In the case one or more routers along the route fail, the Path and Refresh messages will flow along a new path created/chosen by the routing protocol of the network, thus the reservation is retained.

RSVP is a nice solution to resource reservation. First, it is receiver initiated. It is possible for a receiver to reserve resources according to its own need and its local network condition. Second, it is easy to adjust the reservation dynamically according to the change of the network condition because reservation is periodically refreshed. Third, since only soft states are maintained in routers, it is not necessary to explicitly tear down



flows. Instead, they will time out eventually and resources will be released. Forth, RSVP supports multicast. Multiple senders can send to multiple receivers concurrently. RSVP builds a multicast tree for each sender. For multiple senders, these trees are merged. In many parts of these trees, receivers can share resources, so network utilization is improved. Finally, “it adapts to changing multicast group membership” [57], which means that it can automatically adjust the resources reserved in each router when flows join or leave the multicast tree.

However, as we discussed earlier, RSVP was designed for wired networks, and it is not straightforward to apply RSVP to wireless networks. Poor links and frequent handoffs make the resource reservations unstable and sometimes unavailable.

### **3.2.3 Mobile IP**

Traditionally, handoff schemes across networks are constructed based on Mobile IP [30][38]. Mobile IP provides a simple and efficient solution to maintain IP connections for MHs. It is a Proposed Standard Protocol by IETF, and it is widely used and supported by current systems.

Mobile IP handles mobility as shown in Figure 5:

- Every mobile host has a Home Agent (HA), which knows the MH’s permanent IP address.
- Every site that wants to allow visitors creates a Foreign Agent (FA).
- When a mobile host moves to a foreign network, it needs to register itself with the FA and get assigned a care-of address (Step 1).

- The FA notifies the HA of the mobile host's current IP address, or care-of address (Step 2). The FA also starts forwarding packets for the MH (Step 3).
- When packets are sent to the MH's home network (Step 4), the HA forwards them to the current foreign network. At the same time, the HA notifies the sender about the new location (Step 5.1 and 5.2).
- The sender then sends following packets directly to the new location (Step 6).

Mobile IP's design principle was to support macro level mobility and slow moving hosts. It requires that the HA be notified each time the MH receives a new care-

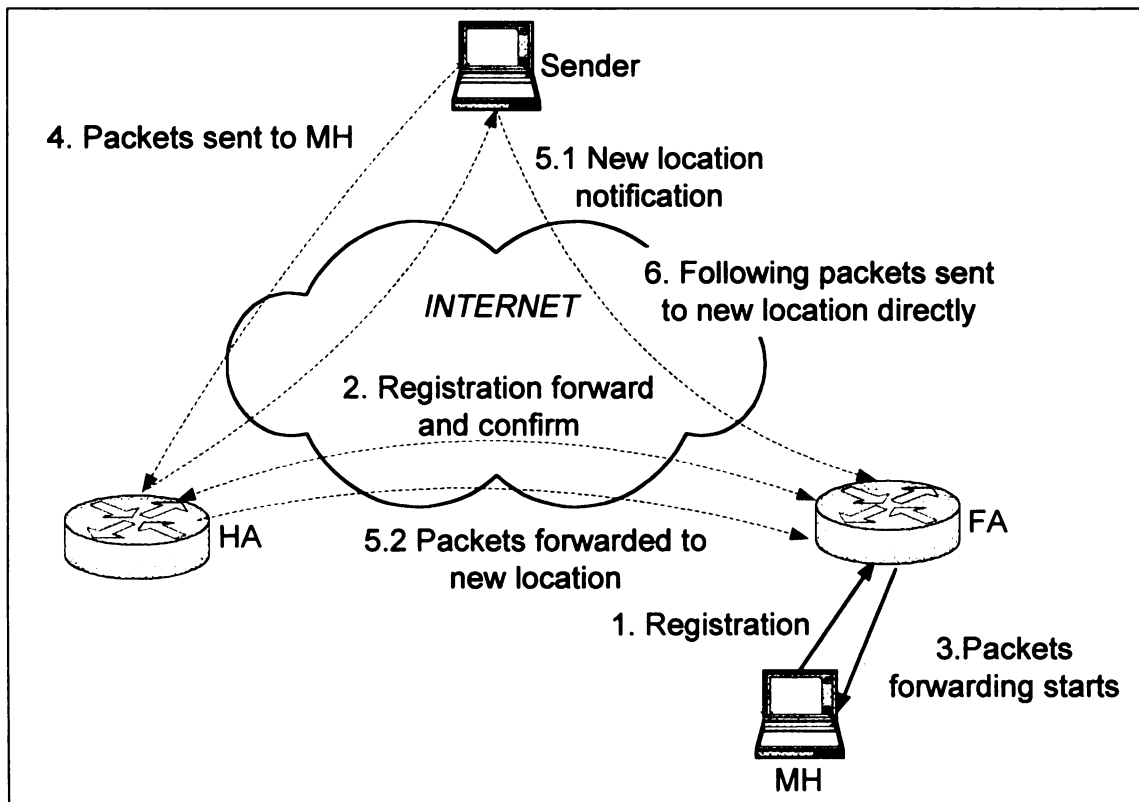


Figure 5. Mobile IP Process

off address, which happens when the MH moves into the foreign network or the MH performs a handoff. Such requirement is OK when MHs move at low speed and handoffs only happen at low frequency. However, as wireless communication evolves, the number of mobile devices increases rapidly, and the use of high-speed moving mobile devices becomes more and more popular. At the same time, to accommodate the increasingly large number of devices, cells have to be designed smaller, thus result in even more frequent handoffs for mobile devices. All of these made the design principle behind mobile IP no longer suitable and the overhead associated with handoff too expensive.

### **3.2.4 Cellular IP**

To provide a low-overhead handoff scheme, Cellular IP is proposed. [28][29] Cellular IP adopts a hierarchical approach to manage mobility. At the higher level, Mobile IP will still be used, and Cellular IP is used to handle lower level mobility.

More specifically, each Cellular IP wireless network has a Gateway Router (GW) and all the Base Stations (BSs) use it. When a MH arrives a wireless network the first time, traditional mobile IP operations are performed to inform its HA. Now all the packets for the MH will be routed to the GW then to the MH. Later, when the MH moves within the wireless network and performs handoffs, it only causes the gateway router to change the base station to forward to. No operation with home agent is necessary. An example is shown in Figure. 3. Here if MH moves from network A to B, Mobile IP will be performed and the HA will be contacted. If MH moves inside network A, no contact with its HA is necessary.

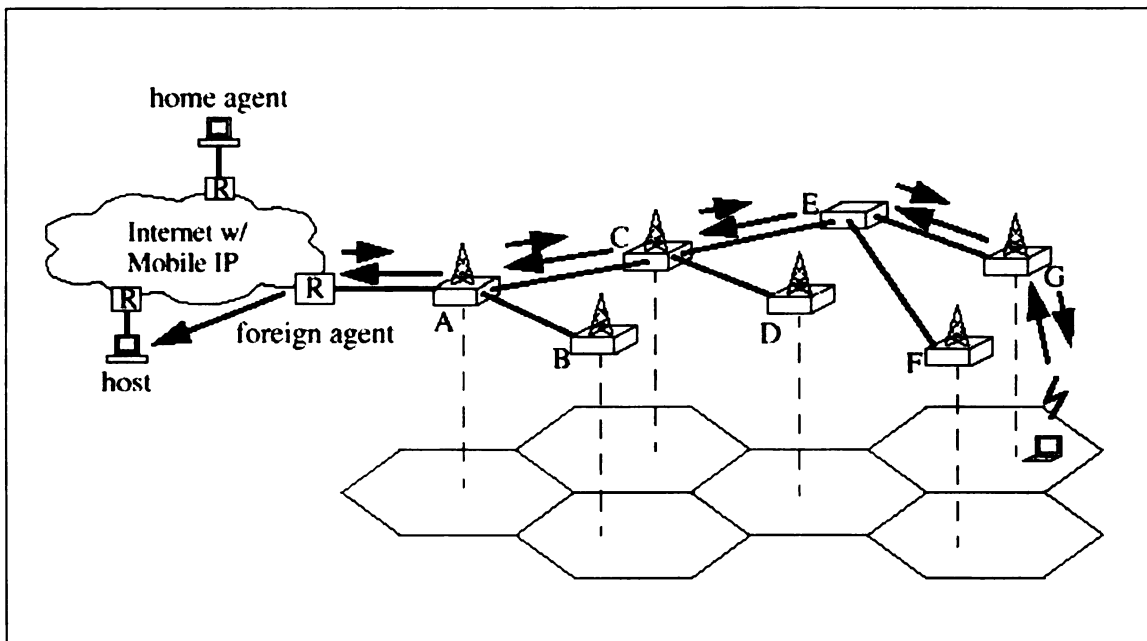


Figure 6. Cellular IP<sup>2</sup>

A clear advantage of Cellular IP is that it greatly reduced the overhead associated with handoff. At the same time, handoffs can be performed at higher speed and easier to be made seamless since the new handoff process no longer involves contacting the home agent.

However, Cellular IP did not provide solution for vertical handoffs directly. In Cellular IP's concept, any handoff between different networks is treated as macro-level mobility and should be handled by Mobile IP. That forces a MH to register with new network each time it connects to the new network. Without special measurements, all the current transmission sessions of the MH will be lost.

### 3.2.5 BARWAN

The Bay Area Research Wireless Access Network (BARWAN) is a wide area network project performed at University of California at Berkeley [6][31]. Using a

<sup>2</sup> Figure adopted from [28]

number of new backbone support mechanisms, BARWAN aims to help mobile devices and applications to adapt quickly to changing network conditions as mobile devices move. An important part of the project is a vertical handoff scheme.

BARWAN's vertical handoff scheme is based on Mobile IP, but with some modifications to make it fast and seamless. The procedure is as follows.

- The HA for the MH creates a forwarding list, which is a multicast address. It includes a small number of BSs and one of them is chosen by the MH to be the "forwarding BS". It is the BS that currently receives and forwards packets for the MH.
- The other BSs are "buffering BSs". They keep a small number of packets for the MH in a circular buffer.
- The MH detects if it is going to encounter a handoff by measuring the strength of the BS beacon signals. If the beacon from current BS is disappearing and the one from a neighboring BS become increasingly strong, the MH will know a handoff is approaching.
- When the MH detects such condition, it sends "start forwarding" packet to the target BS it is moving to. This packet will make the new BS change from buffering state into forwarding state. It also sends a packet to the old BS so it will change from forwarding to buffering state.
- The new BS forwards the buffered packets that the mobile has not yet received.

BARWAN improved handoff performance. However, its enormous overhead and requirement of one multicast address for each MH make it an unrealistic approach in real

life. Basically, BARWAN supports handoff by converting packet delivery mode from 1-1 to 1-n. To support each mobile host, its packets are multicasted to a number of BSs. Considering the huge number of mobile devices today, clearly this is not a realistic requirement. With IPv4, the limited number of multicast addresses simply has no way to satisfy the requirement of so many mobile devices. IPv6 could be able to solve the multicast address problem. But even so, the huge packet-transmitting burden placed on the Internet backbone is still beyond the network's ability.

### **3.2.6 Other Related work**

[21] and [22] introduced the concept of wireless ATM. By setting up virtual channels and virtual paths, wireless ATM allows flows to reserve resources, and thus ensure the required QoS. But there are several problems with that scheme. First, it does not allow dynamic adjustment of reserved resources. Second, in wireless ATM, resource reservations are sender initiated, instead of receiver initiated. It does not work well if the sender is in the wired part and the receiver is in the wireless part of a heterogeneous network. Under such circumstances, only the receiver knows the bottleneck: the bandwidth available in the wireless part.

### **3.3 WRSVP: Wireless Resource ReserVation Protocol**

The first component in SIMA for QoS assurance is a new resource reservation protocol designed for wireless networks: Wireless Resource ReSerVation Protocol (WRSVP).

To guarantee QoS of real-time flows, flows must be allowed to reserve network resources. As we mentioned before, two big problems prevent RSVP from being used in

wireless networks. One is the poor link problem, and the other is the handoff problem. Poor link problem stems from the high bit error rate associated with wireless links. Such poor links cause difficulties in reserving and refreshing resources. Handoff problem occurs when the MH moves between cells. Without special care, the MH can suffer from terminated flows due to un-established resource reservations in the new cell and along the path from the new cell to the transmission parties.

These two problems are the most important motivation that we designed WRSVP. Based on RSVP, it helps to solve these two problems and extend resource reservation to wireless networks. At the same time, we try to provide mobile hosts more ability to make use of dynamically changing bandwidth. In the following, we describe the design of WRSVP. We discuss the case where MHs are receivers of the flows. If MHs are senders, the case is similar or simpler. WRSVP has been published in [1]. For more detailed description, readers are referred to that paper.

### **3.3.1 Solving poor link problem**

Unlike wired links, wireless links have very high bit-error rates. If RSVP is used in wireless networks, mobile host receivers will have trouble in both reserving and refreshing resources. In RSVP, only when a reservation can NOT be made will the receiver be notified. If a Reservation or Refresh message is lost, the MH that sent the message will not be notified, thus it will falsely assume the reservation or refresh has been successful. If several successive Refresh messages are lost, a flow may be torn down automatically by the network, because the reservation will time out. Such problems are ignored for wired networks, where links are much more reliable.

To solve these problems, there are some obvious methods. One of them is to have all the routers send back acknowledgment (Ack) messages to the MH receiver no matter the reservation/refresh is successful or not. If the MH receives the Ask message(s) over a pre-defined period of time, it knows that the Reservation/Refresh message has been received. The drawback of this method is that it will increase the overhead and use much bandwidth each time the MH refreshes the reservation. Another method is to use larger reservation time out values in routers. It will certainly reduce the possibility of unwanted automatic torn-down, but at the same time, resources will be wasted when the MH does stop the reservation, because the MHs are not required to send explicit Release messages. Moreover, even if the MHs do send, the messages can get lost in the wireless environment.

WRSVP solves the poor link problem by separating the reservation process into two parts. One part is between the MH receiver and the BS of the cell where the MH stays, and the other part is between the BS and the wired network. In the wired part, we let the BS do the refresh work for the MH, i.e. the BS periodically sends Refresh messages to the wired network on behalf of the MH receiver. In the wireless part, we add Ack messages to confirm message exchanges. We assume that a MH receiver wants to keep its flows until it explicitly sends message to the BS to release them.

More specifically, it works as follows:

1. The MH receiver sends the first Reservation message to the BS.
2. If the BS can offer the required resources in local network (the cell), then it forwards the requirement to the wired network.



3. After a certain period of time (a predefined time out value), if no error message is received from the network, the BS knows the reservation has been accepted. It then sends an Ack message to the MH, indicating that the reservation is successful. If one or more error messages are received from the wired network, which means that the reservation has been refused by the network, the BS also sends an Ack message to the MH, indicating the failure of the reservation.
4. On the MH's side, it waits for the Ack message from the BS. If after a period of time (another predefined time out value), it receives no Ask message and the flow it is waiting for does not come, then it knows something went wrong with the transmission. It sends the Reservation message again. The above process is repeated until an Ack message is received from the BS (positive or negative) or the flow comes.
5. Once the reservation is made, the BS periodically refreshes it in the wired network on behalf of the MH.
6. When the MH wants to terminate the flow, it sends a Release message to the BS (Release is sent back even if the MH moved to another cell. That case is discussed later). Again, Ack message from the BS to the MH is used to guarantee that the BS can finally receive this Release message. Upon receiving this message, the BS stops refreshing the flow, or sends an explicit Release message to the wired network.

Now MHs no longer need to periodically send Refresh messages. Thus, the reservation overhead for the wireless part is greatly reduced, and more valuable wireless bandwidth is saved. Also, unwanted automatic flow torn-downs are avoided.

### **3.3.2 Solving handoff problems**

In current wireless networks, handoffs often introduce loss of packets, even interruption of flows. For real-time services, it is undesirable, and often unacceptable. With resource reservation, handoff handling becomes even more complicated. When a MH enters another cell, the reservation information needs to be handed over to the new cell's BS. If the information cannot be transferred in time, the MH will suffer loss of flows. One of WRSVP's goals is to help to provide seamless handoffs, which means that not only a flow will not be interrupted, but also the QoS is retained during the handoff period.

Here we assume that all the BSs periodically broadcast beacons, a design feature found in almost all modern wireless networks. By measuring the signal strengths of these beacons, a MH can decide if it is going to encounter a handoff.

WRSVP helps to accomplish the handoff task via *pre-reserving*. When a MH receiver finds that it is going to move out of the current cell, it decides the possible BSs it is heading based on the beacon measurements, and pre-reserves resources along the paths between the sender and those cells. When it arrives at the destination cell, it can start to use the pre-reserved resources right away.

Apparently, to provide smooth handoffs, the handoff protocol in use needs to take care of packet losses. A suggested and natural solution is buffering. To avoid loss of

packets and interruption of flow, the MH can ask the chosen BS(s) to buffer some packets for it. As soon as it enters the new cell, the new BS forwards the buffered packets to it.

Suppose the handoff support buffering, the detailed handoff procedures are described below:

1. The MH that is going to encounter handoff chooses a small group of neighboring BSs, and sends Pre-reservation messages to them (WRSVP daemon). The neighboring BSs send back Ack messages to the MH to confirm the reception of the Pre-reservation messages. The MH sends Pre-reservation messages again if necessary.
2. Upon receiving a Pre-reservation message, a BS WRSVP daemon makes resource reservation using the information contained in the message, thus adds itself to the resource reservation tree of the desired flow. Also, the WRSVP daemon sends a Start-buffering packet to the handoff daemon on the new cell. If the handoff daemon understands this packet, it can contact the old BS to arrange packet forwarding and start buffering packets for the MH.
3. When the MH decides it is the time to handoff (this can be decided based on the comparison of BS beacons' strengths), it sends a handoff message to the BS (handoff daemon) of the new cell it is moving to. The new BS then begins forwarding packets to the MH.
4. After the MH starts receiving packets from the new BS, it sends Release message to the old BS (WRSVP daemon) and all other BSs that are buffering for it.

5. The BSs (WRSVP daemon) that received the Release message delete themselves from the resource reservation tree they previously joined for that MH by sending a Release message. Thus, the network releases those reserved resources. Another possibility is that a Release message from the MH gets lost. In that situation, a BS buffering for the MH will eventually find that the MH is not moving to its cell, thus it releases the reservation.

It should be pointed out that we are taking advantage of RSVP's ability to adapt to changing multicast group membership. Although several BSs make reservation from wired network, these reservations can usually be merged and will not cause much additional burden to the network.

### **3.3.3 Improving wireless bandwidth utilization**

Many mobile real-time applications are able to adapt to network conditions. [55][56]. For example, layered encoding of video frames enables receivers to adjust the bandwidth requirement by adjusting the size and quality of the frames they receive. For such applications, it is desired that the network can notify the applications when available resources change, so that they can do adjustments accordingly. In the above video example, an application can receive bigger-size/higher-quality frames when there is more available bandwidth and smaller-size/lower-quality frames when the network is busy.

To achieve such flexibility, we change the traffic bandwidth requirement in the Reservation message (from a MH to a BS) into two token bucket filters representing the minimum and desired upper-bound bandwidth requirements respectively. There can also be other parameters such as range of allowed delay, jitter, or loss rate.

Initially, the BS assigns to the flow the minimum required bandwidth and reserves that amount of bandwidth in the wired network for the flow. If new wireless bandwidth becomes available in the cell after the reservation is successfully made (which may be due to a newly left MH or a newly terminated flow), the BS sends an Ask-For-Add message to the MH. If the MH accepts this offer, the BS refreshes the flow with the increased bandwidth in the wired network. If the wired network can also offer the increased bandwidth, then the MH begins to receive more bandwidth.

WRSVP also allows base stations to reduce bandwidth from a flow (but still within the range of the flow's acceptable bandwidth) later when there is new bandwidth demand of higher priority.

Wireless links' bandwidth is usually much lower than wired links, and it easily becomes a transmission bottleneck. With the above method, we can avoid idle bandwidth and enhance the quality of real-time flows whenever possible.

### **3.3.4 Deployment Issues of WRSVP**

#### **3.3.4.1 Cooperating with regular RSVP components**

For any new protocol, the expectation of a one-shot deployment everywhere is unrealistic. A very important and highly desirable property is that the new protocol can cooperate with the old network components. That allows a smooth, step-by-step deployment. WRSVP follows this design philosophy.

All modifications WRSVP introduced to regular RSVP are local, and the "interface" to other components of the network is unchanged. Only standard RSVP packets are exchanged with other parts of the network. So for components outside of the

WRSVP network (or network segment), the use of WRSVP is transparent. They only need to understand the standard RSVP. The conversion from RSVP to WRSVP in one network or part of a network will not affect other networks or other parts of the network.

Unlike wired networks where network components are fixed, wireless networks have moving components: MHs. This gives WRSVP further requirements in deployment. WRSVP BS should be prepared to provide service for regular RSVP MHs that are from other networks. Similarly, WRSVP MHs should be prepared to work with regular RSVP BSs when they travel to other networks. The procedures described in last section require both MH and BS understand WRSVP and cooperate with each other. Fortunately, there are options available for both BSs and MHs, so they can work with regular RSVP counterpart.

### **BS part**

When a Reservation packet is received from the MH, the BS can tell which protocol is used: WRSVP or RSVP. They differ in the “protocol name” field in the packet. If it is WRSVP, the BS can rely on the MH to follow the procedures discussed above. Otherwise, the BS should follow the following corresponding options.

- Do not send ack packets to the MH
- Forward MH’s Reserve packet to the wired network
- Refresh on behalf of the MH periodically
- Discard MH’s refresh packet
- Send Release packet to the wired network if no Refresh packet is received from the MH for a very long period. The length of the period is well longer than expected in RSVP, and the consideration of several lost

Refresh packets is included. Since regular RSVP does not require mandatory Release, MHs may rely on the soft state in the network to timeout.

With the above options, regular RSVP MHs will enjoy a far more reliable reservation with the help of WRSVP BSs and not necessarily understand the new protocol.

### **MH part**

For a WRSVP MH, it can tell if a BS supports WRSVP after the first Reserve packet is sent. If a WRSVP ack packet is sent back, a WRSVP BS is in place. Otherwise, the BS supports only the regular RSVP. For a regular BS, MH should follow the normal RSVP procedure. But when the MH is encountering a handoff, it can try to send Pre-reserve packets to the neighboring cells. If fortunately, the targeting cell supports WRSVP, the MH can expect a better service after it gets there.

### **3.3.4.2 Using base stations to help transmitting handoff control packets**

In the process of handoff described in last section, to make the description easier to understand, the MH is used to send Pre-reserve packets before the handoff and Release packets afterwards. In fact, there is another option that uses the BSs to transmit those packets. Before handoff, the MH chooses the neighbor cells and instructs the current BS to send Pre-reserve to those cells. Since the BS knows all the resources currently reserved for that MH, it is easy for the BS to generate such packet on behalf of the MH. After the handoff, the MH furnishes the new BS with the buffering BSs information and instructs the new BS to send Release message to the old BS and all the buffering BSs.

The new option makes the process faster, more reliable, and it saves resources in terms of processing power of MHs and wireless bandwidth. However, its use is limited by the environments, especially when WRSVP is initially deployed and WRSVP BSs coexist with RSVP BSs. Only if the old BS supports WRSVP, can MH rely on it to transmit Pre-reserve packets; only if the new BS supports WRSVP, can the MH rely on it to transmit the Release packets. When such support is unavailable, the MH should send corresponding packets itself.

It is possible that a BS supports only regular RSVP, but a neighboring BS supports WRSVP. So sending Pre-reserve packets give the MH the chance of a smooth handoff.

### **3.3.5 Discussion**

#### **3.3.5.1 Querying MH position from routing daemon**

For a regular RSVP MH, it is possible that the MH sends no Release packet to the old cell after the MH handoffs to another cell. The current solution is to use the WRSVP to monitor the Refresh packet from the MH. If no Refresh is received for very long time, the resource will be released. This solution suffers from two problems. First, it causes wasted resources since resources can only be released after long time. Second, even with long timer, it is still possible that the release of resources is not safe or appropriate. For example, by chance, the uplink (from MH to BS) is down but downlink (from BS to MH) is working, the MH can fail to refresh for very long time but still actively receiving packets.



Table 1 WRSVP Options

Option	When should be used	Effect
Querying MH position from routing/handoff protocol	Support from routing/handoff protocol available	Release resources faster and more accurate.
Using BSs to help transmitting handoff control packets	The associated BS supports WRSVP	Make the process faster, more reliable and saves wireless bandwidth

Table 2 WRSVP Parameters

Parameter Setting		Effect	Resource Consumed
Timer for releasing resources	↑	Safer	↑
	↓	More unwanted resource releasing	↓
Time to start pre-reserving before handoff	↑	Smoother handoff	↑
	↓	More chance of lost packets	↓
Number of neighboring BSs to buffer packets	↑	Smoother Handoff	↑
	↓	More chance of lost packets	↓
Timer for Ack packets	↑	Slower reservation and handoff process	↓
	↓	Faster reservation and handoff process	↑

To overcome these problems, WRSVP daemon can query the Routing/Handoff daemon. If the MH has moved to another cell, the local reserved resources can be released and Release packet can be sent to the wired network. This method is very simple, but it requires the help from routing/handoff daemon.

### 3.3.5.2 Properties of WRSVP

WRSVP retains all the advantages of RSVP. We repeat some of the most important characteristics below. First, the resource reservation process is independent with the choice of routing protocol. Second, flows keep soft state instead of hard state in routers. These properties make the resource reservation process very flexible. No matter

which routing protocol is used, resources can be adjusted automatically to deal with network state changes. If a router fails, the routing protocol will route the Path and Reserve packets using an alternative route, thus new resources along the new route will be reserved and the old resources will be released automatically due to timeout. Last, WRSVP supports multicast, which is a key requirement for real-time services like web broadcasting and web TV.

WRSVP have many parameters and options, which makes it suitable for different environments. In addition, these parameters and options make the performance and overhead of WRSVP flexibly adjustable. Table 1 and Table 2 summarize the effect of these options and parameters.

### **3.3.6 Required support from handoff protocol**

Clearly, WRSVP helps MHs in establishing resource reservation and maintaining it during handoffs. But smooth handoffs could only be achieved if the handoff protocol and WRSVP work together.

The required support from the handoff protocol include understand of Start-buffering packet and MH position query from WRSVP. Another important support is for MHs which do not understand WRSVP. In such cases, the handoff protocol should invoke WRSVP actively. Otherwise, the reservation will not be made until the MH realizes the loss of flows and starts the reservation process. The handoff protocol should notify WRSVP about the MHs' location change (entering and leaving of the cell) promptly, so the reservation can be established or removed.

Only with such support, the most effective use of WRSVP can be ensured.

### **3.4 HOPOVER: HandOff Protocol for OVERlay networks**

In this section, we present the second QoS assurance component in SIMA, HandOff Protocol for OVERlay networks (HOPOVER).

For most applications with QoS requirements, handoff support is critical. Without proper support, handoffs could cause big gaps in packet streams. Sometimes, even connections could be lost. As we mentioned in previous chapter, a lot of research work has been done to improve the performance of horizontal handoffs. [32][19][33][34][35][36][37][38][39][40][41] Many solutions can provide satisfactory performance. However, much less effort has been spent on providing a fast, scalable and smooth vertical handoff schemes. Up to now, most networks still support vertical handoff through Mobil IP [30], which was designed to provide macro-level and slow moving mobility. The scheme suffers from high overhead when high frequency handoffs are needed. More importantly, it incurs big packet flow gaps. There are only few other approaches and none of them get to extensive use due to their limitations and shortcomings.

HOPOVER is our solution to this problem. It is low overhead, scalable and close to seamless. The solution incurs low overhead, even when large number of MHs and BSs are involved and frequent handoffs are performed. It is scalable: the overlay network is easily expandable without affecting the performance of handoffs. Seamless means that when a MH performs a handoff, its current sessions will not be interrupted, no user interference is necessary and applications will not be affected. Perhaps no solution can really guarantee seamless performance. To be modestly safe, we call our solution “close to seamless”.

HOPOVER is designed to address both macro and micro level mobility. Vertical handoffs represent macro level mobility and it involves many complex issues in the process. Horizontal handoffs represent micro level mobility and are often easier to support. The methods and approaches used in HOPOVER are suitable in both cases. HOPOVER dynamically decides the appropriate action based on what kind of handoff it is handling. Parts of the following discussion are applicable to vertical handoff only, and they are easy to tell based on the context.

### **3.4.1 System model and overall approaches**

The protocol is designed for overlay networks. Such network consists of a number of layers of wireless networks. Each layer has many cells. Each cell has a base station (BS) that connects mobile hosts (MHs) in the cell to the wired network.

Each layer of network in HOPOVER has a gateway router (GW) and all the packets to or from that network go through it. The GW has complete knowledge of the network topology.

In the design of HOPOVER, we applied the following approaches.

- **Mobile IP compatible:** Mobile IP is the suggested standard by IETF, and HOPOVER is designed to be compatible with it.
- **Resource reservation:** the handoff process utilizes WRSVP to guarantee that the flows will have required resources once the MH enters the new cell/network.
- **Buffering:** To avoid loss of packets and interruption of flow, the MH can ask the target BS to buffer some packets for it. As soon as it enters the new cell, the new BS forwards the buffered packets to it. To be sure the

sessions will not be interrupted, several nearby BSs can be instructed to buffer packets for the MH simultaneously.

- **Advanced authenticate:** This will reduce the time required for the MH to prove its eligibility to receive service after it arrives at the new network.
- **Forwarding across different layers:** When a MH vertically handoffs to another layer, its packets can be forwarded to the new layer by the BS/GW of the old layer.
- Each MH has state information in the network (BS and GW), and such information is forwarded to the new network when the MH performs handoff. An example of such information is the MH's resource reservation status.
- **Data-driven process.** Two GW routers contact each other and maintain information for each other only when MHs perform handoffs between them.
- **Contact home agent only when necessary.** It is frequently the case that a MH goes back and forth between two layers of networks for a number of times. HOPOVER will not contact the HA each time a vertical handoff is performed. Instead, it waits for a while, and only when it is sure that the MH has entered into a stable condition in the new network, it contacts the HA to have the care-of address (Foreign Agent) updated.

#### **3.4.1.1 Beacons**

In the BS/MH structure, an important feature is that each BS periodically broadcast beacons to let MHs know its status. Each MH often receives multiple beacons

from neighboring BSs or from different network layers. By comparing beacons' strength, MHs can decide if they will encounter handoffs. In the beacons of HOPOVER base stations, network information such as bandwidth is included. Also, MHs can detect traffic condition by observing the loss frequency of those beacons. Such information enables the MHs to make the best selection from available services. Also, in the beacon the BS broadcasts information such as public key information as required by other SIMA components.

#### **3.4.1.2 Authentication**

Current authentication schemes usually require the newly arrived user to exchange with the authentication server some “challenges” and answers, most importantly the username and password. Such schemes are too slow for users who need to access services immediately upon arrival, because it causes lost packets and interrupted transmissions.

We integrate into the overlay handoff scheme an authentication method using Kerberos [10]. At the MH's home network, there is a *Home Authentication Controller (HAC)*. At each network the MH visits, there is a *Foreign Authentication Controller (FAC)*. For networks cooperating with each other, trust relationship between authentication controllers are created beforehand. The MH receives a ticket from HAC verifying that the MH is a legal user in the home network. That ticket is replaced periodically. When the MH visits a foreign network, it presents that ticket to the FAC. If the home network is trusted, the ticket will be accepted by the FAC immediately. This scheme brings a number of benefits.

- The user does not need to be a registered user for all the networks. If only trust relations are created between its home network and foreign networks, the user will be accepted by those networks.
- The user does not need to present the password each time it gets into a new network.
- The authentication is much faster.

### **3.4.1.3 Pre-resource reservation using WRSVP**

HOPOVER provides the support WRSVP requires from handoff protocol, which includes processing of Start-buffering packet and MH position query, and actively informing the WRSVP daemon about the MHs' location changes.

With the help of WRSVP, HOPOVER can guarantee that not only the required resources are available for an incoming MH, but also the resources along the path from each sender to the MH are also reserved. The process will be clearer when we get into the detail of the handoff process.

Remember, many MHs do not understand WRSVP. For them, the help from handoff protocol is necessary to reserve resources in advance.

### **3.4.1.4 Cooperating with Mobile IP and packet forwarding**

Mobile IP is the suggested standard by IETF, and HOPOVER is designed to be compatible with it. After a handoff occurs, the new BS may send update to home agent (HA), to update HA's information of the FA.

In HOPOVER, HA is not contacted each time handoff occurs. A MH can move between layers of network frequently. For example, when a postman delivers mail in a

school campus, he needs to frequently go into and out of buildings. When he gets into a building, he may choose to use the building network, when he gets out of a building, he will need to use the campus network again. To contact his HA each time he switches between building network and campus network is unnecessary.

Since we enabled forwarding across different network in HOPOVER, we can contact HA only when “necessary”. We set up a grace period each time a handoff occurs, only if the MH stays in the new network longer than the grace period, we conclude the MH is now relatively stable and contact the HA. If before the timer expires, the MH moves back to the previous network, no contact with the Mobile IP HA is performed.

### **3.4.2 Handoff process**

Utilizing the approaches presented above, we have the following detailed handoff process. We present the more complex vertical case. Horizontal case is simply part of it.

#### **3.4.2.1 Handoff prepare**

Handoff prepare process is started either by the MH or the wireless network. MH can detect that it is going out of current cell or a better network is available in terms of bandwidth and other factors. The wireless network can detect that the MH is now in a boundary cell and is moving away from the network. The MH location information can come from other sources such as GPS, and how to gather such information is not the job of handoff protocol. The job of handoff protocol is to make best use of such information when it is available. To simplify the description, we assume the handoff process is started by the MH. If it is started by the network, the process is similar and simpler.



1. The MH chooses a small group of neighboring BSs and sends them a *Handoff-Prepare (HP)* packet. An HP message includes the following information: the IP address of the old BS, the IP address and location of the MH, the MH's Kerberos authentication ticket, and current resource reservation information (including the protocol being used). Each BS forwards the packet to its GW.
2. The Kerberos server of the new network verifies the validity of the authentication ticket. If it is invalid, a HP\_NACK (negative Ack of the HP) is sent to the MH, and no further prepare work will be performed.
3. If it is valid, the GW sets routing state along the path from the GW to the chosen BS(s). The detailed process of creating routing state depends on the routing protocol being used.
4. Based on what resource reservation protocol the MH is using (RSVP or WRSVP), the BSs decides if WRSVP Pre-reservation is needed for the MH. If it is, each of the BSs constructs a WRSVP Pre-reserve based on the resource reservation information included in the HP packet. Then this Pre-reserve packet is sent to the WRSVP daemon at each of the new BSs.
5. Then with the help of WRSVP, resources in target cells and along the path(s) from the MH's current session sender(s) to the cell(s) are reserved.
6. Each of the chosen BS allocates a piece of circular buffer for the MH and prepares to buffer packets for the MH.

7. Each of the new BS(s) sends the old BS a HP\_ACK packet. Upon receiving such a packet, the old BS adds the corresponding new BS to the forward list for that MH and begins forwarding packets to the new BS.

By the end of these procedures, routing information along the path from the new GW(s) to the new BS(s) is set up and packets have been buffered for the MH. Also, with the help of WRSVP, necessary resources have been reserved in the new cells and along the path(s) from the MH's session senders to the new BS(s). All of these guarantee a smooth handoff.

### **3.4.2.2 Handoff**

Only MH can decide if it is actually moving to another cell/network. The decision is based on the comparison of BS beacons' strength and other factors such as service and cost in different networks. When the MH decided to handoff to another cell, the following procedures are performed.

1. The MH sends a *Handoff* message to the BS of the cell to where it is actually moving. The new BS then begins forwarding packets to the MH including the buffered packets. An *Handoff* message includes the following information: the IP address of the old BS, the IP address of the new BS, the IP address of the MH, the IP addresses of the BSs the MH used in handoff preparation phase, and the MH's authentication ticket.
2. The new BS sends a *Leave* message to the old BS and all the "handoff preparing" BSs.

3. The old BS records the MH's current network. Later packets will be forwarded there. It stops forwarding packets to other BSs which are on the forwarding list.
4. The old BS notifies its GW about the leaving of the MH.
5. The old BS removes the MH from its "current MH" list. Also, it notifies the WRSVP daemon about the leaving of the MH.
6. Other "handoff preparing" BSs remove the related routing information, allocated buffers and buffered packets for the MH.

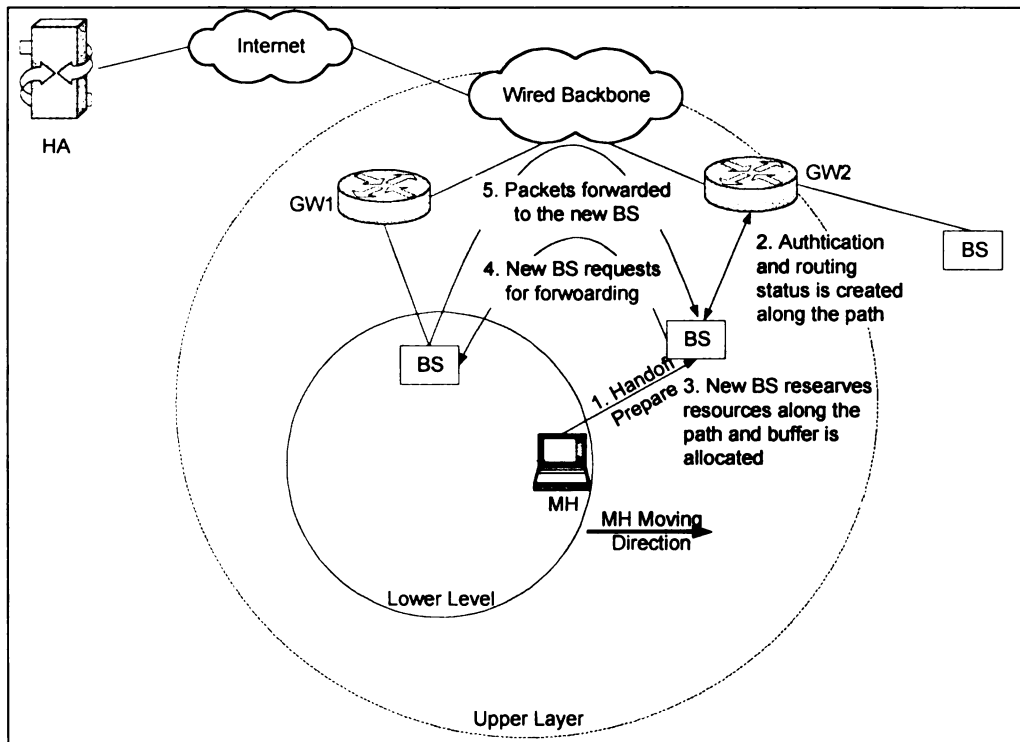


Figure 7. Handoff Process in HOVER (part1)

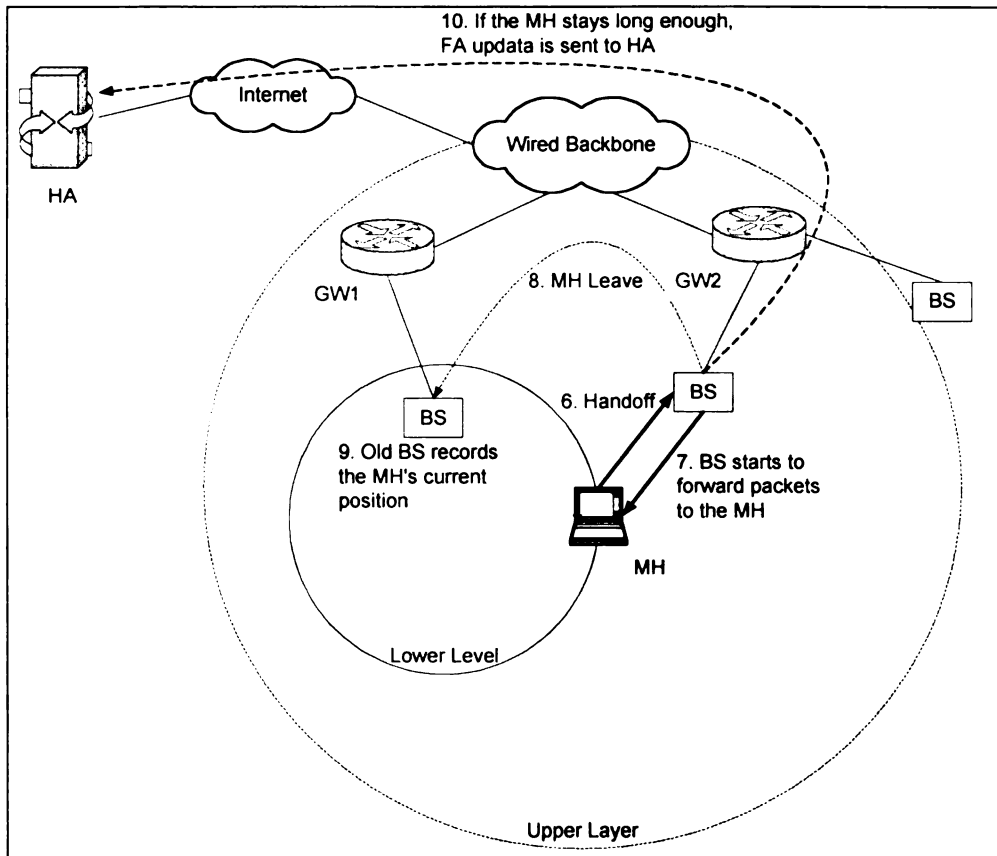


Figure 8. Handoff Process in HOPOVER (part2)

### 3.4.2.3 Updating Mobile IP information

After handoff, the previous BS maintains the forwarding address for the MH. To avoid wasted resources and additional delay, the MH's Mobile IP FA information should be modified to reflect the MH's current address.

1. After a handoff, both the old and new BSs set up a timer for the MH.
2. If after the timer expires, the MH is still in the new network. The new BS sends Mobile IP update packet to the MH's home agent, so the new BS becomes the new FA for the MH. Also, the new BS notifies the old BS to remove the forwarding information for that MH.

3. If before the timer expires, the MH moves back to the previous network. No contact with the Mobile IP HA is performed. Both timers are removed.
4. If before the timer expires, the MH moves to a third network. The second BS sends Mobile IP update packet to the MH's home agent to make itself the new FA of the MH. At the same time, the second BS begins monitoring the MH's stay in the third network.

This whole process is shown in Figure 7 and Figure 8.

### **3.4.3 Utilizing HOPOVER for horizontal handoffs**

The above discussion is for vertical handoffs, the case HOPOVER is designed primarily for. HOPOVER can also be used for horizontal handoffs easily. Compared to vertical handoffs, horizontal ones usually involve closer BSs, easier resource reservation and simplified authentication (MH is moving in the same network). In short, it is a simplified version of handoff problem.

In the horizontal case, since the BSs involved are in the same network, usually there is no need for authentication. The job for GW is now simply setting routes to the new BS. And in many times, it is on the same LAN as the original BS, so no change in routing is needed.

To achieve smooth handoffs, resources still need to be reserved in the neighboring cells before the handoff takes place. But given the fact that the related BSs are in the same network, the resource reservation process can be expected to happen purely locally. Resources on the path outside the local network should have already been reserved by the MH when it is in the original cell.

Due to the short distance, to forward packets from the old BS to the new one is easier too. If they are on the same LAN, it is possible that no forwarding is needed at all: the new BS can receive the packet from the LAN already.

In summary, HOPOVER and WRSVP can be used to improve handoff performance for both horizontal and vertical cases. In real use, HOPOVER adjusts the process by comparing the IP addresses of current and new BSs.

### **3.5 Performance Evaluation**

A set of simulations were run to prove the effectiveness of the mechanisms we are proposing with HOPVOER and WRSVP.

#### **3.5.1 Simulation environment**

The wireless testbed used in the experiment consists of two wireless LANs that connects the GWs, BSs and the mobile host, and a 100M Fast Ethernet network that connects the two GWs/BSs. The wireless communication devices are Aironet 802.11b wireless devices that use Direct Sequence Spread Spectrum modulation technique to provide reliable communication and to protect against eavesdropping. All hosts run the Linux OS, kernel 2.2.14, which supports TCP/IP protocol package.

The machines used to simulate the overlay wireless network include two Pentium II PCs and a Pentium III laptop. A natural arrangement would be to use two PCs as base stations and the laptop as the mobile host. But to work as the mobile host which roams between two different networks, the machine would have to have two wireless network cards plugged in at the same time. Due to the space limitation of laptop's PC card slots, we had to use one of the PCs to work as the mobile host.

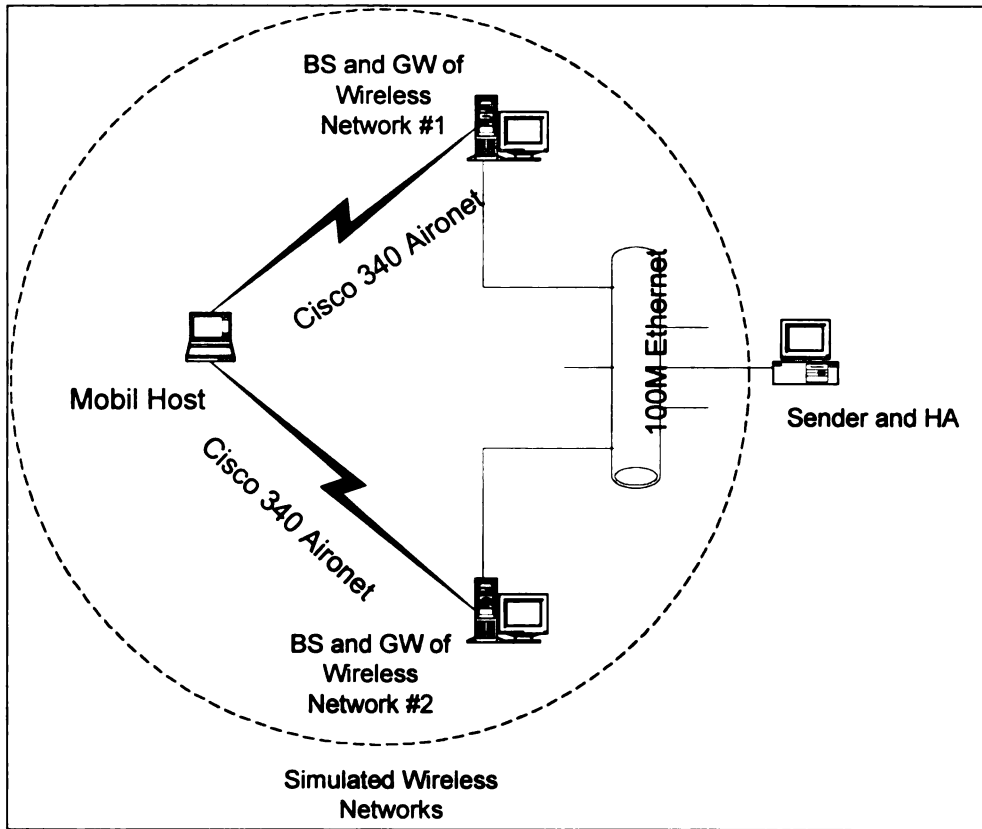


Figure 9. Simulation Topology

The code for simulating gateway routers, home agent, mobile host and sender were written in C++. On the mobile host machine, only one program was run. The program simulated both the mobile host handoff daemon and a user sound player application. Gateway program was run on the other two machines, so they played as both the gateway routers and base stations. On another machine, we run a HA program and a user sender program. The topology was shown in Figure 9.

To represent the transmission delay, packets to be transmitted are placed in a queue and held until the delivery time. The BSs used circular buffer to hold packets for the MHs. Since the buffer is circular, only the newest packets are kept and the old ones dropped.

### 3.5.2 Soft and hard handoffs

We experimented with two kinds of handoffs: soft and hard. A soft handoff is caused by the moving of MH. In this case, gradually, signals of the current BS become weak and those from the new BS become strong. This is the most common case. Hard handoffs happen less frequently, which is caused by the sudden change of links. For example, strong interferences could make the signals from a specific BS temporarily unavailable. In such cases, the MH is forced to switch to a new BS immediately.

To simulate soft handoffs is not an easy task. Generally, wireless link layer condition is physically determined by characteristics such as frequency, distance, noise and so on. It is difficult to quantify all the elements and the associated level of errors. A natural idea would be to move the MH between the two BSs, so it hands off to the closest BS in the movement. But this idea fails in reality, because in such short distance as in one room, the wireless link is always highly reliable. Also, it is difficult to get the movement of MH into an accurately repeating pattern so the results are comparable.

In our experiments, we designed a *controllable error model* to manually introduce packet losses. As just mentioned, the links of the wireless LAN in short distance can be considered 100% reliable, so only the errors generated by the error model would affect the transmission. In our model, the error rate in each BS changes periodically, to represent different distances with the MH. We simulated the case that the MH is moving back and forth between those two BSs, so we set the two BSs to have same error change period, but different phases. That is, when the error rate is at the highest point in one BS, it is at the lowest point in the other BS. In other words, when the MH is closest to one BS, it is farthest away from the other one.



The controllable error model was implemented by modifying the Aironet wireless device driver, which is a loadable Linux kernel module. It can be easily integrated into the kernel or detached from it by using `insmod` and `rmmod` commands respectively. With this error mode, a packet will be discarded by the BS if the error generator determines that an error has occurred.

To simulate hard handoffs is much easier. In our experiments, a hard handoff was introduced by running a shell script that turns on and off the two Aironet cards with adjustable interval. Aironet provides an interface for controlling the wireless cards in `/proc/aironet` directory. A card can be turned off using the command “`echo Radio: off > /proc/aironet/ethX/Config`”. Here `X = 0` or `1` depending on the interface to be shut down. Similarly, “`echo Radio: on > /proc/aironet/ethX/Config`” can be used to turn a card back on.

### **3.5.3 Sound file and sound player**

In the experiments, a WAV format sound file was transmitted. The WAV file is a 3-minute long real human talk and downloaded from the Web. It was sent from the sender to the MH sound player simulator using UDP. A log file was used to record information for all the packets received, such as time and packet number. Also, the sound player simulator recorded the number of packets lost and missed time slots (explained below). The player simulator used a playback buffer and the packets were played back with a small period delay to reduce jitter. Both methods are common approaches to enhance playback quality.

Table 3. HOPOVER Metrics and Parameters

Metric or Parameter	Meaning
Lost packets	Number of packets lost during transmission
Missed slots	Number of time points at which a packet needs to be played but there is no packet in the playback buffer
Packet interval	The time interval between two consequent packets, it decides the sending speed
Playback buffer size	The size of buffer where packets are stored until being played
BS buffer size	The size of buffer where the BS temporarily holds packets for the MH
Playback delay	The delay of playback. It allows the receiver to accumulate more packets before starting playback, thus the chance of missing playback slots becomes smaller.
Remote host distance	The distance of sender and home agent.

#### 3.5.4 Metrics and parameters

Before we describe the results, we summarize the metrics and parameters used in the experiments in Table 3. Packet interval indicates how fast packets are sent from the sender. It is also the frequency at which packets are played back. The playback buffer size and BS buffer size are measured in seconds instead of bytes. The actual size of the buffer (in packets or bytes) depends on how fast the packets are played back. If the playback buffer size is one second and packets are played back at 50 packets/sec, then the buffer can hold 50 packets. Missed slots indicated the number of time points at which a packet needs to be played but there is no packet in the playback buffer.

To simplify the simulation, we used “Remote host distance” to represent the distance (in one-way time length) of both the sender and home agent.

### **3.5.5 Simulation results and analysis**

The following default parameters are used in the simulations unless specified otherwise. The handoff threshold is the loss of 10 consecutive beacons. The packets are sent at 50/second frequency, and the play back delay is 0.1 second. When HOPOVER is not used, handoff depends of mobile IP only.

In observing the experiments results, we found that packet losses and missed slots rarely happens in normal transmission. Most of them are caused by handoffs. So instead of the total packet losses and missed slots, we draw the average numbers caused by each handoff.

With appropriate setting, e.g. sufficient buffer, HOPOVER can cut the numbers of lost packets and missed slot by 80% or even more. The result is especially apparent when the sender and home agent is far way, the case HOPOVER is designed for.

#### **3.5.5.1 Soft handoff simulation results**

The results of soft handoff are shown in Figure 10. and Figure 11.

Figure 10 shows effect of playback buffer size. The playback buffer size changes from 0.1 (very small) to 1 second (very big). BS buffer is half the size of playback buffer, which is a reasonable number for resource allocation. The playback delay is 0.1 seconds and the remote host distance is 150 milliseconds. As expected, the number of lost packets and missed slots decrease as the size of playback buffer gets bigger. When HOPOVER is not used, the decrease is very steady. When HOPOVER is used, there is an elbow point where buffer size is 0.25 seconds. The effect of HOPOVER is more apparent when the buffer size is larger. As can be seen, HOPOVER cuts the number of lost packets/missed slots by about 70%. Even with very small buffer, the effect is about 60%. Without

HOPOVER, the gap caused by each handoff is 0.8 to 0.9 seconds, which is apparent to human ears. With HOPOVER, the gap is reduced to 0.2 to 0.4 seconds.

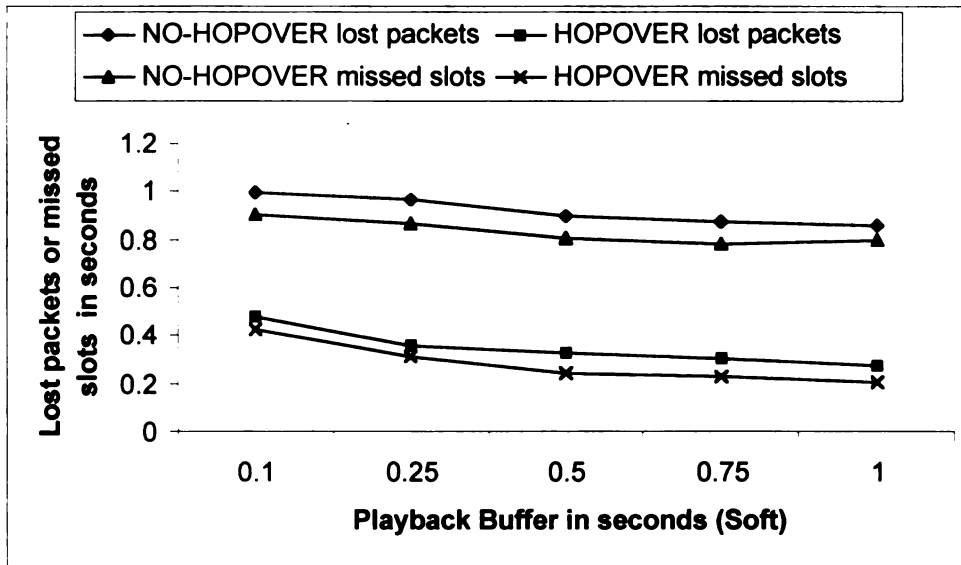


Figure 10. Effect of Playback Buffer in soft handoffs

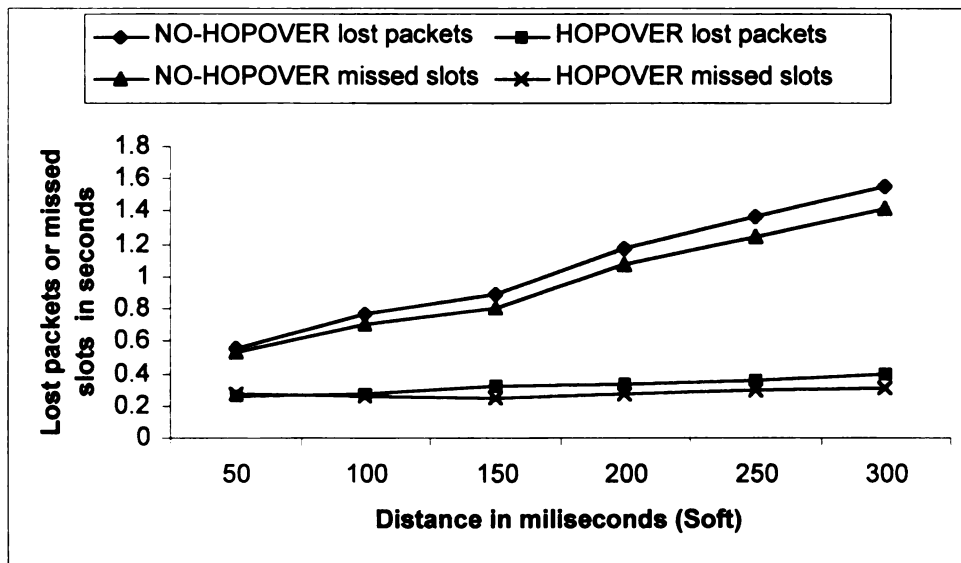


Figure 11. Effect of Remote host distance in soft handoffs

Figure 11 shows effect of remote host distance. The remote host distance changes from 50 milliseconds (very close) to 300 milliseconds (very far away). The playback delay is 0.1 seconds. As expected, the number of lost packets and missed slots increase as the distance of remote hosts gets bigger. But with HOPOVER the increase is much

slower. As can be seen, without HOPOVER, the gap caused by each handoff increased from 0.53 to 1.41 seconds, nearly a triple. With HOPOVER, the change is only from 0.26 to 0.31 seconds. With very close HA and sender, HOPOVER cut the gap by about 50%; with very far away condition, the cut is nearly 80%. With HOPOVER, since the packets sent to old BS are not lost, QoS becomes less sensitive to the distance of sender and HA.

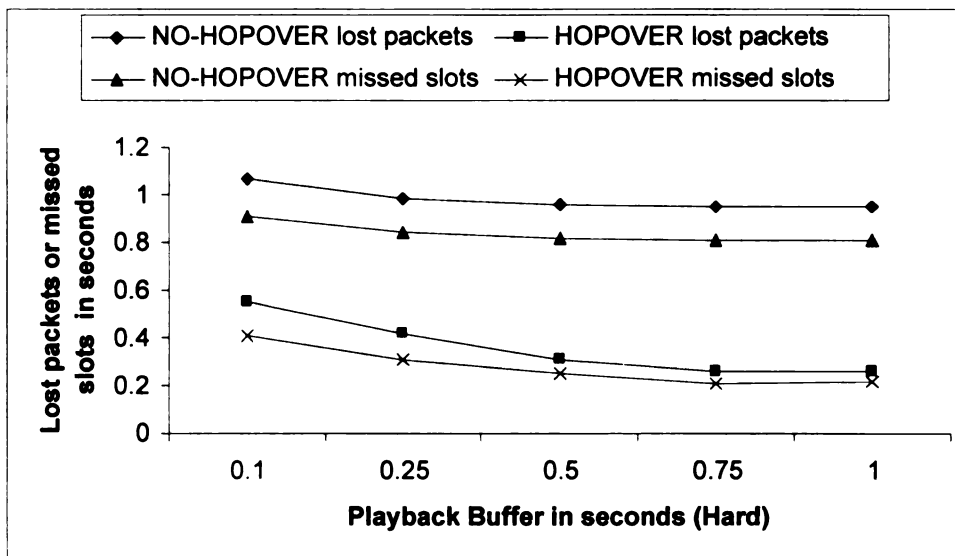


Figure 12. Effect of Playback Buffer in hard handoffs

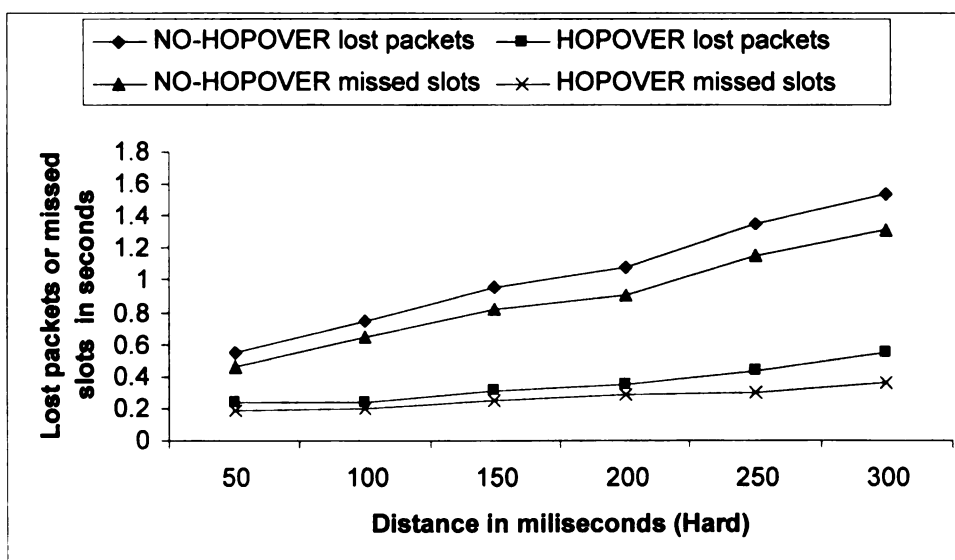


Figure 13. Effect of Remote host distance in hard handoffs

### **3.5.5.2 Hard handoff simulation results**

Figure 12 and Figure 13 show the results of the hard handoff experiments. The same settings and parameters are used as in soft handoff experiments.

Nearly identical change and trends are observed as in soft handoff experiments, except this time the number of lost packets are larger. This is reasonable: since the handoffs are caused by sudden termination of interfaces, once the handoff happens, no packets will be available from the old link. In the case of soft handoff, the old link could still forward some packets even at high packet lose rate.

### **3.5.6 Overhead analysis**

The overhead of HOPOVER mainly includes the following parts:

1. Handoff packets exchange
2. Processing power and spaced used by buffering and forwarding operations at BSs
3. Resources occupied by pre-reservation in targeting cell and reservation with neighboring networks.

The first part of HOPOVER overhead includes the necessary packet exchange between MHs, BSs and GWs. In the whole handoff process, the following packets will be exchanged (assuming vertical case):

1. MH sends “Handoff-Prepares” to possible BSs, then they are forwarded to corresponding GWs
2. “RSVP reserve” sent to the network by each of the possible BSs
3. “HP\_ACK” packets sent to old GW
4. MH sends an “Handoff” message to the new BS

5. New BS sends a “Leave” message to the old BS and all other preparing BSs.
6. Possible Mobile IP update packet sent to the HA

Packet 4 and 6 are also used by regular mobile IP. Packet 5 is a single packet. Numbers of Packet 1 and 3 are limited by the possible number of neighboring cells. The number is unlikely to be more than 3 for each packet. Numbers of Packet 2 are limited by RSVP. RSVP merges reservations along the tree routed at the sender. Once a reservation is satisfied at one point of the network, the reservation is no longer forwarded upwards because the up-links must already have the necessary resource reserved. The conclusion of the above analysis is that the additional packet introduced to the network is very limited and will not consume much bandwidth.

In analyzing the second part of HOPOVER overhead, we point out that as part of the wired network backbone, BSs usually hold much more processing power and space than those of MHs. In addition, more resources can be added as necessary since there is no space limitation. As the processors become more powerful and memory technology advances rapidly, we don not anticipate this part of HOPOVER overhead to be an issue.

The last part of HOPOVER overhead also poses no big concern due to two reasons. First, it is limited by the possible number of neighboring cells. Usually the number is very small. Second, the overhead is mitigated greatly by the use of WRSVP/RSVP structure again. Although several BSs make reservation from wired network, the common part of the reservations is merged. So, the overhead is usually limited to the BSs, where we argue resources can be added with little effort.

## 3.6 Discussion

### 3.6.1 Handoff among more than two networks

As we mentioned earlier, a grace period timer is used to decide if the new GW should contact the MH's home agent. The underlying assumption is that if the MH leaves the new network before the timer expires, it is moving back the original network, so there is no need for contacting the HA. An interesting case is that the MH moves to a third network. That is, a MH moves from network A to B, and before the timer expires, moves to C again. Now, which network should work as the MH's FA?

With HOPOVER, the new BS of B should make itself the FA of the MH. This way, we limit the packets to be forwarded at most once (from one network to another network) and we give the MH a chance of moving back to B in short time.

The other possibilities would be to use either BS\_A or BS\_C as the FA. Using BS\_A means we do not modify the FA information until the MH is stable. The problem with it is that we could have a long forward link if the MH keeps jumping from network to network. Using BS\_C gets packets sent directly to the newest network. The problem with this approach is that it can result in very frequent Mobile IP updates.

We made the decision based on the following observation. For inactive MHs, those handoff infrequently from network to network, they stay in a network for long time. So their current network will finally become their FA. For active MHs, those handoff frequently from network to network, they stay in a network for short time. But usually that happens when the MH "bounces" between two layers of networks. The MH goes to the lower network when it is available and goes up when the lower network becomes unavailable. Using the example of postman again, most of the time, he utilizes the



campus network. But when he enters a building, he utilizes the building network for faster service and better bandwidth. Each time he switches to a building network, he just stays for a short time. Making the building network his FA is apparently not a good choice. Instead, a better solution is to maintain the campus network his FA and forward his packets to the building network when necessary. This way, the packets need to be forwarded at most once and his state is much more stable. In addition, we save both bandwidth and processing work associated with the Mobile IP control packets.

### **3.6.2 Using separate HOPOVER and WRSVP component**

HOPOVER and WRSVP protocols consist of components on MH, BS and GW. The design of HOPOVER and WRSVP made each entity independent and compatible with open standards such as Mobile IP and RSVP, so HOPOVER- or WRSVP- installed components can work in mixed environments.

For MHs, they can decide if a BS supports HOPOVER or WRSVP from the beacons. When they found HOPOVER or WRSVP aware BSs, they will enjoy the enhanced service.

For BSs and GWs, the upgrade to HOPOVER and WRSVP benefits all MHs, not just those HOPOVER or WRSVP aware ones. For old MHs, they may just find that resource reservations become more stable and handoffs smoother. The reason is that WRSVP BS daemons help to establish and refresh resource reservations; and HOPOVER BSs help in resource pre-reservation, buffering and forwarding packets.

The independent feature of those entities allows the deployment of WRSVP and HOPOVER to be performed gradually, which is a critical requirement for any new protocol to be adopted.

### **3.7 Conclusion**

In this chapter, we presented the first and second components of SIMA in ensuring QoS: WRSVP and HOPOVER.

In wired networks, no matter with Intserv or Diffserv, QoS is essentially provided by allocating abundant resources (bandwidth, buffer, processing power etc.) to higher priority traffic. If only such resources are available, usually packets can be delivered as required. In wireless networks, having enough resources is still critical for QoS. But in addition to that, there is an equally important factor, which is the performance of handoffs. Without enough resources, no QoS can be talked about. Without a good handoff scheme, no QoS can be sustained after a handoff takes place.

Correspondingly, in SIMA, QoS assurance is provided primarily by WRSVP and HOPOVER, addressing resource reservation and handoff respectively.

To guarantee QoS of real-time flows, flows must be allowed to reserve network resources. But two big problems prevent the most widely used resource reservation protocol, RSVP, from being used in wireless networks. One is the poor link problem, and the other is the handoff problem. Poor link problem stems from the high bit error rate associated with wireless links. Such poor links cause difficulties in reserving and refreshing resources. Handoff problem occurs when the MH moves between cells. Without special care, the MH can suffer from terminated flows due to un-established resource reservations in the new cell and along the path from the new cell to the transmission parties.

WRSVP solves the poor link problem by separating resource reservation into wired and wireless parts and by adding Ack messages into wireless message exchanges.

Using pre-reserving in neighbor cells, WRSVP protects flow reservations from handoffs, both in the cell and along the path. Resource reservation becomes a natural part of handoff.

In wireless networks, handoffs often cause severe QoS damage. Users could experience an apparent interruption, or sometime even the connection could get lost. HOPOVER enables smooth handoffs intra- and inter-network, and it is compatible with mobile IP. HOPOVER helps mobile devices using the following measurements: facilitating WRSVP to pre-reserve resources in the new cell and along the path from the new cell to the flow transmission parties; authenticating in advance of the actual handoff; buffering in the new network for the MH and forwarding packets from the old network to the new network. With these methods, HOPOVER significantly enhances handoff performance.

Our simulation results proved the effectiveness of these measurements. HOPOVER significantly reduces the gap caused by each handoff. The effectiveness of HOPOVER is most apparent when the sender and home agent are far away, the case QoS improvement is needed the most.

## **4 Server Selection In Wireless Networks**

SIMA has three QoS assurance components, WRSVP, HOPOVER and a server selection scheme. The first two components have been described in Chapter 3. In this chapter, we present the third one.

### **4.1 Background**

Network services often suffer degraded availability and response time as they become popular. This problem is painfully evident in the Internet, where individual servers and network links are often swamped with tremendous, global service demands. To solve the problem, an effective way is to provide replicated servers to share the working load. With multiple replicated servers running simultaneously, the number of requests serviced by each server is reduced and a reasonable response time could be achieved. Also, resources such as bandwidth are utilized more efficiently, since requests can be handled by nearby server(s). From the point of view of clients, properly placed replicated servers increase network proximity and thus enhance service availability and reduce access latency they perceive.

Traditional solutions can be roughly divided into two categories. Client side approaches often require clients probe for best server each time service is wanted. Server side approaches select next server based on workload and other server side consideration.

Mobile hosts usually possess lower computing power and slower link than wired nodes. In addition, mobile hosts encounter handoffs. As the result, they are more sensible to quality of the service, and an appropriately selected server is of even more importance than in wired networks.

The attributes of mobile hosts also poses more stringent requirement on server selection mechanisms than in wired networks. Mobile hosts possess limited resources in computing and bandwidth, so they can hardly participate in server probing processes. The desirable solution should involve the mobile hosts as little as possible. Mobile hosts could encounter handoffs frequently. If the MH stays with the same server all the time, their changing location could make their chosen server no longer a good choice. This is particularly true for vertical handoffs. Different networks could use different ISPs, thus the network condition and topology could change significantly after a handoff. Due to the possibly frequent handoffs, a very fast and low overhead solution is needed.

We proposed a new server selection mechanism named *smart server selection* (S3). The central idea is to utilize the client side DNS and router jointly to decide the best server for the clients. Our simulation shows that this method significantly saves network resources and improves service quality perceived by end users.

S3 has been published in [3], and was a joint work by Wenting Tang and the author of this dissertation. The idea of querying a router from the client DNS was the result of several discussions. The detailed design of S3 mechanism, as well as the design and implementation of the simulations were all joint work. The integration of S3 into SIMA structure is the author's own work.

In the remaining part of this chapter, we first review related work in Section 4.2, then we present the design of S3 in 4.3. Section 4.4 presents the way S3 is integrated into SIMA. In 4.5, we discuss the properties of S3 and some related issues. In Section 4.6, we discuss the design and results of our simulation, and finally in 4.7, conclusion is given.

## **4.2 Related Work**

Many methods attacking the server selection problem have been proposed. They can be divided into two categories depending on where the server selection is performed, the server side or the client side.

### **4.2.1 Server side approaches**

In server side approaches, server side network services gather information and make a decision. An example of server side approach is Cisco Distributed Director [44]. The advantage of server approaches is that they do not need changes to be made at the client network. All work is done by the service provider. The disadvantage is that the service provider needs to purchase and maintain special devices. Also, such an approach generally suffers a linear increment in cost as the number of replicated servers increase. In a long run, such a problem (finding nearby replicated servers) should be addressed at the client side, which will make support global replicated services much easier for the content providers. Server side approaches often focus on load balancing and resource utilization, but they hardly address the different needs of each individual client.

### **4.2.2 Client side approaches**

In client side approaches, client applications/network services collect information from networks and replicated servers, and make a server selection. Based on the manner and time server information is collected, client side approaches can be further divided into static methods and dynamic ones.

In static methods, e.g. [81][82], server information is collected beforehand. Due to the feature of static methods, frequently, the metric being used is the distance in hops to each server, because it is stable over longer period. Static server selection scheme is used

in the distribution of network news using NNTP (Network News Transfer Protocol). Round robin selection using DNS can be classified as static. In such a configuration, the DNS server maintains a list of IP addresses and answers IP requests in a round robin fashion from this list.

Dynamic methods, e.g. [83], find good service providers without a prior knowledge of server location or network topology. It is assumed that the client has been provided with a list of IP addresses of servers. When a client wants to connect to the service, it measures the performance of the candidate servers and selects the best one. Since the measurement takes place real-time, the metric can be response time and latency, which represent the current network condition.

Dynamic server selection often provides significantly improved response time for users when compared with static server assignment policies (which are often purely based on network distance in hops). Many times, dynamic server selection also helps overall data traffic distribution in the network, since clients are given the chance to avoid congested network segments. Certainly, dynamic server selection pays extra cost in runtime measurement. It introduces overhead to the clients, and costs additional traffic to the network.

Smart Clients [54], probabilistic model [48] and automatic selection [47] provide application-level implementation to collect the metric and make server selection. The SONAR service [49] is a special server that prioritizes a list of IP addresses for a client according to the information it collected. The first three mechanisms force the applications to do the redundant work, thus waste resources. The last one requires the

applications to be configured with the special server information. All of these mechanisms are not transparent to applications and it limited their usability.

### **4.3 S3 - Smart Server Selection**

Our method, Smart server selection utilizes client side DNS server [72][73] and routing metrics to help server selection. DNS has been used to help server selection before, but in most cases, a round-robin method is used. In such a setting, the DNS server keeps an IP address pool of the target service, each IP representing a server. When client queries come in, these IP addresses are returned following the round-robin sequence.

The unique point of DNS is to utilize routing information combined with DNS server. Updated routing information is kept in addition to the IP addresses, thus the clients can query based on their preferences. The combination of DNS server and routing information makes S3 exceptionally efficient and accurate.

#### **4.3.1 Utilizing DNS server and routing metrics in server selection**

In S3, all replicated servers are aliased to the same DNS name. When a client accesses the service, the client DNS select from the IP address pool corresponding to the replicated servers and return the server address best matching the user's preference, such as shortest path or shortest latency.

In order to select a server based on shortest path or latency, the DNS server must know routing information leading to different servers. However, such information is only available from the routers. Routers know the metrics of the routes corresponding to each IP address but they have no idea which IP addresses are providing same services. This is only known by DNS.



To solve the server selection problem, a mechanism is needed to enable DNS servers and routers to communicate with each other. We have proposed router extensions to support a route metric query for IP addresses. Extensions to current DNS are proposed to allow it collect and cache routing metrics and select the best server. We name the mechanism *Smart Server Selection (S3)*.

S3 removes the need of individual server probes performed by each client. This not just avoids redundant work. More importantly, it avoids remote information exchanges. S3 also takes advantage of DNS's role in the activity of accessing the replicated service. Due to the popularity of the replicated services, it is very likely that multiple hosts access the same service within a short period of time. When that happens, the routing metric information to the service cached in the DNS can be shared by multiple DNS queries, and therefore reduce the number of routing metric queries and the DNS server response time.

Our simulation results show that S3 provides substantial performance improvement over the DNS round robin approach, which is the default method of server selection, both in terms of the number of hops used and total latency experienced. The overhead analysis shows that the overhead introduced by our routing metric collection scheme is negligible.

S3 is a general solution for server selection. It can be used for both wired and wireless clients. But comparing to other methods, S3 is particularly suitable for wireless networks. S3 performs server selection without involvement of the client, which saves valuable wireless resources. More importantly, the whole process of S3 server selection involves no remote information exchange, which makes the process extremely fast. This

is a critical requirement for a smooth handoff. With S3 support, SIMA allows mobile clients to be always connected to the best server, even while they move. All the process of server selection and switching to the new server happen in the background and is transparent to the clients.

S3 is a client side approach but it is different with existing approaches:

- The Client DNS server is the central point to collect and prioritize the IP addresses. Such an extension has three advantages: 1) It avoids redundant work from individual applications. 2) The extension is transparent to the client application. 3) More information may be collected by the DNS, such as geographical location of servers [46].
- Efficiency of metric collection process is improved by directly querying the routing tables. This reduces the overhead and improves the DNS response time.

S3 operates as follows:

1. The server side DNS maps a DNS name to all replicated servers' IP addresses.
2. When a client DNS asks for the name-to-IP-address mapping for the replicated services (DNS name), the server DNS sends all available IP addresses to the client DNS.
3. When a host sends a domain name query to the client DNS, the client DNS examines whether multiple addresses for this service are available. If multiple addresses are available and the metric is not cached, the client DNS sends a query to collect necessary routing metrics.

4. The DNS selects a server according to the route metric it collected from the routers and other information available to DNS servers (such as the geographical distance between the client and the replicated servers) and returns the selected IP address to the host.

### **4.3.2 Collecting routing metrics**

The DNS needs to query a router that has full knowledge of Internet routes, which is usually a BGP (Border Gateway Protocol) router [45]. We propose two different approaches for this job. The approach to use depends on the conditions of the local network.

#### **4.3.2.1 Query the gateway router**

In the first approach, we assume that the client DNS knows at least one gateway router when it is configured. Armed with such information, the DNS sends queries directly to this gateway router. The gateway router searches its routing table and supplies necessary information to the DNS. The DNS needs to find the gateway router through other means, and is probably configured manually. A daemon runs on both the DNS and the gateway router, which will allow them to communicate with each other. The daemon on the DNS enables the sending of routing information queries and the one on the gateway router accepts such queries and sends responses back to the DNS. The main advantage of this approach is the ease of deployment. Only the gateway router needs to be upgraded to support the routing metric query. All other routers need not be upgraded.

A technical issue arises if one replicated server is in the same Autonomous System (AS) and may be reached without passing through the gateway router. In this

case, the gateway router may not know the routing metric from the DNS to the replicated server. This is the case where OSPF (Open Shortest Path First) [50] is used as the Interior Gateway Protocol (IGP) and the gateway router is in a different area than the DNS. The gateway router provides routing metrics based on the routes from itself to the replicated servers, which may be different from what is perceived by the DNS and the actual clients. Fortunately, since the difference is only for routes inside the same AS, the inaccuracy will not likely cause noticeable performance degradation.

#### **4.3.2.2 Query the directly attached router**

In the second approach, the DNS may send a query to the router to which it is directly attached (default router) to obtain the routing metric information. This approach relieves the requirement of DNS's knowledge of the gateway router.

We extended ICMP (Internet Control Message Protocol) [52] to support such a mechanism instead of developing a brand new protocol because ICMP is implemented on every router and such a mechanism is a natural extension of ICMP.

The ICMP packet header has a "Packet Type" field. We extended ICMP to make use of two currently obsolete packet types: type 15 ("information request") and 16 ("information reply"). We specify that routing queries from the DNS use ICMP packet type 15 and responses from the router use type 16. The scenario of a routing information collection is described as follows:

1. In a query packet, the DNS sets the packet type to 15 and specifies in the packet body the IP addresses and the routing metric for which it is looking (hops, latency or others). Then, it sends the packet to its default router.

2. Upon receiving such a query, a router looks up its own routing table. If it can provide all the required information, it replies to the DNS. Otherwise, it fills the information available in the ICMP packet and forwards the packet along the default path to its default router, if there is one.
3. The upper level router tries to find the missing information. It will not overwrite or repeat the information that is supplied by lower level routers.
4. This process continues until a router supplies the last piece of information. This router creates and sends a reply packet (ICMP type 16) to the DNS. The query packet eventually reaches an EGP router if some routing information about destinations outside the AS is requested. If there is no default path on a router and the information is still not complete, then it means some IP addresses are not reachable from that network. This router may safely send a reply to the DNS.

#### **4.3.2.3 Comparison of the two approaches**

The second approach has obvious and important advantages over the first approach. First, it relieves the requirement of DNS's knowledge of the "working" router address. It is completely transparent to the network and DNS administrators, and it works independently of the routing protocol used. Second, it results in highly accurate routing metrics because ICMP packets go through the path that the data packets to those IP addresses are actually forwarded. Third, because it queries IGP routers, the problem of the first approach is addressed. Last, because ICMP extension provides a way to collect

route metric for multiple IP addresses, this service can be exploited by other servers or hosts too. Compared to the first approach, the second one involves all the routers along the default path up to a router where all routing metric can be supplied. All these routers need to be upgraded to support the new ICMP functions. This may result in longer deployment, and more investment. These two approaches are complementary to each other. The first approach may be deployed quickly, while the second one should be used when enough new ICMP function enabled routers are available.

### **4.3.3 DNS extensions**

DNS servers are extended to handle routing metric queries. The DNS cache entries will also be extended to store routing metrics. Under the current approach, the selection criteria based on the available metrics is configured in a central place: the DNS server. Technically it is easy to let the host or application specify the selection criteria (by applying different weights to each criteria), however, this will lead to the extension of the DNS protocol. At this point, it is not clear if such an extension is necessary.

## **4.4 Supporting Server Selection in SIMA**

S3 is integrated into the SIMA structure. For MH in static status, S3 works with no difference as in wired networks. However, for mobile hosts performing handoffs, special measurements must be applied. SIMA support for S3 mainly focuses on handoffs, and the support consists of two parts: on the wired backbone and on mobile devices.

#### **4.4.1 Backbone part**

The server selection job is mainly carried out by the wired backbone components, which include the BS, the DNS and the router. For DNS and router, they run regular S3 protocol. For the BS, it performs the following tasks.

- Keep track of the MH's status of server selection, so the BS knows all the domain names the MH queried and the corresponding IP addresses in use. This requires the BS to monitor all the MH's DNS queries.
- Each time a handoff happens, the MH's server selection status is transferred to the new BS.
- The BS queries the new DNS for the domain names, and records the new results. Here, regular S3 is used to perform the server selection and the client's server preference such as number of hops or cost is used.
- If a better server is found, the BS sends an "Application server change" packet to the MH with the new server information.

#### **4.4.2 Mobile host part**

The requirements of mobile hosts are the appropriate processing of S3 packets and SIMA "Application server change" packets. First of all, a MH needs to understand S3 in order to query the server address according to its preference such as number of hops or cost. In addition, the mobile host should be able to handle the "Application server change" packets at handoffs. The MH keeps an IP addresses buffer for DNS queries. When an "Application server change" packet is received, the buffer is updated accordingly. So next time an application queries the IP address of its server, the address in the buffer will be used directly if it is still in its valid period.

## **4.5 Discussion**

The design of S3 is quite unique, which makes S3 a fast, efficient and low cost solution.

### **4.5.1 S3 is both static and dynamic**

S3 is not purely static or dynamic, instead it is somewhere in the between. The clients are not required to perform server probe each time they make a connection. From this point, S3 is similar to static methods. Server information is always up to date, and it includes not only stable information such as distance but also real-time information such as latency. From this point, S3 is similar to dynamic methods.

In short, S3 enables server selection based on fresh, dynamic information without incurring the overhead of client probing.

### **4.5.2 S3 is suitable for mobile clients**

Traditional approaches choose the best server either at the server side or the client side. They come across difficulties in wireless networks due to two fundamental properties of mobile applications.

The first property is that the clients always move around and the topology and network condition always change. For traditional client side approaches, the information collected by MH's becomes invalid very quickly. Frequently, the client chooses a best server in one cell, and then moves to another cell where that server becomes a bad selection. Server side approaches are affected also, because they assign server based on current traffic and network condition. With changing topology, such information is of little help to predict the condition in the next time frame.



The other property is that clients are of limited computing and bandwidth resources. Many existing methods probe the current network condition in making server selection decisions. Wireless clients' limited resources make such probe hard to perform.

With S3, these two problems are avoided. Now each time a client performs handoff, a new server can be chosen automatically. Since it is always provided by the current network (the router, which is in the best place to observe the network condition), the information is always correct and up-to-date. There is no need of the client's participation, the mobile clients' limited resources no longer pose a difficulty.

#### **4.5.3 Server load consideration**

Server load is an issue closely related to server selection. Intuitively, server load should be considered when server selection is performed. In some approaches the user will always be directed to the lowest latency server so that user perceived latency can be minimized. One apparent drawback of such an approach is that in order to obtain accurate estimate of server load condition, expensive measurement has to be conducted frequently (in the order of minutes) due to the fact that server load changes dramatically in short period of time. The principles of S3 are:

- (a) The server selection criteria should select the server based on network metrics.
- (b) Temporary server overload should be addressed by efficient server side load balancing approaches such as [42][43].
- (c) Persistent overload should be addressed by capacity planning and monitoring on the server system, given today's hardware is inexpensive.

We argue the principle of S3 makes better use of network resources. Load balancing may redirect a user to a server half around the world simply because that server

has lighter usage. By doing this, valuable network resources are wasted. We believe overloaded servers should be avoided by adding new servers, moving the locations of the servers, or choosing server locations that better match users access pattern.

#### **4.5.4 S3 is scalable**

We discussed in previous section that as the number of servers increases, other server selection methods suffer either a linear increment in metric collecting or a linear increment in resource consumption. With S3, the latency of metric collection is independent of the number of replicated servers and may always be done by sending a query to the router. Therefore, it is fast and resource conservative.

#### **4.5.5 S3 enhances fault tolerance**

Traditional DNS does not know the status of each replicated server. It may select the IP address in random or round robin. The server it chooses may be unreachable because the DNS has no knowledge of the routing information. With S3, a server that cannot be reached from a router's point of view will not be selected by the DNS. This gives flexibility to service providers. Servers which are down or being maintained are avoided automatically.

### **4.6 Performance Evaluation**

In this section, we present the results of several simulations conducted to evaluate S3's performance. The simulations were designed to prove the effectiveness of S3 in general. The simulated environment is a fixed network, but the validity of the results holds in wireless networks as well.

#### 4.6.1 Simulation setup

We measured in our simulations the web accessing performance from a network to a site that provides the service globally. The client network modeled in the simulations is the public computer laboratory connected by a 100 Mbps Fast Ethernet in the Department of Computer Science and Engineering at Michigan State University. Yahoo ([www.yahoo.com](http://www.yahoo.com)) is chosen to be the site that provides the global replicated services. The web site of Yahoo is replicated at over 20 places, and each place has at least two different IP addresses. To be more realistic, only servers providing the English language are selected. The 5 places chosen were: [yahoo.com](http://yahoo.com), [ca.yahoo.com](http://ca.yahoo.com) (North America), [sg.yahoo.com](http://sg.yahoo.com) (Asia), [uk.yahoo.com](http://uk.yahoo.com) (Europe), and [yahoo.com.au](http://yahoo.com.au) (Australia). Each group is on a different continent. The hops and latency from each place was measured by traceroute [53], and the collected data is shown in Table 4.

Our client network is modeled as a cluster of  $N_H$  hosts. In our simulation, this number is set to 100, which is roughly the number of machines in the public instructional laboratories in our department. Each host generates requests to replicated servers independently according to a Poisson process with arrival rate of  $\lambda_H$ . Different arrival rates are used to model different usage levels of the labs in our department. Usually the labs are heavily used during daytime and lightly used during the nights. Only the number of requests generated by the hosts are considered without simulating the network condition of labs.

Table 4 Route Metrics of Yahoo Replicated Services

Location	Hops	Latency (ms)
United States	15	66
Canada	15	69
United Kingdom	18	154
Australia	20	425
Singapore	15	340

A change in the route metric occurs as another Poisson process with rate of  $\lambda_R$ . In our simulation, this parameter is fixed to approximate the real situation between our local network to the five targeted Yahoo sites. Based on the data we observed, the hop number change is modeled as a weighted discrete uniform distribution. The latency change is modeled as a weighted linear combination of Gamma distributions  $\alpha$ ,  $\lambda$  and a uniform distribution, which happens as a Poisson process with rate of  $\lambda_L$ . For some sites, the Gamma distribution has been reported as the model of latency change<sup>3</sup>. But for many other routes, the latency change is so vigorous that it cannot fit into a Gamma distribution. To simulate such a condition, a uniform distribution is introduced to the model. The parameters  $\alpha$  and  $\lambda$  found by manually fitting the data we collected. The client DNS caches domain-name-to-IP-address mapping during the simulation period. It refreshes the route metric information at a constant rate of  $TTL_{DR}$ . The resolver at each host refreshes the knowledge of the best replicated server at a constant rate of  $TTL_H$ .

#### 4.6.2 Simulation results

In our simulations, the following two metrics are used to evaluate the performance: total hops traversed by all the packets traverse, and total network latency (in milliseconds) experienced by all the packets.

Figure 14 shows the hops traversed by all the packets during different length of simulations and on different host request arrival rates (RI). The route selection criterion is shortest path first. The figure shows that the total number of hops increases linearly as the simulation time increases. In all cases, the number of hops experienced by the S3 method is 12% less than that generated by the RR method. We have 3 different sites in these experiments with the same mean of the path length (15). If the difference between different routes becomes larger, the saving due to S3 would be greater.

Figure 15 shows the hops versus  $TTL_{DR}$  (the frequency DNS refreshes the routing metric from routers). It should be noted that the experiments were performed under different  $TTL_H$ . Each dot in the graph represents the results of 6 experiments. The experiments use the same DNS method, host request rate,  $TTL_{DR}$  value, but with different  $TTL_H$  values at 300, 600, 900, 1800, 2700 and 3600 seconds. Since each set of the 6 numbers are similar (less than 1% difference), the average is used to represent them.

For the round robin method, there is no concept of server selection based on the routing metric. It is not affected by the value of  $TTL_{DR}$ . For S3, the hops increases when  $TTL_{DR}$  becomes larger. This suggests that in order to save hops, the DNS needs to refresh the routing metric information more frequently. The slope of the line has a major change when  $TTL_{DR}$  changes from 3600 seconds to 1800 seconds, which marks the critical point where DNS changes from the status of being able to update routing information promptly to the status that it can no longer do so.

---

<sup>3</sup> Based on work done by the ACS lab, Dept. of Computer Science and Engineering, Michigan State University.

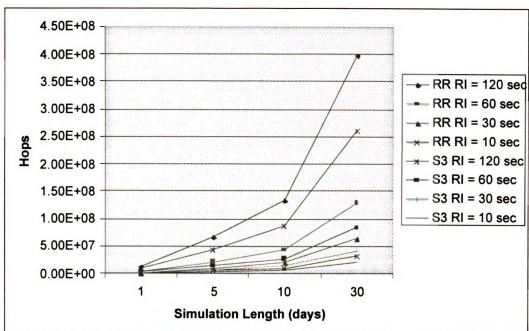


Figure 14. Hops vs Simulation Length

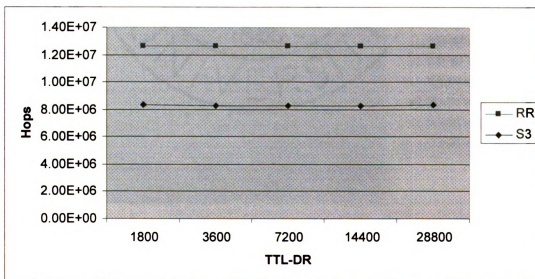


Figure 15. Hops vs  $TTL_{DR}$  ( $\lambda_H = 1/60$ )

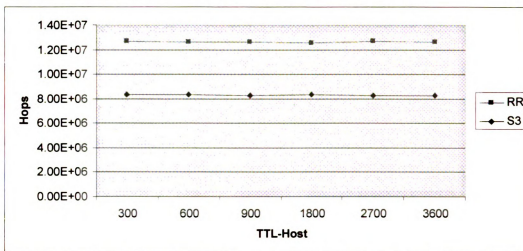


Figure 16. Hops vs TTL<sub>H</sub> ( $\lambda_H = 1/60$ )

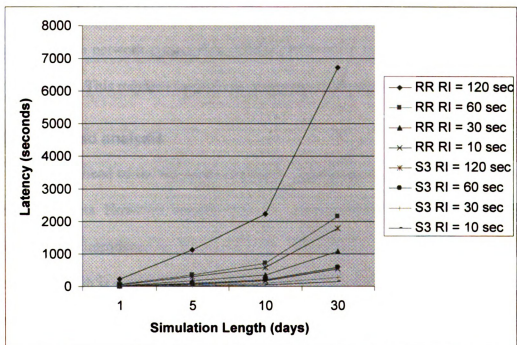


Figure 17. Latency vs Simulation Length

Figure 16 shows hops versus  $TTL_H$ . Similar to Figure 15, this is an aggregate representation of a set of experiments. Each dot in this figure corresponds to different  $TTL_{DR}$  values at 1800, 3600, 7200, 14400, and 28800 seconds. The graph shows that smaller  $TTL_H$  results in larger hop saving.

A large  $TTL_H$  means that the host machine refreshes a domain-name-to-IP-address mapping for a long period. In practice, it may also be a result of longer session. In a long session, once a host establishes a connection with a replicated server (such as TCP), all packets of this session should be forwarded to this particular server no matter how the route metric changes at the DNS.

Figure 17 shows the simulation results of finding the best latency routes. Overall, they are very similar to the results based on hops, but the latency changes may occur frequently due to network congestion, and the number of hops between two networks is relatively stable. This requires the DNS to collect the route metric more frequently.

### **4.6.3 Overhead analysis**

The overhead of the S3 method mainly occurs at the exchange of routing metric collection packets. However, compared to the total packets forwarded by a router, the traffic overhead introduced by S3 is negligible. The smallest  $TTL_{DR}$  in our simulation is 10 seconds, which is significantly smaller than what is used in a real route (current router information exchange happens every 30 seconds). Since the ICMP query and response each will need one IP packet, there will be only 2 IP packets exchanged every 10 seconds.



## 4.7 Conclusion

In this chapter, we presented the last QoS component: S3, a new server selection scheme. After reviewing related works, we presented the design of S3 and the way S3 is integrated into SIMA. We also discussed the properties of S3 and related issues. Then, the design and results of our simulation are presented.

For replicated services, a good server selection mechanism is of the key interest to both the service provider and the clients. It directly decides the eventual system performance such as latency perceived by the users, as well as how effectively the additional servers and related resources enhance service capacity.

Traditional client side server selection mechanisms are either static or dynamic. Static methods are of lower overhead, since the clients are provided with selected server directly, based on history information and stable network information such as the distance to the servers in number of hops. The problem is that the current network condition is not known to the clients in making the selection, so a currently busy server could be chosen or the network which carries the connection could be congested. Dynamic methods make server selection based on more up-to-date information, but a lot of overhead is introduced due to large number of and possibly redundant server-probes performed by the clients.

S3 uniquely combines the DNS server's central position advantage and up-to-date network information from the router. The result is a fast, accurate and low-cost server selection scheme. S3 proposes a number of extensions to current DNS and routers. With S3, users may choose the best server among the replicated servers according to their preferences. The selection metric may be hops, latency, monetary cost or a combination

of them. In short, S3 enables server selection based on fresh, dynamic information without incurring the overhead of client probing. Our simulation results show that S3 significantly reduces network resources usage and the latency perceived by the user.

Mobile hosts usually possess lower computing power and slower link than wired nodes. In addition, mobile hosts encounter handoffs. As the result, they are more sensible to quality of the service, and an appropriately selected server is of even more importance.

At the same time, these exact two attributes make traditional approaches difficult to apply in wireless networks. Constantly changing topology and network condition makes it very hard to collect network information either from the clients or the servers. Limited computing and bandwidth resources of the clients make them hardly able to participate in server selection processes.

With S3, these two problems are avoided. Now each time a client performs handoff, a new server can be chosen automatically. Since it is always provided by the current network (the router, which is in the best place to observe the network condition), the information is always correct and up-to-date. Also, there is no need of the client's participation, the mobile clients' limited resources no longer pose difficulties.

SIMA also helps mobile clients with server selection during handoff processes. Mechanisms are provided to enable the backbone components, including the BS, the DNS and the router, to select server for mobile hosts automatically when handoff happens. With support from SIMA and S3, mobile clients enjoy "always-best" server selection with low cost, fast speed and high accuracy.

## 5 Wireless Secure Multicast

In this chapter, we present the last component of SIMA, a secure multicast scheme named *Secure Transmission Backbone (STB)*. For many applications, multicast already showed its superiority than traditional unicast. Some applications involve many parties by nature. Multicast is a superior solution for such applications because it is logically clearer, as well as more cost-effective. For many multicast applications, open group scheme is not applicable. Security must be enforced. Examples include confidential audio and video conferencing, sensitive database synchronization, software update distribution, stock market information update and paid newsletter/information services.

Such needs and applications exist in wireless networks as well, so in the design of SIMA, secure multicast was an important issue. We designed a new secure multicast scheme which can be used for heterogeneous environment where both wired and wireless networks exist. In addition to addressing the common secure multicast issues found in both types of networks, it particularly helps to overcome the problems imposed by wireless networks' inherent attributes: high bit error rate, frequent handoff and limited resources.

### 5.1 Background

To support multicast on the Internet, two components are needed: multicast routing and multicast group membership control. With secure multicast, security measurements are introduced on one of these parts or both, depending on the target security level and the specific design.

### **5.1.1 Multicast routing protocols**

Multicast routers execute a multicast routing protocol to define delivery paths that enable the forwarding of multicast packets. The most widely used multicast routing protocols include Distance Vector Multicast Routing Protocol (DVMRP) [17], Multicast OSPF (MOSPF) [18], Protocol Independent Multicast (PIM) [80], and Core-Based Trees (CBT) [14][15].

As the name suggests, Distance Vector Multicast Routing Protocol (DVMRP) is a distance vector routing protocol. DVMRP constructs source-based multicast delivery trees using the Reverse Path Multicasting (RPM) algorithm. The RPF principle is quite simple: if a packet arrives via a link that is the shortest path back to the source of the packet, then forward the packet on all outgoing links. Otherwise, discard the packet. In DVMRP, the RPM distribution tree is created on demand to describe the forwarding table for a given source sending to a multicast group. The forwarding table indicates the expected inbound interface for packets from this source, and the expected outbound interface(s) for distribution to the rest of the group. Forwarding table entries are created when packets to a "new" (source, group) pair arrive at a DVMRP router. As each packet is received, its source and group are matched against the appropriate row of the forwarding table. If the packet was received on the correct inbound interface, it is forwarded downstream on the appropriate outbound interfaces for this group.

DVMRP's tree-building protocol is often called "broadcast-and-prune", because the first time a packet for a new (source, group) pair arrives, it is transmitted towards all routers. Then the edge routers initiate prunes. The prune process is as follows. Multicast routers periodically transmit IGMP (Internet Group Management Protocol) Queries to update their knowledge of the group members present on each network interface. If the

router does not receive an IGMP Report from any members of a particular group after a number of Queries, the router assumes that group members are no longer present on the corresponding interface. Assuming this is a leaf subnet, this interface is removed from the delivery tree(s) for this group. This way, the distribution tree is quickly trimmed to serve only active receivers.

MOSPF is an extension of Open Shortest Path First Routing Protocol (OSPF). OSPF is a link state routing protocol, which requires that participating routers periodically monitor the state of all of their neighboring links. This status information is then transmitted to all other participating routers by means of a special purpose flooding protocol. To support multicast routing, the presence of a multicast group on a link now becomes part of the "state" of that link. Thus, whenever a group appears or disappears, the state of that link changes, resulting in the designated router for that link flooding the new state to all other routers in the network. Therefore, MOSPF routers have complete knowledge of which groups are present on which links, throughout the domain of operation. Using this information, a router can compute the shortest path tree from any source to any group. Routers receiving multicast packets use the same computation to decide if they fall within the computed delivery tree with respect to the packet's source, and if so, to which next hop(s) a packet should be forwarded.

DVMRP and MOSPF are so called dense mode routing protocols, because they assume the existence of multicast group members are very dense (almost everybody wants to participate). In contrast, new protocols which assume that members are sparsely located (almost everybody does not want to participate) have been proposed and implemented. The representatives are Protocol Independent Multicast - Sparse Mode

(PIM-SM), and Core-Based Trees (CBT). There is another version of PIM: Dense Mode, which is very similar to DVMRP, but with some modifications to reduce overhead.

PIM-SM and CBT are very similar. Both of them set up rendezvous points and receivers receive packets through these rendezvous points. Also they are required to send explicit join request to the group in order to participate. We will discuss CBT in more detail in next section.

### **5.1.2 Issues in secure multicast**

Secure multicast differs with open group/plain text multicast mainly in two aspects: membership control and transmission secrecy.

A group membership protocol is used to control membership of specific multicast group. When a host joins a multicast group, it transmits a join message for the group(s) that it wishes to participate. Upon receiving such a join message, a multicast group membership controller authenticates the request and accepts or denies the request. For open groups, the first step simply utilizes Internet Group Management Protocol (IGMP), and the second step does not exist at all, because anybody can participate and even assumed willingness in participation beforehand. For secure multicast, membership must be controlled. Membership qualification is decided by the purpose of the group. For example, each standard group of IETF or IEEE could form a secure multicast group to facilitate the discussion of their work. Then the qualified members should be the people either working with the group or being invited to join the discussion.

Group membership control is closely related to transmission secrecy, because naturally, a restricted group usually wants to keep the transmitted material available only to its members. That means: transmissions must be done in a way that only the members

can access the data, so messages must be encrypted. A straightforward solution is as follows: For each user of the group, encrypt the message with the user's public key and transmit the message to him/her. If we use  $N$  to denote the size (the number of members) of the group, there will be  $N$  encryptions and  $N$  transmissions for each multicast message.

As a multicast group becomes large, this 1- $N$  mode apparently incurs too much overhead. To avoid such overhead, and at the same time keeping secrecy, the message can be multicast with encryption using a key known by all members. That key is usually referred to as Data Encryption Key (DEK). For large multicast groups, it is not easy to provide a mechanism to enable users to obtain the same DEK without revealing the key to unauthorized parties. Also, securely re-keying becomes a difficult task. This problem is often referred to as *multicast key management problem*.

At first glance, multicast key management may not be of much challenge. But the fact is that there are many issues and situations need to be considered, and up to now there is no perfect solution. When the size of a multicast group becomes very large, membership changes frequently and each change requires a new key distribution or re-key process. There are three basic situations need to be handled. Firstly, when a user joins a multicast group, the DEK should be given to that new member. Naive solutions use a centralized server, and each new member obtains the key from the server. They are not scalable since the server is a single point of failure. Secondly, after participating in a group for a while, members may leave the group, willingly or not. The current DEK must be changed to prevent that member from further accessing transmitted information. Naive solutions suffer the  $N$  encryptions/transmissions problem, because typically they use a key server to generate a new key and to send  $N$  re-key messages. Each message is

encrypted by a different member's public key and sent to that member. Thirdly, after a DEK is used for a while, it will "wear out" and is treated not as secure as before. From time to time, a new DEK needs to be created and distributed to all users.

### **5.1.3 Secure Transmission Backbone**

Multicast key management problem greatly prevented secure multicast from being extensively used. In this chapter, we will discuss some representative solutions to secure multicast, especially key management problem. Then, we describe a new scheme named *Secure Transmission Backbone* (STB). As the name suggests, this method constructs a Secure Transmission Backbone which all multicast groups can use. With such a backbone, it is no longer necessary for each multicast group to maintain its own key(s), so key management problem is solved naturally. This method is a general solution for both multicast and unicast. And by relieving the key management burden from individual multicast groups, STB tends to make multicast as easy as unicast. This work has been published in [2].

### **5.1.4 Supporting secure multicast in wireless networks**

One of the design goals of STB was to make secure multicast available in wireless networks. Particularly, the design of STB helps to overcome the problems imposed by wireless networks' inherent attributes: high bit error rate, frequent handoff and limited resources.

For wireless networks, the secure multicast group key update process is very hard to ensure due to high BERs. In wired network, the process can be expected to finish in relatively short period because the network condition is reliable. But in wireless networks, mobile hosts have to be given extra time for packet delivering retries. This



lengthen process not only causes additional overhead, but also may break down the whole multicast group. For a large group, huge number of key updates are generated, so each of them has to be finished with very limited time.

With STB, multicast DEK updates happen at extremely low frequency. If the mobile host does not move, the frequency is actually zero. The detail is explained in later sections. Certainly, this attribute cuts overhead, such as processing power and network resources. More importantly, it avoids the whole group to be broken down by frequent key updates.

Unlike other solutions, most handoffs involve only local hosts in the STB solution. In one hand, this feature ensures the multicast service will not be affected by handoffs. In the other hand, it avoids putting large overhead to the multicast group backbone. Given the large number of handoff devices and high frequency of handoffs, a multicast group backbone could be overwhelmed if they need to be involved with each handoff.

In the following of this chapter, we will first review some of the representative solutions of secure multicast in 5.2. Then in 5.3, we present our solution, STB. In 5.4 and 5.5, we describe how it fits in the SIMA structure and is applied in wireless networks. Section 5.6 discusses the properties of STB and finally in 5.7, we conclude this chapter.

## **5.2 Current secure multicast solutions**

We will check three representative works. In comparing them, we use the metrics listed in Table 5.

Table 5 Metrics for Evaluating Key Management Problem Solutions

Metric	Explanation
Robustness	Whether a scheme can keep working when some components fail. Especially, if there is a single point of failure.
Scalability	If a solution scales well when the group size increases.
Generality	Whether a scheme can be applied to general situations or it can only be used for special cases.
Cost of transmission	The resources used to transmit one message. Especially, whether messages need to go through multiple encryptions/decryptions processes during transmission.
Number of keys	Number of keys need to be remembered by each group member.
Cost of re-keying	Number of re-key messages needed for each re-key process.
Reusability	Whether key management structure is reusable for different multicast groups or it can be used only once then discarded.

### 5.2.1 Iolus

Iolus was proposed by S. Mitra [11]. In Greek mythology, Iolus helped Hercules with slaying the many-headed water serpent Hydra. In Iolus, a multicast group is organized into independent subgroups. Each subgroup can be further divided into lower level subgroups. Thus, a hierarchy of subgroups is formed. For example, in Figure 18, multicast group G consists of 4 subgroups, subgroup 1 through 4.

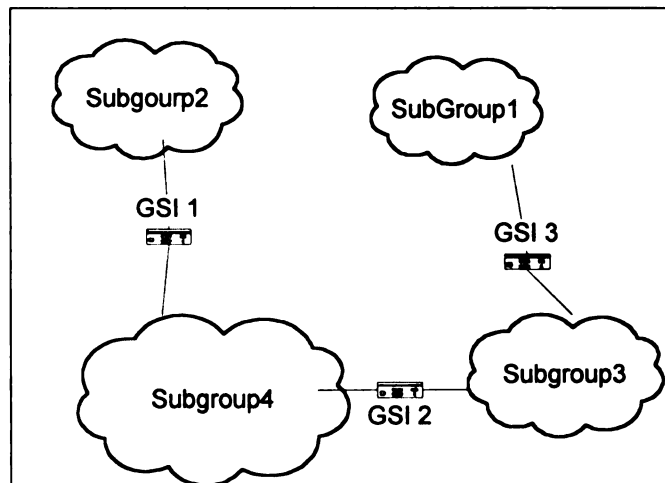


Figure 18. Iolus Structure

In Iolus, there is no global DEK for the whole group. Each subgroup maintains its own DEK. A new member joins the multicast group by joining to a subgroup and only the DEK of that subgroup is given to it. When a member is removed from the multicast group, it is removed from its own local subgroup, only the DEK of that subgroup needs to be changed and other subgroups are not affected. Since each subgroup maintains key independently, the key management problem is greatly mitigated.

In each subgroup, messages are encrypted using that subgroup's DEK, so they are only understandable by the members of that subgroup. To send messages to other subgroups, messages need to be decrypted and encrypted again using the DEK of target subgroup. This work is performed by special members called Group Security Interfaces (GSIs). Each GSI is a member of two adjacent subgroups concurrently, and it knows both subgroups' keys. For example, in Figure 18, GSI 1 is a member of both subgroup 2 and subgroup 4. It can "translate" messages between the two groups.

Iolus is easy to understand and it offers strong scalability. There is no single point of failure, and each member only needs to remember one key (two keys for GSIs).

There are also problems with Iolus. First, how to form subgroups is not specified. It is difficult for each group to form the subgroup structure by itself. Special protocols and tools are needed. Second, some mechanism is needed to enable each user to know which subgroup to join. Third, if users are allowed to join multiple subgroups, then synchronization of the membership becomes an issue. For example, if a member should be excluded from the whole group, all the subgroups should exclude that member. Fourth, Iolus needs to encrypt/decrypt a message multiple times to transfer it across

subgroups. However, the overhead can be reduced dramatically if we encrypt only the DEK instead of the data. We will discuss this technique further later.

### **5.2.2 Hierarchical Tree Approach**

Wallner et al. [12] proposed a Hierarchical Tree Approach to group membership control. A similar method using Key Graphs is proposed by Wong et al. [13]. In the Hierarchical Tree Approach, one DEK is shared for the whole group. But, each user remembers multiple keys. All the keys are generated by a Key Server (KS), and this server organizes all the keys into a virtual hierarchy. To change the DEK, instead of  $N$  messages, only  $O(\log N)$  re-key messages are needed.

In Figure 19, the multicast group consists of 16 users, and each user keeps 5 keys. In the KS, all the keys are arranged as 5 levels. The DEK is the key at the root, in this case, Key 15. Each user has a unique pairwise key known only to itself and the KS. It also knows the 4 keys along the path between the user and the root. When a new member joins the group, the KS puts this member into a subgroup and sends the member 5 keys. For example, if Host 2 just joins, it will receive Keys 1, 9, 13 and 15 and its unique pairwise key. When a member leaves the group, the DEK needs to be changed. To perform the re-key operation, several messages will be sent to the members. For example, suppose Host 2 is leaving the group, then all the keys known by Host 2 should be replaced. The KS will firstly send a re-key message containing a new Key 1 to Host 1, encrypted using H1's unique pairwise key. Then, two re-key messages containing a new Key 9 are sent, one encrypted with new Key 1, the other encrypted with Key 2. The first message can only be decrypted by Host 1 and the second message can only be decrypted by Host 3 and 4. Afterwards, two re-key messages to replace Key 13 are sent, encrypted

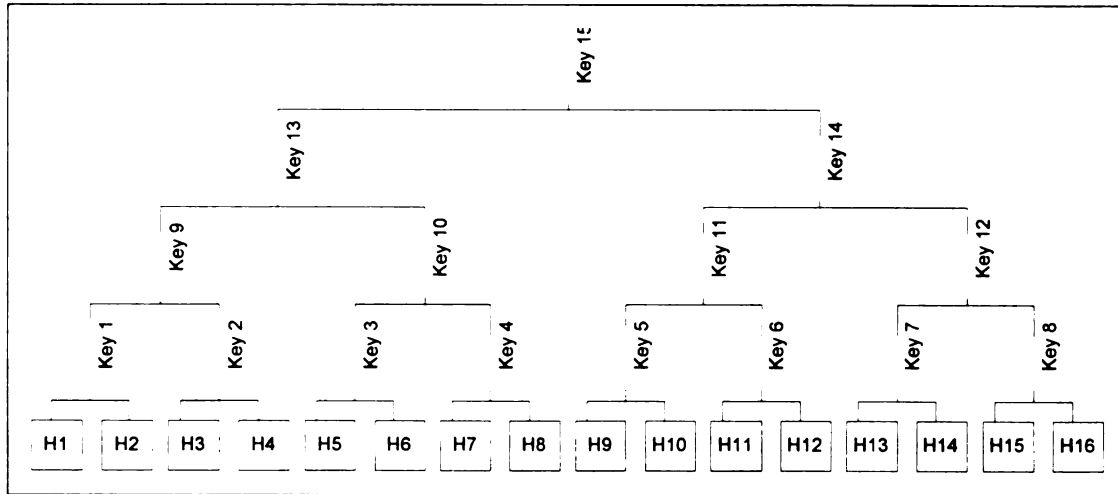


Figure 19. Hierarchical Tree Approach<sup>4</sup>

using new Key 9 and Key 10 respectively. These two messages can be decrypted only by (Host 1, Host 3, Host 4) and (Host 5 through Host 8). After new Key 13 is set up, the KS sends another two re-key messages to replace Key 15, encrypted using new Key 13 and Key 14 respectively. So, in total, 7 messages are used to perform this re-key process.

The Hierarchical Tree Approach successfully reduced the number of re-key messages from  $N$  to  $O(\log N)$ . But the price is that each user needs to keep multiple keys. The key server is involved in all aspects of the key management process, but with replicated servers, a single point of failure can be avoided.

### 5.2.3 Core Based Tree

Core Based Tree (CBT) approach is proposed by A. Ballardie [14][15]. In this method, Group initiator designates a number of Cores, and the routing tree connecting these cores is called core tree. A CBT multicast group is shown in Figure 20. Each core receives the membership control information from the group initiator, which is in the

<sup>4</sup> Graph adopted from [12].

form of either a member inclusion list or member exclusion list. After the core tree is set up, regular members can join the group by sending request to one of the cores. When the request is approved, the path connecting the member to the core tree forms a new part of the group multicast tree. Similar to Hierarchical Tree Approach, this method uses a key distribution center (KDC), but in a different way. When the group is created, the KDC generates two keys: Data Encryption Key (DEK) and Key Encryption Key (KEK). Each time a new member joins the group, it is given these two keys. To re-key the DEK, the KDC multicasts to the group a new DEK encrypted with the KEK, then after a predefined time interval, members begin to use the new DEK to encrypt messages.

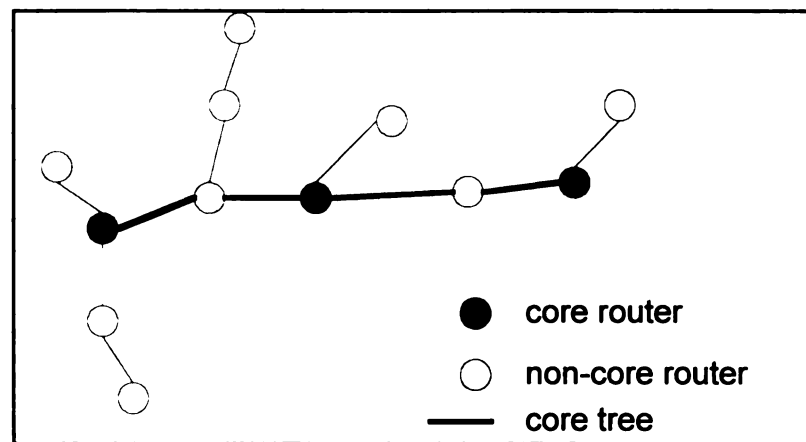


Figure 20. CBT Structure

CBT is very scalable, because each node on core tree can perform membership authentication and key distribution. KDC is required to generate new keys, but if there are multiple KDCs to use, there is no single point of failure.

However, there are some limitations on the use of CBT method. CBT is coarse-grained. It assumes that in a multi-access network, either all the hosts participate in a multicast group or the group has no member at all in the given network. For some multicast groups, that may not be the case. Further, CBT assumes that a member will stop

receiving any multicast message once he/she leaves or is removed from the group. Again, this is not always true. If a user can tap to a link used by the core tree after he/she is removed from the group, he/she will be able to see all the messages. Then, the user can easily decrypt all re-key messages to know new DEKs, since new DEKs are encrypted using KEK and it is not changed. Thus, CBT cannot securely remove such a user from the group.

Table 6. Comparison of Key Management Solutions

Metric	Iolus	Hierarchical Tree Approach	CBT	STB
Existence of single point of failure	N	Y	N	N
Scalability	Good	Good	Good	Good
Generality	Good	Good	Poor (Strong Assumptions)	Good
Multiple encryptions/decryptions	Y	N	N	Y
Number of keys for each member	1	$O(\log N)$	2	$ \text{Neighbor} $
Number of messages for each re-key process	$N/(\text{number of subgroups})$	$O(\log N)$	1	N/A
Reusable structure	Y	N	N	Y

### 5.3 Secure Transmission Backbone

To solve Multicast Key Management problem, in [2], we proposed an approach that is dramatically different than existing methods. We construct a secure transmission backbone (STB) which all multicast groups can use. The result is that it is no longer

necessary for each multicast group to maintain its own key(s), so key management problem is solved automatically.

### **5.3.1 Public STB**

In many security solutions, key distribution servers or security authorities are used [8][9][16], and all the users trust them. Often, these servers or authorities can access any data being transmitted because they know all the keys. STB also uses the concept of trusted entities and further relaxes the constraints. In STB, the trusted entities include all the routers performing multicasting, which is a reasonable assumption under most situations. For applications or multicast groups which do not accept this assumption, STB can work in another mode to ensure the desired level of security, certainly, at higher cost. That case is discussed in section 5.3.2.

In STB what we build is a trusted backbone which consists of a number of routers and provides secure transmission for all multicast groups. Instead of maintaining its own key(s), each multicast group can rely on this backbone for secure transmission. Each group just sends packets to the backbone, and then the packets will be delivered to the group members and nobody else. So except the routers along the path, nobody else can access the transmitted data.

To ensure transmissions through the backbone secure, STB modifies current transmission procedure in two aspects. First, public key information is added to the routing table so that each router knows the public key(s) of neighboring router(s). The packets used by the routing protocols to exchange routing information are augmented. A new field is added, so that public keys will be exchanged each time routing tables are updated. With Open Shortest Path First Routing Protocol (OSPF), the standard internet



interior gateway protocol, neighboring routers exchange routing information periodically or when major routing condition change occurred (new link, bandwidth increase etc.). The addition of the new field enables each router to exchange public key information with neighboring routers.

Second, packets are encrypted just as in all transmission methods which provide secrecy. With STB, there is no global DEK. Security of transmission is achieved by securely transmitting packets on each hop. Certainly, if each hop encrypts/decrypts the whole data packets, the overhead would be very large. We avoid such high overhead by encrypting/decrypting only the key part instead of the entire packets. The scheme works as follows.

- The sender encrypts the data using a DEK, and encrypts the DEK using the public key of the first router. The packet containing these two parts is then sent to the first router. Figure 21 shows the structure of the packet. Here, each DEK is only used for a specific packet, and new DEKs will be generated and used for other packets.
- Upon receiving a packet, a router decrypts the DEK using its private key, and encrypts the DEK using the public key of next router/host. The data

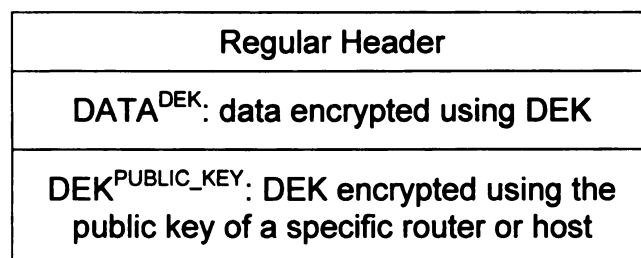


Figure 21. STB Data Packet Format

part of the packet is not modified. Then, the new packet is forwarded to next router/host.

- This process is repeated until the packets are transmitted to the receiver.

In this scheme, the data part of each packet is not modified along the path. Routers only need to decrypt/ encrypt the DEK part. An example is shown in Figure 22. Host 1, 2 and 3 belong to a multicast group. The secure transmission backbone consists of three routers: A, B and C. The actual multicast routing protocol in use is not important, if only they are enhanced with the encryption/decryption function STB requires. Host 1 is sending a packet to the group. It encrypts the data using a DEK, and encrypts the DEK with the public key of its gateway router, Router A. Then it sends the packet to Router A. From the multicast routing protocol in use, Router A knows that it needs to forward the

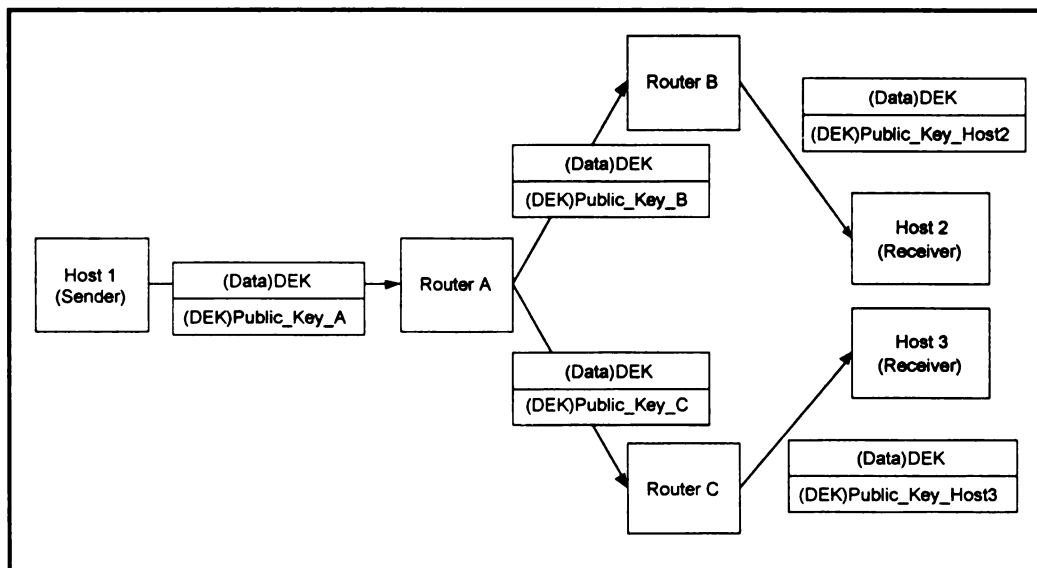


Figure 22. STB Transmission Sequence

packet to Router B and C. To forward the packet to Router B, it decrypts the DEK, then encrypts it using the public key of B. To forward the packet to Router C, it decrypts the DEK, then encrypts it using the public key of C. Upon receiving the packet, Router B leaves the data part intact, but decrypts the DEK and encrypts it using the public key of Host 2. When the packet is transmitted to Host 2, Host 2 decrypts the DEK using its own private key, then decrypts the data using that DEK. The packet sent to Router C and Host 3 is treated similarly.

### **5.3.2 Private STB**

For multicast groups which require extremely high-level security, no entities can be trusted. The above public secure transmission backbone can no longer be used. The multicast group must encrypt the data so that it is only understandable by the group members. Now the multicast group needs to construct its own private secure transmission backbone, which is not accessible to other groups. Certainly, the reusability will be lost.

The private backbone is constructed and used in a very similar way as the public one. The basic procedure is:

- The group initiator selects a number of members to form a private STB. These members are called backbone members, and all members of the group can be reached through them. In a sense, the backbone members work as routers for the specific group.
- The backbone members exchange their public keys.
- Each regular group member registers its public key with one of the backbone members (the nearest one or one chosen based on other metrics).

- All packets are sent to backbone members and then forwarded to all group members. As shown in Figure 21, each packet consists of data and a DEK. The data is encrypted using the DEK, and the DEK is encrypted using the public key of next router/host.
- The sender sends the packet to the first backbone member, which forwards the packet to all the other backbone members. Each time, the DEK is encrypted using the public key of the target backbone member. Finally, each backbone member forwards the packet to all regular members who have registered public key with it. Again each time, the DEK is encrypted using the public key of the target member.

As in the public STB structure, with this method, the multicast group does not need to maintain a global DEK, so the key management problem is solved.

## **5.4 Applying STB in wireless networks**

STB provides a scalable, low overhead and secure way to perform multicast. However, in wireless environment, STB cannot be used directly due to the frequent handoffs MHs encounter. In designing SIMA, we included support for STB to make secure multicast available to wireless networks.

### **5.4.1 Supporting Public STB**

To participate in a multicast session, the essential requirement on a MH is to know the directly attached router's public key. And in the wireless environment, the directly attached router is a BS. The challenge lays in the handoff. Each time the MH performs a handoff, to continue its multicast session without interception, the MH has to

know the public key of next BS in advance, and packets for the MH must be forwarded to the new position.

To notify MHs the public key of a BS, the best solution is to have the BS include the public key information in the broadcast beacons. But in the case of vertical handoff, the validity of the public key is not proved by default. In fact, BSs have to use key certificates to prove the validity of the keys. When MH perform handoffs horizontally, it is reasonable to assume that the related BSs use the same certification authority to sign their keys. But if the MH is performing vertical handoffs, this assumption becomes less realistic. It is common for different networks to use different key certificate authorities. The new key certificate authority information should be provided to the MH by the GW or BS of current network, an entity the MH can trust.

In supporting packet forwarding, the STB daemon of the original cell must decrypt the DEK and encrypt it with either the key of the MH or the next BS. In STB header, there is a field named “UDK” (using destination key). A router does not perform the regular decryption/encryption of DEK if this field is 1, because the DEK is already encrypted using the public key of the final destination party. So in our case, the STB daemon in the old BS encrypts the key with the MH and forward to the new cell.

So altogether, the following mechanisms are included in SIMA to support STB.

1. GWs exchange key certificate authority information. If there are different authorities, such information is made aware to all the BSs.
2. In the beacons of each BS, the following information is broadcast: the public key of the BS, public key and other information of all the key certificate authorities used in neighbor networks. It is not the case that all

the information is included in every beacon. Instead, the information is broadcast in a round robin fashion. For example, the first beacon includes the BS's public key. The second beacon includes the public key of the first authority and third beacon serves for the second authority, and so on.

3. The new BS joins the corresponding multicast group in advance of the handoff, which is done by sending an IGMP join group message to the upper layer router. In fact, this step is not for STB supporting only. Having the new BS join the multicast group is necessary no matter what secure multicast scheme is used.
4. Forward multicast packets from the old BS using the public key of the MH.

#### **5.4.2 Supporting Private STB**

For private STB, additional support is needed for core member registration. Each MH registers with a STB core member in order to participate in a particular secure multicast group. When the MH moves, the registration may need to be changed. Frequently, different networks are linked to different core members due to geographical or management reasons. In such case, the MH should change its registration to the new core member. It is true that packets can still be forwarded to the MH from the original core member eventually, especially with the help of HOPOVER, but we argue that to change to the new core member is a better solution. First, the forwarding of packets apparently involves additional overhead. Secondly, since the new core member is preferred by the new network, it is frequently the case that the new core member is of shorter distance or is connected via a better link. Last but not least, it cannot be

anticipated that HOPOVER is always available from the original network from where the MH is leaving. Without the help of HOPOVER, packet forwarding will not be available from the old network. If the MH does not change registration to the new core member, it will not receive packets until the original core member is notified by mobile IP about the new FA of the MH. That process will cause many packet losses and may cause interruption of sessions.

In SIMA, the process of re-registration involves only the new BS and the MH.

- The MH generates a STB registration requirement, which includes a timestamp and is signed using its own private key
- The MH includes the STB registration requirement into the HOPOVER Handoff packet as an option.
- The new BS HOPOVER daemon detaches the STB registration requirement and forwards to the preferred STB core member for that multicast group.
- The new core member verifies the validity of the registration requirement and registers the MH to its attached member list.
- The new core member notifies the old one about the re-registration. After certain amount of time, the old core member removes the MH from its attached member list and stops forwarding packets for it.

In addition, if the old BS is HOPOVER-aware, it will forward packets to the new BS. So the MH experiences less packet losses and smoother transition. This is part of HOPOVER standard operation.

## **5.5 How SIMA QoS components help STB**

By itself, STB provides no QoS guarantee, because it is not part of its job. The network is supposed to provide help in QoS assurance. In wireless network, as we discussed, QoS assurance is very complicated. Fortunately, SIMA provides help with its QoS components: WRSVP, HOPOVER and S3.

When a MH performs a handoff, all the three QoS components of SIMA participate to ensure the QoS. First, if the MH participated in a private STB group, SIMA application server selection component helps to choose a closest STB core member. Then WRSVP component jumps in to make reservation in the new cell and along the path from the cell to the multicast group sender or core member. After that, HOPOVER daemon in both networks work together to ensure a smooth handoff. For static mobile hosts, they do not need to utilize HOPOVER. But still, RSVP and S3 help in resource reservation and core member/server selection respectively.

## **5.6 Discussion**

STB can be utilized in both wired and wireless networks. In both cases, STB helps clients to participate secure multicast at high level of security and low overhead.

### **5.6.1 Attributes of STB**

STB is quite unique and the last column of Table 6 summarizes it. Its main attractive attributes include:

- Robust. There is no single point of failure in STB. In addition, the failure of small number of routers will not cause big trouble. The underlying routing protocol will automatically choose alternative routes.



- Reusable. All multicast groups can share a single STB.
- Secure. STB provides a secure transmission method and it avoids many potential security risks which are possible if individual multicast group fails to manage keys properly.
- General. STB is not limited to be used for multicast. It is also suitable for unicast. Generally, it is hard to securely transmit information between two users who do not know each other's public key and there is no shared secret. STB provides a natural solution, since basically any packet can be securely transmitted to anyone else.
- Scalable. STB relies on existing multicast and unicast routing protocols, and inherits their high scalability
- Low overhead in transmitting and re-keying. As shown in section 4.3, the overhead of transmitting is low. STB does not need to re-key as regular methods do, the cost associated with re-keying is almost removed.
- Small number of keys. Each router only needs to remember the public keys of its neighbors, which is a small constant number.

There are also some aspects of STB which need further improvement. The first is the need of multiple encryption/decryption processes during packet transmission. Although only DEK part of the packets needs to be processed, it does add cost to transmission. A useful observation is that the number of encryptions and decryptions can be reduced dramatically if each router knows a little more about global routing topology. That is, if each router knows a few remote (not neighboring) routers' public keys, then it is not necessary to decrypt and encrypt the DEK at each router. OSPF does not exchange

global information. However, all public keys known by a router can be added into the routing control packets it sends. Then each router can gradually learn the public keys of other routers in the network through routing information exchanges.

Figure 23 shows a transmission process. Router A is sending a packet to the other members of a multicast group, Host 2 and 3. With standard STB method, each router would perform a decryption/encryption process. If A knows the public key of Router D and E, and it knows that the packet will go through them eventually, then it could directly encrypt the DEK with the public key of D and E respectively. Router B and C no longer need to do the decryption/encryption work and the overhead is reduced.

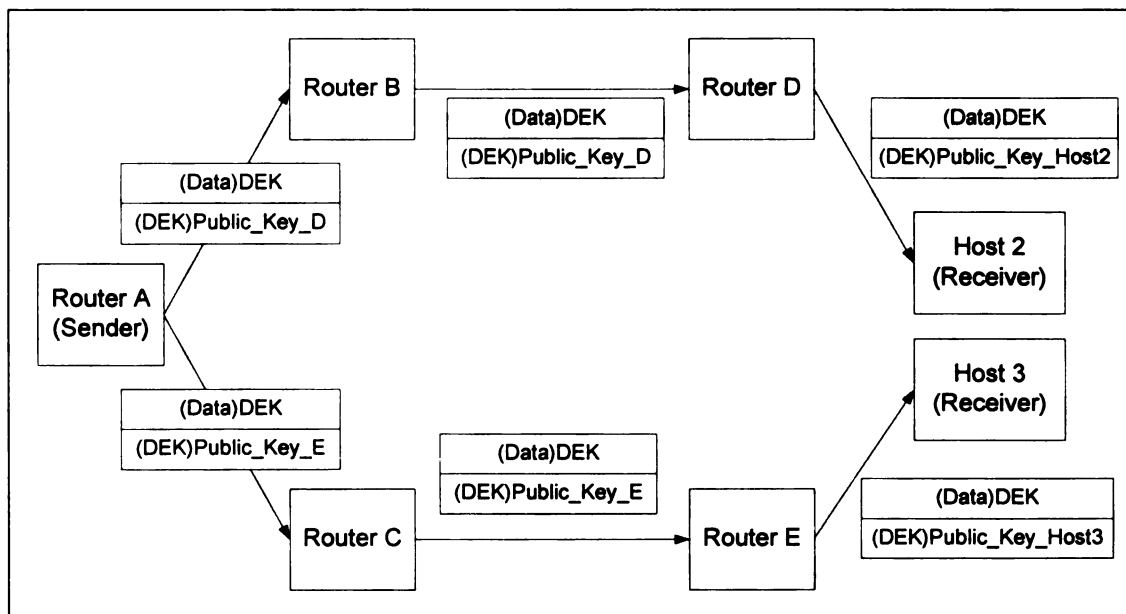


Figure 23. Reduce Number of Encryptions/decryptions

STB relies heavily on routers or backbone members. It is very important to protect those nodes from being attacked, or to detect attacks on them. Some methods

have been proposed to detect misbehaving routers. STB will benefit from those methods. Interested readers are referred to [76][77] for more detailed description.

### **5.6.2 The benefits of STB in wireless networks**

Compared with Iolus, HTA, CBT and other secure multicast mechanisms, STB is particularly suitable for wireless network and mobile clients. In fact, it is not a coincidence, one of the design goals of STB was to make secure multicast available in wireless networks.

The biggest benefit from STB is that it requires no multicast DEK updates. This attribute reduces overhead, such as processing power and network resources. For mobile devices, be free with key update processing mean the limited bandwidth and processing power can be used for more valuable purposes.

More importantly, as discussed earlier, in wireless networks, key update processes are hard to ensure due to high BERs. Mobile hosts have to be given long time for packet delivering retries. This lengthen process could break down the whole multicast group, because if huge number of key updates need to be processes, each of them has to be finished with very limited time.

Another key STB benefit is its local involvement feature. Handoffs with public STB require no action from a central control at all. All the changes happen locally. For private STB, it shared similar background design idea with Iolus and CBT, which is: divide and conquer. When different serving point is needed, they all require the members to perform the re-registration process. Here, involvement from the multicast backbone is unavoidable.

## 5.7 Conclusion

In this chapter, we first reviewed some of the representative solutions of secure multicast, with special focus on key management problem. Then we presented our solution, Secure Transmission Backbone. With that, we described how it fits in the SIMA structure and how it is applied in wireless networks.

Key management problem is a big obstacle in utilizing secure multicast in both wired and wireless networks. When the multicast group becomes very large, due to the very high frequency of re-key processes, it is very difficult to provide a re-key mechanism that enable users to obtain the same DEK and without revealing the key to unauthorized parties.

Secure Transmission Backbone makes the realistic assumption that routers are trustable for most applications and multicast groups. Under this assumption, STB provides a general solution for both secure multicast and secure unicast. In STB, with a secure transmission backbone, it is no longer necessary for each individual multicast group to maintain keys. Thus key management problem is solved/avoided naturally. If a multicast group requires extremely high level of security, the assumption becomes invalid. For those cases, a private secure transmission backbone will be constructed which consists of the group's own members.

In addition to addressing those common secure multicast issues found in both types of networks, STB successfully overcome those problems imposed by wireless networks' inherent attributes: high bit error rate, frequent handoff and limited resources.

For wireless networks, the secure multicast group key update process is very hard to ensure due to high BERs. Mobile hosts have to be given extra time for packet

delivering retries. This lengthen process not only causes additional overhead, but also may break down the whole multicast group. If huge number of key updates are generated, each of them has to be finished with very limited time.

With STB, multicast DEK updates happen at extremely low frequency. If the mobile host does not move, the frequency is actually zero. Certainly, this attribute cuts overhead, such as processing power and network resources. But more importantly, it avoids the whole group to be broken down by frequent key updates.

Unlike other solutions, most handoffs involve only local hosts in the STB solution. In one hand, this feature ensures the multicast service will not be affected by handoffs. In the other hand, it avoids putting large overhead to the multicast group backbone. Given the large number of handoff devices and high frequency of handoffs, a multicast group backbone could be overwhelmed if they need to be involved with each handoff.

STB cooperates with other SIMA components. Such cooperation is represented especially apparent when a handoff happens. When a MH moves to a new cell, all the three QoS components of SIMA participate to ensure the QoS. First, if the MH participated in a private STB group, SIMA application server selection component helps to choose a best serving STB core member. Then WRSVP component jumps in to make reservation in the new cell and along the path from the cell to the multicast group sender or core member. After that, HOPOVER daemon in both networks work together to ensure a smooth handoff.

For static mobile hosts, WRSVP and S3 may still be needed when they start a new secure multicast session, based on whether the session require QoS and if multiple core members are available.

In summary, STB successfully enables secure multicast in heterogeneous networks. And participating with other SIMA components, STB enables mobile devices to participate in multimedia and real-time secure multicast sessions at the move.

## **6 Conclusion and Future Works**

In this dissertation, we proposed Supporting Infrastructure for Mobile Applications (SIMA), a framework in supporting mobile applications. This framework was designed to overcome the difficulties in wireless networks and provide support for mobile applications from multiple aspects.

Included in SIMA are WRSVP, HOPOVER, S3 and STB. WRSVP is a resource reservation protocol for wireless network; HOPOVER enables smooth handoff inter- and intra- network; S3 helps mobile devices in choosing the best application server and STB is a secure multicast solution suitable for wireless environment.

### **6.1 Resource reservation**

Resource reservation is an important part in providing QoS. The representative solution RSVP is applied in both Diffserv (as a signaling protocol) and Intserv. The essence of resource reservation is to guarantee the availability of necessary network resources for clients, such that packets can receive expected treatment.

But two big problems prevent RSVP from being used in wireless networks. One is the poor link problem, and the other is the handoff problem. Poor link problem stems from the high bit error rate associated with wireless links. Such poor links cause difficulties in reserving and refreshing resources. Handoff problem occurs when the MH moves between cells. Without special care, the MH can suffer from terminated flows due to un-established resource reservations in the new cell and along the path from the new cell to the transmission parties.

WRSVP solves the poor link problem by separating resource reservation into wired and wireless parts and by adding Ack messages into wireless message exchanges. Using pre-reserving in neighbor cells, WRSVP protects flow reservations from handoffs, both in the cell and along the path. Resource reservation becomes a natural part of handoff.

WRSVP keeps the excellent properties of RSVP. It is receiver initiated. For mobile applications, this is very important, since now the bottleneck is in the mobile receiver part. Also, resource reservations are merged in many parts of the reservation trees, thus network utilization is improved.

All modifications WRSVP introduced to regular RSVP are local, and the "interface" to other component of the network is unchanged. So for components outside of the WRSVP network (or network segment), the use of WRSVP is transparent.

## **6.2 Handoff support**

In wired networks, no matter with Intserv or Diffserv, QoS is essentially provided by allocating abundant resources (bandwidth, buffer, processing power etc.) to higher priority traffic. If only such resources are available, usually packets can be delivered as required. In wireless networks, having enough resources is still critical for QoS. But in addition to that, now we have an equally important factor, which is the performance of handoffs. Without enough resources, no QoS can be talked about. Without a good handoff scheme, no QoS can be sustained after a handoff takes place.

Correspondingly, in SIMA, QoS assurance is provided primarily by WRSVP and HOPOVER, addressing resource reservation and handoff respectively.



In wireless networks, without special care, handoffs often cause severe QoS damage. Users could experience an apparent interruption, or sometimes even the connection could get lost. HOPOVER enables smooth handoffs intra- and inter-network, and it is compatible with mobile IP.

HOPOVER was designed specifically for overlay networks, where multiple networks coexist geographically. HOPOVER facilitates inter-network cooperation. And working with other components of SIMA, HOPOVER helps to ensure QoS for mobile applications across multiple networks.

HOPOVER helps mobile devices using the following measurements: facilitating WRSVP to pre-reserve resources in the new cell and along the path from the new cell to the flow transmission parties; authenticating in advance of the actual handoff; buffering in the new network for the MH and forwarding packets from the old network to the new network. With these methods, HOPOVER significantly enhances handoff performance.

With HOPOVER, a handoff involves mostly local hosts. This feature not only reduces network overhead of remote control packets estranges, but also ensures the handoff processing to be fast and reliable.

The cooperation of SIMA components is reflected especially apparently when a handoff happens. In the handoff process, multiple components work jointly to make sure the mobile application is handed to the new cell smoothly. At the central control point, HOPOVER provides necessary support to other SIMA components, so the corresponding function will be continually available to mobile applications during the handoff process and in the new cell.

### **6.3 Server selection**

For replicated services, the server selection mechanism directly decides the eventual system performance such as latency perceived by the users, as well as how effectively the additional servers and related resources enhance service capacity.

S3 uniquely combines the DNS server's central position advantage and up-to-date network information from the router. The result is, S3 enables server selection based on fresh, dynamic information without incurring the overhead of client probing. We proposed a number of extensions to current DNS and routers. With S3, users may choose the best server among the replicated servers according to their preferences. The selection metric may be hops, latency, monetary cost or a combination of them. Our simulation results show that S3 significantly reduces network resources usage and the latency perceived by the user.

For mobile hosts, traditional approaches come across difficulties due to two fundamental properties of mobile applications. The first property is that the clients always move around and the topology and network condition always change, this makes it very hard to collect network information either from the clients or the servers. The other property is that clients are of limited computing and bandwidth resources, so they can hardly participate in server probing processes.

With S3, these two problems are avoided. Now each time a client performs handoff, a new server can be chosen automatically. Since it is always provided by the current network (the router, which is in the best place to observe the network condition), the information is always correct and up-to-date. Also, there is no need of the client's participation, the mobile clients' limited resources no longer pose difficulties.

SIMA also helps mobile clients during handoff processes. Mechanisms are provided to enable the backbone components, including the BS, the DNS and the router, to select server for mobile hosts automatically when handoff happens. With support from SIMA and S3, mobile clients enjoy "always-best" server selection with low cost, fast speed and high accuracy.

## **6.4 Secure multicast**

In supporting secure multicast on the Internet, we identified "key management problem" as a big obstacle for both wired and wireless networks. When the multicast group becomes very large, due to the very high frequency of re-key processes, it is very difficult to provide a re-key mechanism that enable users to obtain the same DEK, and without revealing the key to unauthorized parties.

Secure Transmission Backbone makes the realistic assumption that routers are trustable for most applications and multicast groups. Under this assumption, STB provides a general solution for both secure multicast and secure unicast. In STB, with a secure transmission backbone, it is no longer necessary for each individual multicast group to maintain keys. Thus key management problem is solved/avoided naturally. If a multicast group requires extremely high level of security, the assumption becomes invalid. For those cases, a private secure transmission backbone will be constructed which consists of the group's own members.

In addition to addressing those common secure multicast issues found in both types of networks, STB successfully overcome those problems imposed by wireless networks' inherent attributes: high bit error rate, frequent handoff and limited resources.

Due to high BERs, mobile hosts have to be given extra time for packet delivering retries in re-key processes. This lengthen process not only causes additional overhead, but also may break down the whole multicast group, because when huge number of key updates are generated, each of them has to be finished with very limited time. With STB, multicast DEK updates happen at extremely low frequency. If the mobile host does not move, the frequency is actually zero. Thus the high BER no longer poses a threat, at the same time, this attribute cuts overhead such as processing power and network resources.

Unlike other solutions, most handoffs involve only local hosts in the STB solution. In one hand, this feature ensures the multicast service will not be affected by handoffs. In the other hand, it avoids putting large overhead to the multicast group backbone. Given the large number of handoff devices and high frequency of handoffs, a multicast group backbone could be overwhelmed if they need to be involved with each handoff.

In summary, STB successfully enables secure multicast in heterogeneous networks. And participating with other SIMA components, STB enables mobile devices to participate in multimedia and real-time secure multicast sessions at the move.

## **6.5 Final comments and future works**

Supporting mobile applications is very challenging, also very important. With SIMA, we systematically address the needs of mobile applications, so users could utilize their favorite applications anytime and anywhere.

As mobile applications evolve rapidly, their requirement on infrastructure keeps increasing. SIMA must be improved and extended from time to time to suite the needs. For example, proxy support could be highly desired by many mobile applications,

especially multimedia ones. In addition to regular caching, proxies must be prepared for mobile environment characterized by dynamic topology, limited resources and high bit error rate. It is desirable if proxies can cope with handoffs such that information can follow the position of mobile hosts. It is also desirable if proxies can prepare layered information so mobile hosts can choose to retrieve different amount of information according to their bandwidth and processing power. For mobile hosts with abundant resources, they can choose to receive high-resolution pictures, high quality audios and at high refresh rates (for online streaming or conferencing). For others, they can still utilize those applications but with reduced requirements on resolution, quality and refresh rate. Some websites already started building or have built special version for mobile customers, but such server side movements affect only one website or service at a time. From mobile applications' point of view, working from the proxies is more effective since any improvement on proxies is effective for all services and websites they utilize or access.

Supporting ad-hoc networks is becoming a hot research topic, as they are found useful in many circumstances. In ad-hoc networks, mobile hosts could be multiple-hop away from any base station and have to rely on other hosts to relay packets for them. It becomes more difficult to guarantee QoS, or just unicast and multicast connectivity. It is necessary to make adjustments on SIMA and possibly also on ad-hoc connectivity software, so that SIMA services can be better utilized by ad-hoc mobile devices.

All SIMA components are designed to follow open standards. For example, WRSVP cooperates with RSVP natively and HOPOVER is compatible with mobile IP. With open standards as interface, new components can be easily integrated into SIMA in

the future. At the same time, such compatibility ensures smooth deployments of our new mechanisms.

Currently, there is enormous effort in standardization work of wireless technologies, such as a unified 3G cellular standard and allowing different wireless LANs coexist. A clear trend in all case is that future technologies must support TCP/IP protocol stack, no matter what air interface is used underneath. This gives SIMA a wide applicability. No matter which air interface standard, W-CDMA or cdma2000 in the case of 3G, and FHSS or DSSS in the case of LAN, SIMA can work with the new standard with easy.

## 7 References

- [1] F. Du, X. Dong, H. D. Hughes and L. M. Ni, "Resource Reservation in Wireless Networks", Symposium on Performance Evaluation of Computer and Telecommunication Systems (SPECTS 1999), July 1999.
- [2] F. Du, L. M. Ni and A. H. Esfahanian, "Towards Solving Multicast Key Management Problem", Eighth IEEE International Conference on Computer Communications and Networks (ICCCN 1999), October 1999.
- [3] W. Tang, F. Du, M. W. Mutka, L. M. Ni and A. H. Esfahanian, "Supporting Global Replicated Services by a Routing-Metric-Aware DNS," 2nd International Workshop on Advanced Issues of E-Commerce and Web-Based Information Systems (WECWIS 2000), June, 2000.
- [4] E. He, F. Du, X. Dong, L. M. Ni and H. D. Hughes, "MPEG Traffic Modeling on Wireless Networks", IEEE International Conference on Communications 2000 (ICC2000), June 2000.
- [5] M. Stemm and R. H. Katz, "Vertical Handoffs in Wireless Overlay Networks", ACM Mobile Networking (MONET), Special Issue on Mobile Networking in the Internet, Winter 1998
- [6] R. H. Katz, E. A. Brewer, E. Amir, H. Balakrishnan, A. Fox, S. Gribble, T. Hodes, D. Jiang, G. T. Nguyen, V. Padmanabhan, and M. Stemm, "The Bay Area Research Wireless Access Network(BARWAN)," Proceedings Spring COMPCON Conference 1996.
- [7] W. Diffie and M.E. Hellman, "New Directions in Cryptography," IEEE Trans. Information Theory, vol. IT-22, pp. 644-654, 1976.

- [8] D. Otway and O. Rees, "Efficient and Timely Mutual Authentication," *Operating Systems Rev.*, pp. 8-10, 1987.
- [9] B.C. Neuman and T. TS'O, "Kerberos: An Authentication Service for Computer Networks," *IEEE Communication Magazine*, vol. 32, pp. 33-38, 1994.
- [10] Kerberos Project, <http://web.mit.edu/kerberos/www/>.
- [11] S. Mitra, "Iolus: A framework for scalable secure multicasting". *Proceedings of ACM SIGCOMM'97*, 1997.
- [12] D. Wallner, E. Harder, and R. Agee, "Key management for multicast: issues and architectures", RFC2627, 1999.
- [13] C. Wong, M. Gouda, and S. Lam, "Secure group communications using key graphs", *Proceedings of ACM SIGCOMM'98*, 1998.
- [14] A. Ballardie, "Scalable multicast key distribution", RFC 1949, 1996.
- [15] A. Ballardie, "Core Based Trees multicast routing architecture", RFC 2201, 1997.
- [16] R. Needham, and M. Schroeder, "Authentication revisited," *Operating Systems Review*, v. 21, n. 1, 1987.
- [17] D. Waitzman, C. Partridge, and S. Deering, "Distance vector multicast routing protocol", RFC 1075, 1988.
- [18] J. Moy, "Multicast extensions to OSPF", RFC 1584, 1994.
- [19] H. Balakrishnan, S. Seshan, and R.H. Katz, "Improving Reliable Transport and Handoff Performance in Cellular Wireless Networks," *ACM Wireless Networks*, 1(4), December 1995.
- [20] K. Lee, "Supporting Mobile Multimedia in Integrated Services Networks", *ACM Wireless Networks*, Vol. 2, No. 3: 205~217, 1996.



- [21] D. Raychaudhuri, "Wireless ATM: An Enabling Technology for Multimedia Personal Communication", ACM Wireless Networks, Vol. 2, No.3: 163~171, 1996.
- [22] D. Raychaudhuri, and N. Wilson, "ATM Based Transport Architecture for Multiservices Wireless Personal Communication Networks", IEEE Journal on Selected Areas in Communication, Vol. 12, No. 8: 1401~1414, 1994.
- [23] R.L.Rivest, "The MD4 Message Digest Algorithm," RFC 1320, 1992.
- [24] R.L.Rivest, "The MD5 Message Digest Algorithm," RFC 1321, 1992.
- [25] "Proposed Federal Information Processing Standard for Secure Hash Standard," Federal Register, V.57, N.21, 1992.
- [26] A. Tanenbaum, Computer Networks, 3rd edition, Prentice Hall, 1996.
- [27] B. Schneier, Applied Cryptography. New York: John Wiley and Sons, 1996.
- [28] A. G. Valko, "Cellular IP - A New Approach to Internet Host Mobility," ACM Computer Communication Review, January 1999.
- [29] The Cellular IP Project at Columbia University, <http://www.ctr.columbia.edu/~andras/cellularip/>.
- [30] IETF Mobile IP WG, <http://www.ietf.org/html.charters/mobileip-charter.html>.
- [31] DAEDALUS project at Berkeley, <http://www-daedalus.cs.berkeley.edu/>.
- [32] A. S. Acampora and M. Naghshineh, "An Architecture and Methodology for Mobile-Executed Hand-off in Cellular ATM," IEEE Journal on Selected Areas in Communications, 12(8):1365–1375, October 1994.
- [33] V. Garg and J. Wilkes. Wireless and Personal Communications Systems. Prentice-Hall, 1996. Chapter 8.

- [34] R. Ghai and S. Singh, "An Architecture and Communications Protocol for Picocellular Networks," *IEEE Personal Communications Magazine*, 1(3):36–46, 1994.
- [35] IBM Infrared Wireless LAN Adapter Technical Reference. IBM Microelectronics - Toronto Lab, 1995.
- [36] J. Ioannidis, D. Duchamp, and G. Q. Maguire. "IP-based Protocols for Mobile Internetworking," In *Proc. ACM SIGCOMM*, 1991.
- [37] J. Ioannidis and G. Q. Maguire. "The Design and Implementation of a Mobile Internetworking Architecture." In *Proc. Winter '93 Usenix Conference*, San Diego, CA, January 1993.
- [38] C. Perkins. *IP Mobility Support*. RFC 2002, Oct 1996.
- [39] S. Tekinay and B. Jabbari, "Handover and Channel Assignment in Mobile Cellular Networks." *IEEE Communications Magazine*, 29(11):42–46, November 1991.
- [40] S. Seshan, *Low Latency Handoffs in Cellular Data Networks*, PhD thesis, University of California at Berkeley, December 1995.
- [41] S. Seshan, H. Balakrishnan, and R. H. Katz, "Hand-offs in Cellular Wireless Networks: The Daedalus Implementation and Experience," *Kluwer Journal on Wireless Personal Communications*, January 1997.
- [42] A. Bestavros, M. Crovella, J. Liu, and D. Martin, "Distributed packet rewriting and its application to scalable web server architectures," In *Proc. of ICNP'98: The 6th IEEE International Conference on Network Protocols*, Austin, Texas, Oct. 1998.

- [43] V. Cardellini, M. Colajanni, and P. Yu, "Redirection algorithms for load sharing in distributed webserver systems," In Proc. of the 19th IEEE International Conference on Distributed Computing Systems, Austin, Texas, June 1999.
- [44] Cisco Systems Inc. Distributed director. <http://www.cisco.com/univercd/cc/td/doc/product/iaabu/distrdir/>.
- [45] J. Doyle. Routing TCP/IP, volume I. Macmillan Technical Publishing, 1st edition, 1998.
- [46] P. Francis, S. Jamin, V. Paxson, L. Zhang, D. Gryniewicz, and Y. Jin, "An architecture for a global internet host distance estimation service," In Proc. of IEEE INFOCOM 99, Mar. 1999.
- [47] J. Heidemann and V. Visweswaraiiah, "Automatic selection of nearby web servers," Technical Report 98688, USC/Information Science Institute, 1998.
- [48] P. S. M. Sayal and P. Vingralek, "Selection algorithms for replicated web servers," In The 1998 SIGMETRICS/Performance Workshop on Internet Server Performance, June 1998.
- [49] K. Moore, J. Cox, and S. Green, "Sonar - a network proximity service," In Internet Draft, Network Working Group, draft-moore-sonar-01.txt, Feb. 1996.
- [50] J. Moy. OSPF version 2. In RFC 2328, Apr. 1998.
- [51] K. Obraczka and F. Silva, "Looking at network latency for server proximity," Technical Report 99-714, USC/Information Science Institute, 1999.
- [52] J. Postel. Internet control message protocol. In RFC 777, Apr. 1981.
- [53] W. R. Stevens. TCP/IP illustrated, volume I: The Protocols. Addison-Wesley Professional Computing Series, first edition, 1994.

- [54] C. Yoshikawam, B. Chun, P. Eastham, A. Vahdat, T. Anderson, and D. Culler, "Using smart clients to building scalable services," In USENIX'97, Jan. 1997.
- [55] S. Chakrabarti, and R. Wang, "Adaptive Control for Packet Video", Proceedings of International Conference on Multimedia Computing and Systems: 1994.
- [56] H. Kanakia, P. P. Mishra and A. Reibman, "An Adaptive Congestion Control Scheme for Real-time Packet Video Transport", Proceedings of SIGCOMM'93: 1993.
- [57] L. Zhang; S. Deering; D. Estrin; S. Shenker and D. Zappala, "RSVP: A New Resource ReSerVation Protocol", IEEE Network, Vol. 7, No. 5: 1993
- [58] IETF RSVP group web, <http://www.ietf.org/html.charters/rsvp-charter.html>
- [59] Resource ReSerVation Protocol (RSVP) -- Version 1 Message Processing Rules (RFC 2209)
- [60] Resource ReSerVation Protocol (RSVP) -- Version 1 Functional Specification (RFC 2205)
- [61] R. Braden, D. Clark and S. Shenker, "Integrated Services in the Internet Architecture: an Overview", Internet RFC 1633, Jun. 1994
- [62] S. Blake, D. Black, M. Carlson, E. Davies, Z. Wang, and W. Weiss, "An Architecture for Differentiated Services", RFC 2475, Dec. 1998.
- [63] K. Nichols, V. Jacobson and L. Zhang, "A Two-bit Differentiated Services Architecture for the Internet", Internet Draft, <draft-nichols-diff-svc-arch-00.txt>, Nov. 1997.
- [64] Y. Bernet et al., "A Framework for use of RSVP with Diff-serv Networks", Internet draft <draft-ietf-Diffserv-rsvp-00.txt>, Jun. 1998

- [65] R. Koodli and M. Puuskari, "Supporting Packet-Data QoS in Next Generation Cellular Networks", IEEE Communication Magazine, Feb. 2001.
- [66] Universal Mobile Telecommunication System (UMTS) QoS concept, TS.23.107, version 3.1.0.
- [67] S. Dixit, Y. Guo and Z. Antoniou, "Resource Management and Quality of Service in Third-Generation Wireless Networks", IEEE Communication Magazine, Feb. 2001.
- [68] 3Gpp Web site, [www.3gpp.org](http://www.3gpp.org).
- [69] IEEE P802.11, [grouper.ieee.org/groups/802/11](http://grouper.ieee.org/groups/802/11)
- [70] HomeRF group, [www.homerf.org](http://www.homerf.org).
- [71] Bluetooth SIG, [www.bluetooth.com](http://www.bluetooth.com)
- [72] Domain Names - Concepts and Facilities, RFC 1034, 1987
- [73] Domain Names - Implementation and Specification, RFC 1035, 1987
- [74] P. Brenner, "A Technical Tutorial on the IEEE802.11 Protocol," Breezecom Wireless Communications
- [75] B. Crow, I. Widjaja, J. Kim, and P. Sakai, "IEEE 802.11 Wireless Local Area Networks," IEEE Communications Magazine, pp. 116-126, Sept. 1997.
- [76] M. Stahl, R. Buskens and R. Bianchini, Jr, "On-Line diagnosis in general topology networks", 1992.
- [77] R. Canetti, "Studies in secure multiparty computation and applications", PhD. Thesis, The Weizmann Institute of Science, 1996.
- [78] R.F.Quick and K. Balachandran, "An overview of the Cellular Digital Packet Data (CDPD) system," 4<sup>th</sup> International Symp. On Personal Indoor and Mobile Radio Commun., 1993.

- [79] M. Rahema, "Overview of the GSM System and Protocol Architecture," IEEE Commun. Magazine, vol. 31, April 1993.
- [80] C. Huitema, "Routing in the Internet", Prentice Hall, 1995.
- [81] C. M. Bowman, P. B. Danzig, D. R. Hardy, U. Manber, and M. F. Schwartz. "Harvest: A scalable, customizable discovery and access system". Technical Report CU-CS-732-94, Department of Computer Science, University of Colorado, Boulder, Colorado, 1994.
- [82] J. D. Guyton and M. F. Schwartz. "Locating nearby copies of replicated internet servers". Proceedings of SIGCOMM, 1995.
- [83] M. E. Crovella and R. L. Carter, "Dynamic Server Selection in the Internet", Proceedings of the Third IEEE Workshop on the Architecture and Implementation of High Performance Communication Subsystems (HPCS'95), 1995.

MICHIGAN STATE LIBRARIES



3 1293 02314 5976