Terms and Conditions of Use of Digitised Theses from Trinity College Library Dublin

Copyright statement

All material supplied by Trinity College Library is protected by copyright (under the Copyright and Related Rights Act, 2000 as amended) and other relevant Intellectual Property Rights. By accessing and using a Digitised Thesis from Trinity College Library you acknowledge that all Intellectual Property Rights in any Works supplied are the sole and exclusive property of the copyright and/or other IPR holder. Specific copyright holders may not be explicitly identified. Use of materials from other sources within a thesis should not be construed as a claim over them.

A non-exclusive, non-transferable licence is hereby granted to those using or reproducing, in whole or in part, the material for valid purposes, providing the copyright owners are acknowledged using the normal conventions. Where specific permission to use material is required, this is identified and such permission must be sought from the copyright holder or agency cited.

Liability statement

By using a Digitised Thesis, I accept that Trinity College Dublin bears no legal responsibility for the accuracy, legality or comprehensiveness of materials contained within the thesis, and that Trinity College Dublin accepts no liability for indirect, consequential, or incidental, damages or losses arising from use of the thesis for whatever reason. Information located in a thesis may be subject to specific use constraints, details of which may not be explicitly described. It is the responsibility of potential and actual users to be aware of such constraints and to abide by them. By making use of material from a digitised thesis, you accept these copyright and disclaimer provisions. Where it is brought to the attention of Trinity College Library that there may be a breach of copyright or other restraint, it is the policy to withdraw or take down access to a thesis while the issue is being resolved.

Access Agreement

By using a Digitised Thesis from Trinity College Library you are bound by the following Terms & Conditions. Please read them carefully.

I have read and I understand the following statement: All material supplied via a Digitised Thesis from Trinity College Library is protected by copyright and other intellectual property rights, and duplication or sale of all or part of any of a thesis is not permitted, except that material may be duplicated by you for your research use or for educational purposes in electronic or print form providing the copyright owners are acknowledged using the normal conventions. You must obtain permission for any other use. Electronic or print copies may not be offered, whether for sale or otherwise to anyone. This copy has been supplied on the understanding that it is copyright material and that no quotation from the thesis may be published without proper acknowledgement.

Authentication and Accounting for Network Services in Next-Generation Mobile Networks

Hitesh Tewari

A thesis submitted for the degree of
Doctor in Philosophy in Computer Science
University of Dublin, Trinity College
Department of Computer Science
May 6th, 2005

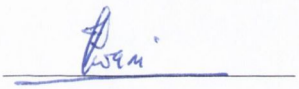
Declaration

I hereby declare that:

(a) This thesis has not been submitted as an exercise for a degree at this or any other University.

(b) This thesis is entirely the work of the author, except where otherwise stated.

(c) The Trinity College Library may lend and copy this thesis upon request.



Hitesh Tewari
May 2005

To my dearest wife Prerna for her love, support, and encouragement
throughout the writing of this thesis

and

To my parents for all the sacrifices they have made
in order to provide me with a good education.

Acknowledgements

First and foremost I wish to thank Professor John Byrne for having the faith to appoint me as a member of staff in the department which afforded me the opportunity to pursue further research avenues. Sincere thanks also due to Donal O'Mahony for introducing me to the world of computer networks and security and for supervising this thesis. Thanks also to the head of department Dr. David Abrahamson for his constant support and encouragement. Finally, I wish to thank my good friend Michael Peirce, whose original work inspired me to look further into the area of electronic payment systems and next-generation mobile networks.

Abstract

Authentication and Accounting for Network Services in Next-Generation Mobile Networks
Hitesh Tewari
Supervisor: Prof. Donal O'Mahony

Mobile communications technologies are in a constant state of flux. They have evolved from simple one-way radio communications systems, to today's third-generation networks that support digital signaling and multimedia messaging. Traditionally mobile networks have been built and operated by large telecommunications network operators on a region or country wide basis. Such operators usually have roaming agreements to allow mobile users who roam outside of their home network to seamlessly access network services in foreign networks, and rely on trust-based billing techniques to charge for network usage in order to maintain their revenue streams.

It is envisaged that the next generation of mobile networks will consist of large numbers of wireless access networks comprising heterogeneous radio access technologies. These networks will be centered on a high-speed network core with IP as the main transmission and communications protocol. With large numbers of different sized independent network operators, value-added service providers and millions of roaming mobile users, there is a need to remove the implicit trust relationships between them, in order to provide simplified authentication procedures and incontestable charging for network services. Also the emergence of mobile ad hoc networks has sparked a great deal of interest in the employing them as a flexible tool in extending the reach of the fixed networking infrastructure in next-generation networks. However ad hoc networks usually consist of closed user groups and require all the nodes in the network to cooperate in the routing of packets. The limited battery life of mobile nodes is an important factor, and users must be compensated for forwarding packets on behalf of other nodes in the network. In addition, the presence of malicious nodes can disrupt of the cooperative nature of the network and lead to transmission failures.

The issues of authentication and accounting in next-generation mobile networks are addressed by designing a micropayment scheme that allows the access network operator to be paid in real time for service provision, and provides for the simultaneous authentication of routing update messages in the network. Next the issue of compensating nodes in an ad hoc network for packet forwarding is addressed by extending the basic payment scheme to provide a flexible multi-party micropayment system for ad hoc networks. Also the problem of misbehaving nodes in ad hoc networks is explored. A secure routing and packet forwarding scheme has been developed to counteract the presence of malicious nodes in ad hoc networks which may advertise false routes or intermittently drop packets that are not destined for them to disrupt the flow of datagrams in the network. Further, a prototype of each of the above three protocols has been implemented on a well known network simulator. Finally, experimental measurements of cryptographic algorithms that were employed as part of this thesis have been benchmarked on a PDA in order to evaluate their suitability on such constraint devices.

Related Publications

Journal Papers:

H. Tewari and D. O'Mahony, "Real-Time Payments for Mobile IP", *IEEE Communications*, vol. 42, no. 2, Feb. 2003, pp. 126-136.

Books:

D. O'Mahony, M. Peirce and H. Tewari, *Electronic Payment Systems*, Artech House, Boston/London, 1997.

D. O'Mahony, M. Peirce and H. Tewari, *Electronic Payment Systems for E-Commerce*, 2nd Ed., Artech House, Boston/London, 2001.

Refereed Conferences:

H. Tewari and D. O'Mahony, "Lightweight AAA for Cellular IP", *Proceedings of European Wireless '02*, Florence, Italy, Feb. 2002, pp. 301-306.

H. Tewari and D. O'Mahony, "Multiparty Micropayments for Ad Hoc Networks", *Proceedings of IEEE Wireless Communications and Networking Conference* (*WCNC*), New Orleans, Louisiana, USA, March 2003.

P. Argyroudis, R. Verma, H. Tewari and D. O'Mahony, "Performance Analysis of Cryptographic Protocols on Handheld Devices", *Proceedings of the 3rd IEEE International Symposium on Network Computing and Applications*, Cambridge, MA, Aug. 30 – Sep. 1, 2004.

URL: http://www.cs.tcd.ie/~htewari/publications.html

Contents

List of Figures

List of Tables

Glossary

3GPP	Third Generation Partnership Project
3GGP2	Third Generation Partnership Project 2
AAA	Authentication, Authorization and Accounting
AAAB	AAA Server Operated by a Broker
AAAH	AAA Server in the Home Network
AAAF	AAA Server in the Foreign Network
ACH	Automated Clearing House
ACK	Acknowledgement
AES	Advance Encryption Standard
AH	Authentication Header
AODV	Ad Hoc On-Demand Distance Vector Routing
AP	Access Point
ARAN	Authenticated Routing for Ad Hoc Networks
AuC	Authentication Center
AuthCache	Authentication Cache
B3G	Beyond-3G
BK	Broker
BS	Billing System
BSC	Base Station Controller
BTS	Base Transceiver System
CA	Certification Authority
CAFE	Conditional Access for Europe
CBR	Constant Bit Rate
CCoA	Colocated Care-of Address
CDR	Call Detail Record
CDMA	Code Division Multiple Access
CEPS	Common Electronic Purse Specification
CG	Charging Gateway
CGF	Charging Gateway Functionality
CHAP	Challenge Handshake Authentication Protocol
CIP	Cellular IP
CMIS	Columbia IP Micromobility Software
CN	Correspondent Node
CoA	Care-of Address
CONFIDANT	Cooperation of Nodes – Fairness in Dynamic Ad-hoc NeTworks
CRL	Certificate Revocation List
DARPA	Defense Advanced Research Projects Agency
D-AMPS	Digital Advanced Mobile Phone System
DCF	Distributed Coordination Function
DES	Data Encryption Standard
DHCP	Dynamic Host Configuration Protocol
Diameter	AAA Protocol
DNS	Domain Name System
DoS	Denial-of-Service
DSA	Digital Signature Algorithm
DSDV	Destination-Sequenced Distance-Vector

DSR	Dynamic Source Routing
DSS	Digital Signature Standard
DTMF	Dual Tone Multi-Frequency
EAP	Extensible Authentication Protocol
ESP	Encapsulating Security Payload
ETSI	European Telecommunications Standards Institute
FA	Foreign Agent
FCC	Federal Communications Commission
FES	Fixed Earth Station
FN	Foreign Network
FSTC	Financial Services Technology Consortium
FTP	File Transfer Protocol
FV	First Virtual
GFA	Gateway Foreign Agent
GGSN	Gateway GPRS Support Node
GMSC	Gateway Mobile Switching Center
GPRS	General Packet Radio Service
GSM	Global System for Mobile Communications
GSMA	GSM Association
GSN	GPRS Support Node
GTP	GPRS Tunneling Protocol
GW	Gateway Node
HA	Home Agent
HAWAII	Handoff-Aware Wireless Access Internet Infrastructure
HLR	Home Location Register
HN	Home Network
HPLMN	Home PLMN
HRPD	High Rate Packet Data
HSCSD	High Speed Circuit-Switched Data
HSO	Hot Spot Operator
HTTP	Hyper Text Transfer Protocol
IDMP	Intradomain Mobility Management Protocol
IEC	Inter-Exchange Carrier
IEEE	Institute of Electrical and Electronic Engineers
IETF	Internet Engineering Task Force
IKE	Internet Key Exchange
IMSI	International Mobile Subscriber Identity
IP	Internet Protocol
IPsec	IP Security Protocol
ISKMP	Used in IPsec Key Exchange
ISP	Internet Service Provider
ISM	Industrial Scientific and Medical
IV	Initialization Vector
LCoA	Local Care-of-Address
LEC	Local Exchange Carrier
LS	Location Management Server
MA	Mobility Agent
MAC	Message Authentication Code

MANET	Mobile Ad Hoc Network
MD5	Message Digest Algorithm – Developed by Ronald Rivest
MIP	Mobile IP
MN	Mobile Node
MOTO	Mail Order/Telephone Order
MoU	Memorandum of Understanding
MSC	Mobile Switching Center
NAI	Network Access Identifier
NAS	Network Access Server
NIC	Network Interface Card
NO	Network Operator
NS	Network Simulator *ns-2*
OAKLEY	Used in IPsec Key Exchange
OFDM	Orthogonal Frequency Division Multiplexing
OLSR	Optimized Link State Routing
OSSL	Open SSL (Secure Socket Library)
PAP	Password Authentication Protocol
PCF	Point Control Function
PDA	Personal Digital Assistant
PDC	Personal Digital Cellular
PDN	Packet Data Network
PDP	Packet Data Protocol
PDR	Packet Delivery Ratio
PDSN	Packet Data Serving Node
PIN	Personal Identification Number
PKI	Public Key Infrastructure
PLMN	Public Land Mobile Network
POS	Point-Of-Sale
POTS	Plain Old Telephone Service
PRAT	Path Rating Agent
PSC	Prepaid Service Center
PSTN	Public Switched Telephone Network
QoS	Quality of Service
RADIUS	Remote Dial In User Service
RAN	Radio Access Network
RC4	Stream Cipher – Developed by Ronal Rivest
RNC	Radio Network Controller
RERR	Route Error
RREP	Route Reply
RREQ	Route Request
RSA	Public-Key Cryptosystem – Developed by Rivest, Shamir and Adleman
RTP	Real-time Transport Protocol
RTSP	Real Time Streaming Protocol
SA	Security Association
SAODV	Secure Ad Hoc On-Demand Distance Vector Routing
SCP	Secure Charging Protocol
SEAD	Secure Efficient Ad Hoc Distance vector
SGSN	Serving GPRS Support Node

SET	Secure Electronic Transactions
SIM	Subscriber Identity Module
SIP	Session Initiation Protocol
SMS	Short Messaging Service
SSL	Secure Sockets Layer
SWIFT	Society for Worldwide Interbank Financial Telecommunications
TADIG	Transfer Account Data Interchange Group
TAP	Transferred Account Procedure
TC	Topology Control
TCL	Tool Command Language
TCP	Transmission Control Protocol
TDMA	Time Division Multiple Access
TKIP	Temporal Key Integrity Protocol
TLS	Transport Layer Security
TTP	Trusted Third Party
TIMIP	Terminal Independent Mobility for IP
UDP	User Datagram Protocol
UMTS	Universal Mobile Telecommunications System
UOBT	Unbalanced One-way Binary Tree
USIM	UMTS Subscriber Identity Module
UTRAN	UMTS Terrestrial Radio Access Network
VASP	Value Added Service Provider
VINT	Virtual InterNetwork Testbed
VLR	Visitor Location Register
VPLMN	Visited PLMN
VPN	Virtual Private Network
WAP	Wireless Application Protocol
WCDMA	Wideband Code Division Multiple Access
WEP	Wired Equivalent Protocol
Wi-Fi	Wireless Fidelity
WIM	Wireless Identity Module
WinCE	Windows CE
WISP	Wireless Internet Service Provider
WLAN	Wireless Local Area Network
WPAN	Wireless Personal Area Network
WTLS	Wireless Transport Layer Security

1 Introduction

"If I have seen further than other men, it is by standing on the shoulders of giants."
Sir Isaac Newton (1642-1727)

1.1 Mobile Communications

Mobile radio communications systems have been successfully deployed since 1980 in different regions of the world to provide basic telephony services to mobile users. The last decade, in particular, has witnessed explosive growth, with the number of mobile phone users worldwide predicted to exceed 2 billion by the year 2010 [UMTS03]. Increasingly, mobile devices are being used for data services such as multimedia messaging and mobile Internet access [Jam03]. Statistics from the GSM Association show that in the fourth-quarter of 2004 there were 626 GSM network operators in 198 countries, providing coverage to more than 1.26 billion subscribers [GSMA04a]. At the present time the number of mobile networks worldwide is relatively small in number. These networks are independently managed and usually cover a large region or country.

Typically mobile users associate themselves with a single Network Operator (NO) who charges them for the provisioning of mobile communications services in their home network. In order to allow users to access network services while roaming outside of the home network, their Home Network (HN) operator may have extensive bilateral roaming agreements for accounting and billing purposes with other Foreign Network (FN) operators. There are predominantly two types of charging strategies in current mobile networks. The first is where a user subscribes to a credit-based or *postpaid* account and is billed by his home network operator periodically. This includes any charges incurred by a user for services used while roaming in distant networks. According to the GSM Association there are more than 20,000 individual roaming agreements or Memorandums of Understanding (MoUs) in place between GSM operators [GSMA]. Thus behind the simple objective of global roaming lies a complex process that gathers detailed information about each call in a Call Detail Record (CDR), and periodically bulk transfers the CDRs to a centralized billing system in the home network [PO99]. If the records are transferred immediately from the Mobile Switching Center (MSC) the process is known as *hot billing*, which allows bills to be processed on request or within a given time limit.

The other popular method for charging is whereby a user purchases *prepaid* credit from his network operator and his account is decremented in real time for the provision of network services [LCR00]. However, there are usually restrictions on the roaming facilities provided for such prepaying users. One of the reasons is that sometimes prepaid charging cannot be performed at the FN, as the home and visited networks may exercise different prepaid service solutions that are incompatible. A major disadvantage from user's point of view is that most networks require the visited MSC to route the prepaid call back through the HN. Prepaid calls are thus charged more than postpaid calls and make prepaid solutions too expensive for roaming for most users. As part of this thesis a solution which eliminates the need for a mobile user to associate themselves with a *home network* operator for the purposes of authentication and accounting is proposed. The solution further

enables payment to any network operator in real time for the provision of network services without the need for long-term contractual commitments between any of the entities in the system.

Today mobile communications have become an essential part of modern everyday life. In the future, with the advent of ubiquitous mobile communications, it will become transparent to the end users, much like the medium within which it operates. Ubiquitous communication will change the present model which is governed by communications between people, to communication between anything and everything. In such an environment, mobile users will have continuous access to network services via wireless devices, which may use *software radio* techniques to adapt themselves to the appropriate networking technology [CPP99, MRPS04]. It is also envisaged that in the future there will be large numbers of public and private network operators, who will provide access to voice and data services in existing fixed networks such as the PSTN and Internet [WAGZ+03]. These Radio Access Networks (RANs) will range in size from private home and office wireless networks, to public and private Wireless Local Area Networks (WLANs) in densely populated urban areas, to wide-area suburban public cellular networks, and global satellite networks as depicted in Figure 1-1.

Users will be able to select connectivity to applications and services using devices and access technologies that best suit their needs for each particular communications session. Factors such as the Quality of Service (QoS), connection speed, and costs will play a major role in determining the choice of provider. Roaming between independently operated networks will become an essential part of everyday life. A typical example will be users moving between their home and work environments via a number of intermediate wide-area networks. In addition to network operators there will be a large number of Value Added Service Providers (VASPs). These entities will provide services such as hosting location management information for Mobile Nodes (MNs), providing street maps, stock quotes and a host of other services.

Each of these network operators and service providers will require payment for services that they may provide to roaming mobile users. With potentially many thousands of NOs and VASPs and a billion plus mobile users with multiple devices, it is essential to simplify the billing and payment processes. It is also imperative to remove any complex trust relationships that may be required between these entities for the purposes of authentication and accounting for network usage. Secure payment for network usage and service provision is one of the main concerns of this thesis.

1.1.1 Mobile Network Generations

First-generation mobile networks were implemented based around analog technologies and provided basic telephony services to end-users. In contrast, second-generation (2G) cellular networks such as the Global System for Mobile Communications (GSM) in Europe, IS-54/IS-95 in the US, and Personal Digital Cellular (PDC) in Japan make use of digital transmission technologies [ZAB99]. 2G systems such as GSM were mainly designed for circuit-switched voice and low-bit-rate data services at 9.6 or 14.4Kbps [VLLX02, RWO95]. Support for High Speed Circuit-Switched Data (HSCSD) has recently been added which allows for the concatenation of multiple time slots and data rates up to 64Kbps in GSM networks. There are a number of entities that are present in one form or another in the various 2G systems, such as base stations, location registers and switches. The Home Location Register (HLR) is one such entity that contains the identity and user data of all subscribers that belong to a NO. The Visitor Location Register (VLR) is another database that contains data of all MNs currently located in a serving MSC.

Studies on the next generation of mobile telecommunications systems, known as the third-generation (3G), began in the mid 1980s with a view to deploying commercial networks by early 2000. However this schedule has been delayed by many technical and commercial difficulties [Gar02]. As an interim solution, GSM

operators in Europe have been moving towards *3G-like* services in an effort to generate new revenue streams. In Europe, the General Packet Radio Service (GPRS), also referred to as 2.5G, provides a packet-switched radio service that enables an *always-on* connection and an average data rate of 115Kbps [DGA01]. GPRS employs packet transmission technology in the core network while making use of existing GSM radio interfaces in the RAN. GPRS adds two new network elements to the existing GSM infrastructure, namely the Serving GPRS Support Node (SGSN) and the Gateway GPRS Support Node (GGSN) in the core network. The SGSN is responsible for the delivery of packets to the correct Radio Network Controller (RNC), while the GGSN is like an IP gateway, and acts as a logical interface to external data networks such as the Internet [CG97].

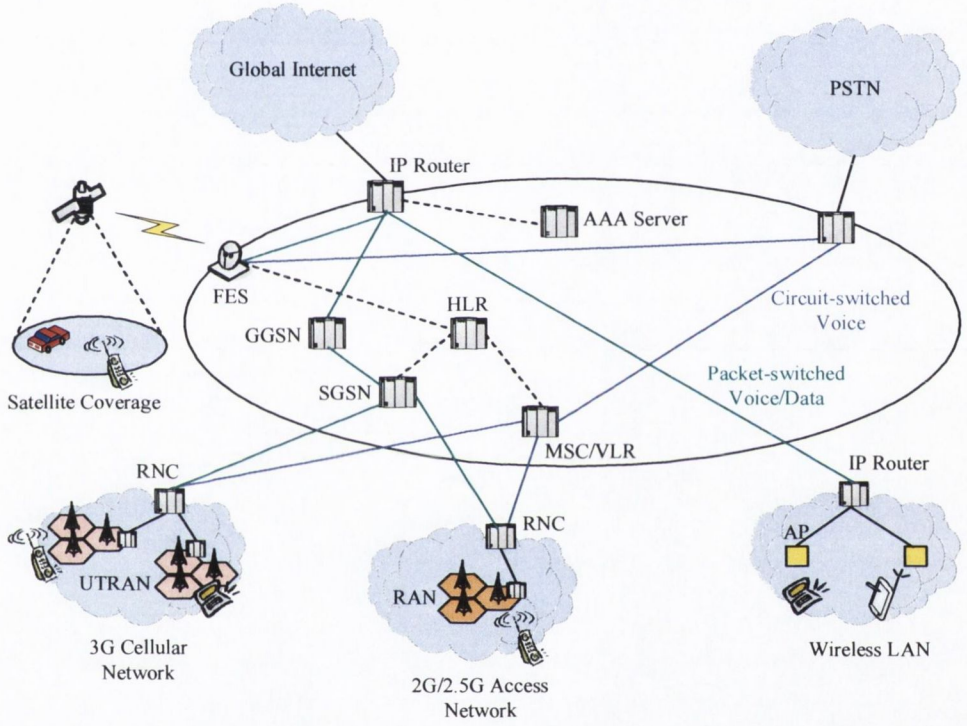


Figure 1-1 Architectural View of Present and Emerging Mobile Networks

3G systems on the other hand incorporate greatly enhanced air interfaces to support wider bandwidths for improved capacity and higher data rates. The 3G standards support a wide range of bearer services from voice to high-rate data services of 144Kbps for vehicular, 384Kbps for pedestrian, and up to 2Mbps for in-door environments. European efforts in 3G, referred to as the Universal Mobile Telecommunications System (UMTS), have been developed under the auspices of the Third Generation Partnership Project (3GPP). UMTS makes use of the Wideband Code Division Multiple Access (WCDMA) standard in the UMTS Radio Access Network (UTRAN) [3GPP].

The UMTS core network consists of circuit- and packet-switched domains and makes use of evolved GSM/GPRS core network entities such as the MSC, SGSN and GGSN for access to the PSTN and Internet. In North America, the Third Generation Partnership Project 2 (3GPP2) has developed the cdma2000 system [3GPP2]. The core network is based on an evolved ANSI-41 network, while the RAN is based on cdmaOne

[Sar00, PK01]. 3GPP2 however has taken advantage of the existing IETF Mobile IP protocol [Per02a, Per02b] to enhance the network architecture in order to provide IP capabilities.

Even as 3G networks are being deployed worldwide, focus has already shifted to the next generation of mobile networks. These networks have been referred to by a variety of names such as Beyond-3G (B3G) systems [Usk03, KJCH+03, LJP03], fourth-generation (4G) networks [BGQS+01, MAGM+03], and all-IP wireless networks [LPHC02, BL01]. These terms often mean different things to different people [MRPS04]. For instance, the 3GPP and 3GPP2 camps have contrasting strategies on the evolution of current 3G networks [PD00, RHD03]. However, there is one common theme that transcends all the proposed architectures, and it is that the Internet Protocol (IP) will play a crucial role in the development of any such future networks [HY03, BPIM+01].

The proposed architectures for IP-based next-generation networks range in complexity and design. At a very minimum these systems may contain evolved network elements to support packet transmission, along with a gateway IP-router between the telecom operators 3G network and the global Internet. In some proposals the Mobile IP protocol plays a major role in the mobility management, in relation to terminal mobility and routing of packets in the network. Other architectures propose IP transport in the radio access network [VVV03], which aims to simplify application development, as well as reduce overall network development and transmission costs.

The proponents of next-generation networks also see WLAN *hotspots* as complementing existing 3G networks in providing high-speed IP access of up to 100Mbps in local-area environments [BCHL+03, AHP03, DTNA+03]. Some of the proposals make use of the IETF Authentication, Authorization and Accounting (AAA) infrastructure [Per00, AAA] in conjunction with the existing 2.5/3G authentication mechanisms for terminal and user authentication, access control and billing [HMT02]. Finally, some of the more avant-garde proposals advocate a core network based around IP with a variety of radio access network technologies at the edges [New04]. They promote end-to-end transfer of IP packets with seamless roaming between 2G, 3G, WLAN and fixed networks [ZVTZ+02, VLLX02, BBT02, PCAG+04].

A crucial consideration that has however not been adequately addressed by the majority of researchers, is the complex issue of settlement of payments in a ubiquitous mobile communications scenario. In such a scenario for example, anyone with a Wi-Fi Access Point (AP) and high-speed connection to the Internet has the potential to become a picocell operator [WiFi]. With large numbers of network operators and service providers, it will become a near impossible task to establish bilateral agreements with all other operators. Even in situations where agreements are in place, the issue of operator trust in generating correct CDRs for service provision will be of vital importance. With so many NOs and roaming mobile users, the system will be open to operator fraud and user abuse. Thus billing based on preestablished trust relationships will no longer be adequate or scalable.

1.2 Thesis Motivation

Our participation in a third-generation European ACTS project COBUCO [COBU96, COBU98], led us (very early on) to believe that third-generation networks would quickly evolve into or be superseded by *all-IP* 4G or next-generation networks [4GT]. As an alternative to retrofitting existing 2G networks with IP capabilities, it seemed much more beneficial to evolve the existing Internet infrastructure into a global mobile telecommunications network. Though the Internet was primarily designed as a universal data communications network, work has been ongoing to enable seamless voice, video and data operations over a single infrastructure [Met00]. To this effect, protocols have been under development by the IETF to address QoS [Wro97, BBCD+99], multicast group membership [MAGMA], and fast label switching [MPLS] issues.

Work has also been underway to introduce mobility into the Internet to enable nodes to seamlessly roam among IP subnetworks [MIP]. Traditionally in IP networks each node is assigned a unique IP address. In normal IP routing, packets are routed from a source to a destination on a hop-by-hop basis, where a router makes a routing decision based on the network portion of the destination address. Thus an IP address is used for routing and also specifies a point of attachment for a node on the Internet. The Mobile IP protocol allows a MN to change its point of attachment to the Internet, and for such a change to be completely transparent to applications in that no IP address changes are needed to allow mobility. A MN can receive datagrams using its home address regardless of its actual position on the Internet [Per98]. Mobile IP solves the *macromobility* problem in wide-area networks by updating the Care-of Address (CoA) at the Home Agent (HA), whenever a mobile node changes its point of attachment on the Internet.

Mobile IP was designed with a view to supporting MNs with relatively slow mobility patterns, and is not adequate for situations which require fast handoffs. With Mobile IP, any change in a MNs point of attachment results in signaling messages being sent to the HA in the home network, even though most of the path between the MN and the HA may remain unchanged. In response to this, *micromobility* architectures have been developed to support mobility within a local network or domain [CGKV+00, CKK02, SMMC04]. The main aim of these protocols is to minimize the number of signaling messages that have to be carried over the core network to support host mobility. With micromobility protocols, local handoffs result in signaling messages being generated which are limited to the administrative domain in which the MN is currently roaming. As long as a MN remains within the administrative domain of a single operator, there is no need to update its route entry in the home network.

However to transform the Internet into a viable telecommunications network, there is still one key element missing, and that is the lack of an efficient, scalable authentication and accounting structure. The Internet initially started life as a research network, and comprises numerous interconnected public and private subnetworks which are independently administered. In the majority of cases, these subnetworks are operated as closed user groups with pre-established trust relationships between the users and the network operator. The current approach to providing secure and trusted communications in IP networks is based on the IETF AAA protocols [RWRS00, CLGZ+03]. Users have an enduring relationship with a network operator, and inter-operator roaming agreements are required to allow mobile users to access resources in third-party networks. Payment for usage of network resources may take place after the services are used and CDRs must be exchanged by the operators for billing purposes. As noted earlier, with large numbers of users and operators, issues of trust and scalability become of crucial importance.

1.2.1 Next-Generation Network Scenario

Figure 1-2 depicts a vision of an all-IP next-generation networking architecture. It envisages that in the future there will be a large number of independent network operators and service providers, who will provide users with wireless access to other fixed or mobile nodes. These access networks will have dedicated high-speed connections via a Gateway router (GW) into the core IP network. Small and medium sized networks may consist of a number of radio cells, and may support micromobility protocols for fast handover within the access network. Each user in the system will be uniquely identifiable by his Network Access Identifier (NAI) which is of the form *user@realm* [Abo99]. A user who arrives in a new access network can immediately start utilizing services in the core network, once he has registered himself with the NO and bought provider specific payment tokens from a trusted Broker (BK). In order to be able to receive incoming calls, a user must register his care-of address with his preferred Location Management Server (LS). The LS keeps a mapping of the user's NAI to his current CoA.

Network operators generate revenue by charging for usage of network resources by roaming mobiles and may also provide other value-added services. Existing credit based mobile billing systems trust users to pay their bills based on strong identity verification and credit history checks. They also place total trust in the operator to bill the user for the correct amount, but provide no mechanisms to prove the authenticity of the generated CDRs. Unlimited credit with postfact punishment is too open to abuse in mobile networks. With a large population of mobile users and independent network operators, it is desirable to remove the need to trust them and thereby minimize fraud in the system. In this thesis micropayment technology [OPT01] is employed, which allows the routers within the access network to authenticate datagrams prior to making a routing decision, and the NO to be paid in real time for service provision. As long as the MN releases the correct payment tokens, the operator provides him with the requested service. This eliminates the need for billing and any long-lived accounting or trust relationship between the MN and the NO.

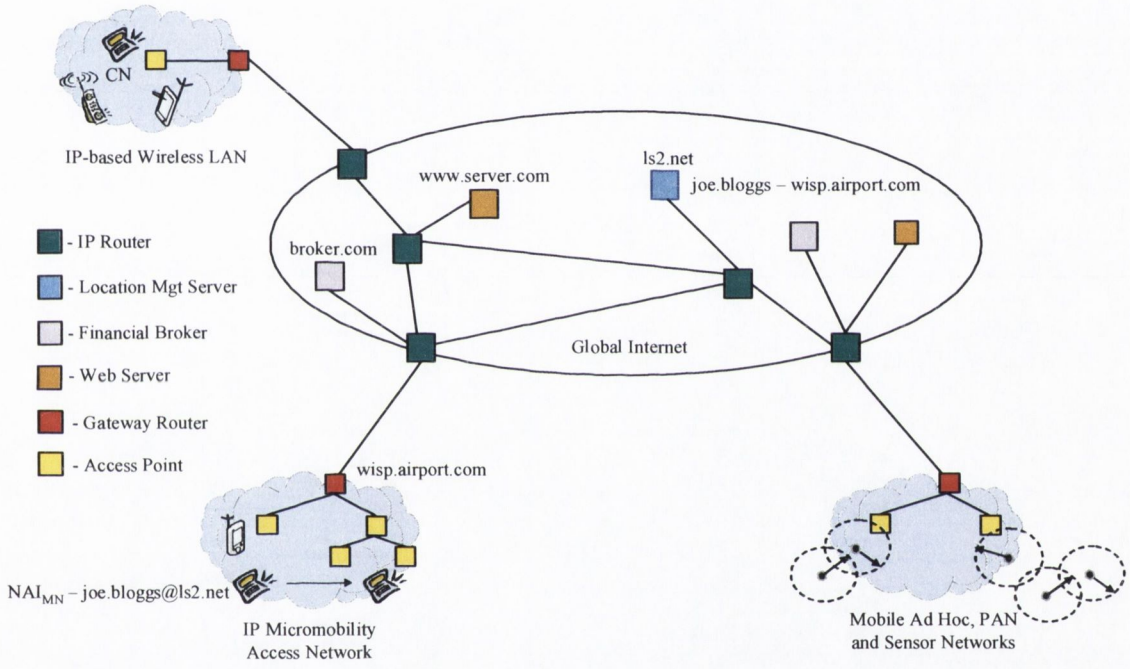


Figure 1-2 All-IP Next-Generation Network Architecture Proposal

In recent years ad hoc networks have gained acceptance as a mainstream networking technology. It is envisaged that ad hoc networks will play an integral part in providing a fast and flexible mechanism to extend the reach of WLAN hotspots and the Internet in next-generation networks. An ad hoc network is an infrastructureless network in which a group of nodes communicate with each other using multi-hop wireless links, and where the topology of the network can change unpredictably. Individual nodes are responsible for dynamically discovering which other nodes they can directly communicate with. Each node also acts as a router to forward packets for other nodes within the network [RR02]. Nodes in such networks may have affiliations with different NOs or could be under the administrative control of individual users. These users may not necessarily wish to cooperate in relaying packets on behalf of other users in the network.

One of the major concerns of such users will be to preserve the limited battery life of their mobile devices, and the relaying of packets directly impacts on this. Therefore, mechanisms have to be in place to compensate the nodes involved in the relaying process, and for any value-added services that they may

provide. However nodes in an ad hoc network do not have the luxury of long-lived trust relationships as is the case in the wired Internet. Also, it is not always possible to contact a Trusted Third Party (TTP) to verify user identities or payment tokens that may be presented to a node. In the latter case, the cost of contacting a TTP to verify the payment instrument may outweigh the actual benefits gained. Once again in this thesis a micropayment scheme is employed, which enables a node to join an existing ad hoc network, and allows it to pay each node that relays packets on its behalf in real time.

Current ad hoc routing protocols [RT99] implicitly trust all the participants in a network to cooperate in the forwarding of datagrams, and are not able to cope with network disruptions due to malicious behavior. For instance, an attacker can masquerade as a legitimate node and advertise false routes in the network. More sophisticated attacks can be mounted whereby a node selectively drops datagrams and points to collisions on the transmission medium as the probable cause. As part of this thesis, mechanisms have been developed to monitor such activity in order to be able to identify nodes that deliberately misbehave, and subsequently blacklist them from the network.

1.3 Scope of the Research

The focus of this research is to develop a lightweight authentication and accounting protocol for all-IP next-generation mobile communications networks. As outlined in Section 1.1, it is widely assumed that such next-generation mobile networks will comprise wireless infrastructure networks (where a MN communicates directly with a base station) as well as mobile ad hoc networks (where a node may relay packets through a number of intermediate mobile nodes prior to their reaching their destination). With this in mind, Section 1.2 proposed an integrated network architecture for such a next-generation mobile network scenario. In each case the NO or mobile nodes need to be paid for any services rendered. For the first scenario, a micropayment scheme for Mobile IP based networks has been developed which allows small and medium sized, independent NOs to be paid in real-time for service provision, i.e. the NO can be assured of payment by verifying the cryptographic tokens sent along with the packets as they travel through its operator's access network. However from time-to-time a MN may also find itself outside the coverage of a fixed network access point or base station. In such situations the MN may get its packets relayed through a number of intermediate nodes. Again each node needs to be remunerated for the relaying operation. The initial payment protocol has been extended into a multi-party micropayment scheme for ad hoc networks to allow each intermediate node to be paid independently without the need to contact a TTP. Finally, to address the issue of malicious or rogue nodes in an ad hoc network scenario which may drop packets to disrupt the operation of the network, a secure route discovery and packet forwarding scheme was developed. This in conjunction with the multi-party micropayment scheme facilitates guaranteed payment for service provision in an ad hoc network. The three protocols combined together allow for an overall secure authentication and accounting infrastructure for next-generation mobile networks.

1.3.1 Outline of the Dissertation

The thesis structure is as follows. In Chapter 2, the mobility management, authentication and billing procedures currently being proposed for third- and next-generation networks are examined in detail. In situations where mobile nodes exhibit high mobility patterns, the Mobile IP protocol is not suitable due to its high signaling overheads. An overview of some of the micromobility protocols that have been proposed is presented. Signaling messages in such networks are required to be authenticated prior to any updates of route entries in routers within the network. With this in mind, the current AAA proposals for IP networks are studied, and their inefficiencies and inadequacies for micromobility environments are highlighted. The issues of route discovery and packet forwarding in ad hoc networks are also examined. The twin problems of authentication and accounting for IP networks need to be clearly understood and scrutinized before effective alternative solutions can be designed as part of this thesis.

This thesis proposes the use of electronic tokens to pay for mobile services rather than using traditional billing methods. Chapter 3 gives an overview of current electronic payment research. The majority of electronic payment solutions mimic real-world payment instruments such as cash, cheques and payment cards. These types of payment solutions aim to provide high-value transactions known as *macropayments* and usually require verification or authorization from a TTP in the network. However these types of solutions are not suitable for repeated small-value transactions of a cent or less, as the cost of contacting the TTP far outweighs the value of the payment instrument. In the scenario for next-generation networks, the aim is to make use of *micropayments* based on *hash chains* to pay the network operator or connectivity provider on a per packet basis. A detailed review of a number of efficient hash chain schemes is performed, in order to understand the various techniques which can enable repeated payments from mobile devices.

In Chapter 4, a proposal for a real-time authentication and accounting scheme for Mobile IP based next-generation networks is presented. The current inadequacies of the existing AAA schemes are highlighted, and the requirements for an efficient and lightweight approach to the problem are identified. The details of the protocol are presented and its operation in a next-generation network environment is explained. The protocol was implemented using a version of the Cellular IP micromobility protocol [CIP] that was developed using the *ns-2* simulator [NS]. The computation and communications costs of the implemented protocol are also analyzed.

Chapter 5 extends the concept of the real-time micropayments scheme to the domain of ad hoc networks. Ad hoc networks present their own unique challenges for authentication and accounting due to the lack of any central authority. In addition, there is no guarantee that a node has access to an online TTP all the time. These issues are addressed by presenting a multi-party micropayment solution that allows each node in the path between the source and destination to be paid in real time for packet forwarding in an ad hoc network. An evaluation using the *ns-2* protocol simulator is presented, along with an informal analysis of the risk assessment and cryptographic processing overheads associated with the protocol.

As part of developing the payment solution for ad hoc networks, it was realized that for it to be effective, a secure route discovery and packet forwarding scheme was required. Hash chains in conjunction with public key cryptography are used to develop a secure routing and packet forwarding scheme for ad hoc networks in Chapter 6. The DSR protocol [JMH03] implementation in *ns-2* was extended to incorporate cryptographic functions to implement the secure routing scheme.

In Chapter 7, a performance evaluation of a number of important cryptographic protocols is carried out on a Personal Digital Assistant (PDA) running the WinCE Pocket PC 2002 operating system. Both symmetric and public-key protocols implemented using the OpenSSL cryptographic library [OSSL] are benchmarked. In addition, the bulk hash chain generation schemes that were used to implement the micropayment and authentication schemes in this thesis are also evaluated. This study highlights the efficiency of hash chain techniques when compared to other cryptographic techniques such as symmetric and public-key ciphers.

Finally in Chapter 8, the contributions of the thesis are summarized and a discussion of possible future work is given. Initial results of this thesis have already been presented at the European Wireless '02 [TO02], IEEE Communications Journal [TO03a], the IEEE Wireless Communications and Networking Conference '03 [TO03b], and the IEEE International Symposium on Network Computing and Applications [AVTO04].

2 Management of Next-Generation Mobile Networks

"Radio has no future."
Lord Kelvin, 1897, on Marconi's experiments

2.1 Introduction

Next-generation wireless networks and services will be much more complex than today's second-generation (2G) systems. Universal access, global roaming and multimedia services will be the key driving factors that will have a profound impact on how these networks will be managed [UK03]. Work has been ongoing over the past decade on third-generation (3G) cellular networks that will achieve global roaming, and seamless voice and data services at speeds of 384Kbps and above. However the roll-out of 3G networks has been somewhat delayed, mainly due to the very high costs associated with licensing the spectrum and building the networks, and problems associated with the handsets. From the end-users perspective, the lack of a clear set of killer applications and expensive tariffs has resulted in a slow uptake of the technology. Therefore in Europe and the United States, network operators have deployed or are in the process of deploying 2.5G systems, such as the GSM-based GPRS and cdmaOne-based cdma2000 1xRTT networks, which offer data rates up to 144Kbps [EHHK+01, HPNL02].

Even though 3G mobile communications systems have barely had time to be deployed, both industry and academia have already started looking beyond them [ZHA04]. The architecture, protocols, services and wireless technologies that will constitute next-generation networks are still under consideration and a subject of great debate. Network operators today are faced with a confusing array of technologies on how to best build next-generation networks. Each technology in turn has its own particular advantages and disadvantages. For example, the IETF Mobile IP protocol [Per02b] represents a simple and scalable global mobility solution for IP-based networks, but lacks support for fast handoff control, real-time location tracking and authentication. In contrast, current 2.5/3G systems offer support for seamless mobility, paging and QoS, but are built on complex connection-oriented networking infrastructure that lacks the inherent flexibility, scalability, and cost effectiveness found in packet-switched networks [BCJ00]. Next-generation networks are referred to by a number of names in the literature. The term fourth-generation (4G) is used in this thesis to refer to the next generation of networks targeting an *all-IP* solution [JL03, JT01, MP01, ZK03].

One repeated theme across many of the proposed architectures is the vision that 4G is not really a new air interface, but instead will consist of heterogeneous wireless access networks with IP technology in the core, and will provide better end-to-end IP services. The bandwidth of current 2.5G and evolving 3G technologies promises burst rates up to 384Kbps and 2Mbps respectively. However the average throughput per user is not expected to be more than 171Kbps during busy periods. In contrast, Wireless LANs (WLANs) already offer

bit rates of up to 54Mbps, with bit rates in excess of 100Mbps planned for the future [Var03]. WLANs are being deployed not only in residential and small offices, but also on a larger scale in public areas such as airports, shopping malls and neighborhoods [Boingo]. This picture is further complicated by the arrival of two fundamentally different network technologies, namely *ad hoc* and *wireless sensor networks*. These two technologies will also play a major role in next-generation pervasive networks. An ad hoc network is a collection of wireless nodes that wish to communicate with each other but have no fixed infrastructure available. The network is autonomously formed among many nodes with varying functionalities and power levels, which also act as routers to forward packets on behalf of other nodes in the network [GL02]. Wireless sensor networks consists of large numbers of sensors which are tiny, low-cost, low power radio devices, dedicated to performing certain functions such as collecting environmental data. Many of the ad hoc network techniques will also be applicable to sensor networks.

In addition to sophisticated mobility management schemes, 4G networks will require scalable Authentication, Authorization and Accounting (AAA) procedures to be put in place [Met99, PCAG+04]. Such procedures will help in limiting the credit-risk exposure for the involved parties. With large numbers of roaming mobile users, there is a need to efficiently authenticate user credentials and remunerate network operators and value-added providers for service provision. At the present time users must place total trust in the accuracy of the usage records and the bills generated by the Network Operator (NO). Next-generation networks will consist of large numbers of independent network operators and the above processes will become subject to abuse. In particular, postfact billing with unlimited user credit has the potential for widespread abuse in next-generation mobile networks. This provides ample motivation and justification for simplifying the overall AAA process. Finally, traditional models of billing based on *flat-rate* pricing models combined with time dependent charging will not be sufficient for the provision of data services in next-generation networks, as they do not enable measurement that is based on the real value of the services offered [CISCO01, KKAM+04].

In the remainder of this chapter the mobility management, authorization and accounting techniques used in current cellular and next-generation networks are examined in detail. The rationale for doing this is to highlight the complex nature of the resulting network architecture if the present strategies for evolving current cellular networks into next-generation all-IP networks are adopted. Section 2.2 describes the current mobility procedures employed in 3G networks. Section 2.3 gives an overview of the billing techniques used in both fixed and mobile telecommunications networks. Section 2.4 takes an in-depth look at the various Internet technologies that will play a major role in the development of 4G networks. The Mobile IP protocol and the various micromobility proposals which have been designed to address some of its shortcomings are examined in detail [CGKW+02]. The inefficiencies and scalability problems of current AAA proposals for IP networks are also analyzed. The mobility and security issues facing two new wireless technologies, namely WLANs and ad hoc networks are then outlined. In Section 2.5, a number of strategies being pursued by the telecom operators to integrate their 3G networks with these emerging network technologies are examined. The chapter concludes by highlighting the problems that remain unaddressed. Prior to proceeding with reading the details of the main body of this chapter, it is recommended that the reader turn to Appendix A, which provides details of the various cryptographic terms, notation and operators used throughout the remainder of this thesis.

2.2 Third-Generation Mobile Networks

Mobile communications networks around the world are now in the process of evolving to the third generation (3G). The third generation as its name suggests is the successor to second-generation (2G) digital, circuit-switched cellular mobile networks, such as the Global System for Mobile Communications (GSM) in Europe, the Digital Advanced Mobile Phone System (D-AMPS) in the United States, and Personal Digital Cellular

(PDC) in Japan [Tan03]. 2G networks in turn were successors to first-generation (1G) analog mobile systems such AMPS in the US. 1G and 2G networks were primarily designed to carry voice traffic, although 2G's design also includes the low bit-rate Short Messaging Service (SMS).

3G networks on the other hand provide higher capacity and enhanced network functionality which allows for advanced data and multimedia capabilities. 3G networks are expected to provide minimum data rates of 2Mbps in indoor environments and 144Kbps in outdoor environments. They provide high-quality voice transmission with improved spectrum efficiency and packet-switched services such as wireless Internet access and real-time video. In addition, the vision of 3G is to provide an *always-on* service with global roaming between different operational environments using a single identity, and to provide the user with the same set of capabilities regardless of where they are in the network.

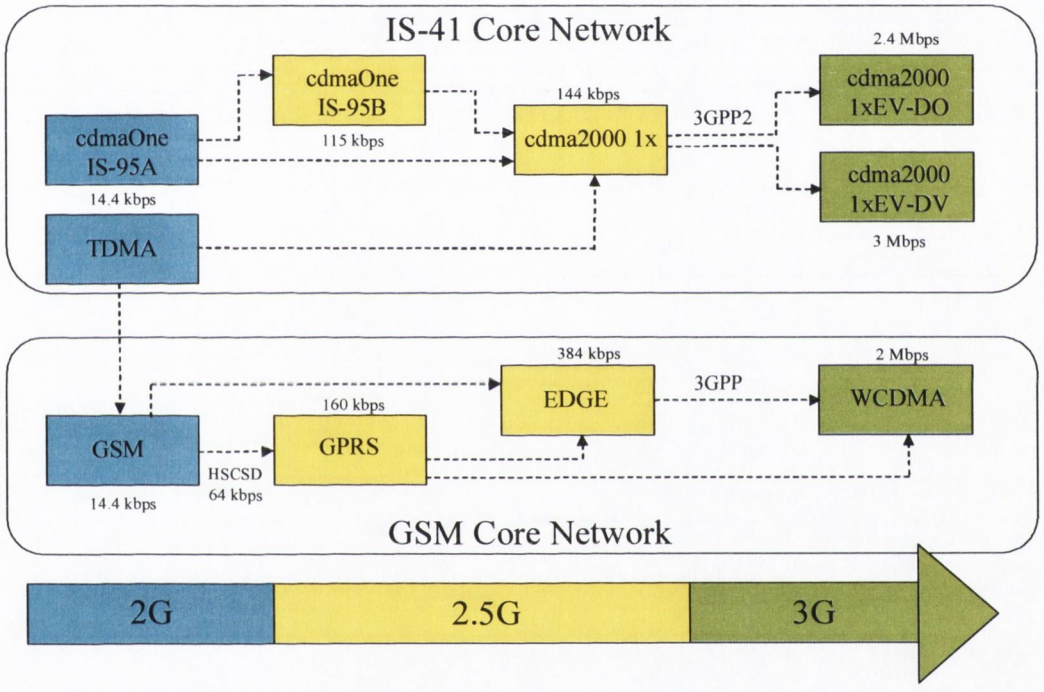


Figure 2-1 Third Generation Evolution Paths

Third-generation technologies were initially expected to provide a global access capability with a unified single radio interface and an advanced core network. However, operators are instead *evolving* their existing mobile networks towards the third generation. For example in Europe, the Universal Mobile Telecommunications System (UMTS) has been developed under the auspices of the Third Generation Partnership Project (3GPP). It is based on existing GSM/GPRS core network technologies and uses the Wideband CDMA standard (WCDMA) in the radio access network.

Similarly in the US, the Third Generation Partnership Project 2 (3GPP2) is a consortium of national standards bodies tasked with developing architectures and standards for third-generation cellular networks. The 3GPP2 has developed the cdma2000 network which is based on the existing cdmaOne IS-95B standard and uses a variant of CDMA in the radio access network [Sar00]. The International Telecommunications Union [ITU] has adopted International Mobile Telecommunications 2000 (IMT-2000) to allow seamless

evolution from the various 2/2.5G mobile standards that are extensively deployed around the world. Therefore it is not likely there will be one unified mobile system in the third generation. Figure 2.1 provides an overview of the various evolutionary paths towards the third generation.

2.2.1 3GPP

European efforts in 3G referred to as UMTS have been progressing in the Third Generation Partnership Project [3GPP]. The 3GPP was established in 1998 and consists of a number of telecommunications standards bodies, known as the 3GPP Organizational Partners [ARIB, CWTS, ETSI, ANSI T1, TTA, TTC], which is developing 3G standards for GSM based systems. UMTS is based on existing GSM/GPRS core network entities such as the GGSN and SGSN [CG97]. The standard interfaces and components of a 3G UMTS network are outlined in [3GPP03a] and illustrated in Figure 2-2 [VLLX02]. There are two land based network segments: the UMTS Radio Access Network (UTRAN) and the core network. The core network is further divided into the circuit- and packet-switched domains [CKK02].

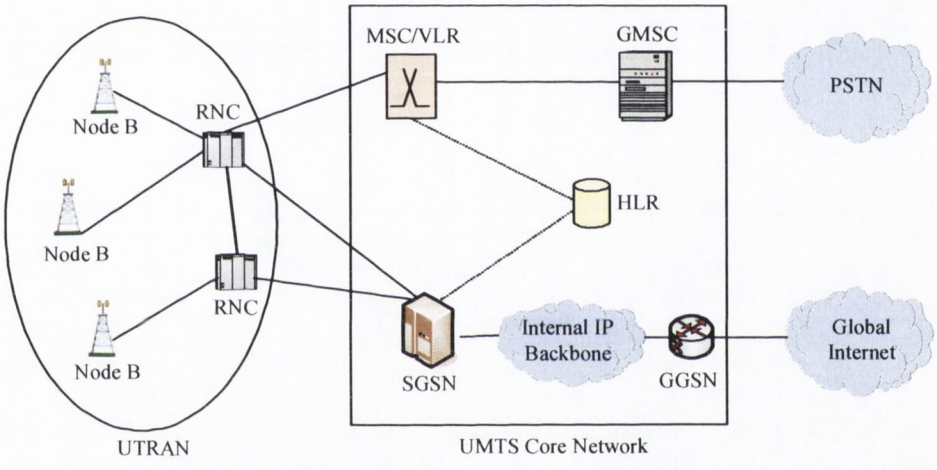


Figure 2-2 3GPP/UMTS Network Architecture

A mobile user's equipment communicates with multiple base stations, called Node Bs using WCDMA access and modulation techniques. The UMTS Radio Network Controller (RNC) can be considered to be roughly equivalent of the Base Station Controller (BSC) in GSM, and the Node Bs equate to the GSM Base Transceiver Stations (BTSs). Unlike GSM however the RNCs are connected together and can handle all radio resource issues autonomously. Each RNC controls a number of Node Bs and can perform *soft handover*. The RNCs and the base stations are collectively known as the UTRAN. From the UTRAN the core network is divided into packet- and circuit-switched parts. The Mobile Switching Center (MSC) and Visitor Location Register (VLR) handle connection-oriented circuit switching and mobility management tasks such as paging and location management. The Gateway MSC (GMSC) deals with incoming and outgoing connections to external networks such as the Public Switched Telephone Network (PSTN) for circuit-switched traffic.

2.2.1.1 Evolution of UMTS from GPRS

In evolving to an IP core network, 3GPP has decided to base it on the General Packet Radio Service (GPRS) [PD00]. GPRS has been standardized by the European Telecommunications Standards Institute [ETSI] to provide packet data services using GSM cellular networks [Par02]. GPRS improves the utilization of the existing radio resources. In addition, GPRS allows the subscriber to send and receive data in an end-to-end packet transfer mode, without using any network resources in circuit-switched mode. Packet-switched

networks are more suitable for bursty data applications. In general, packet-switched data services provide greater flexibility and allow for charging strategies dependent upon the volume of data transmitted, which is in contrast to the time-oriented charging techniques applied for circuit-switched connections [BW97].

The packet-switched portion of the network in GPRS/UMTS consists of two types of GPRS Support Nodes (GSNs), the Serving GPRS Support Node (SGSN) and the Gateway GPRS Support Node (GGSN). The SGSN can be viewed as a *packet-switched MSC* [PCCA04]. It is responsible for session management, authentication, ciphering, and routing of data packets to mobile stations within its service area. The GGSN is a gateway node between the GPRS and external Packet Data Networks (PDNs) such as the Internet. In the case of an external IP network, the GGSN is seen as an ordinary IP router and is responsible for the allocation of IP addresses. It converts GPRS packets from the SGSN into the appropriate Packet Data Protocol (PDP) and sends them out to the corresponding PDN. In the other direction, PDP addresses of the incoming data packets are converted to the GSM address of the destination user. The readdressed packets are then sent to the responsible SGSN. For this purpose, the GGSN stores the current SGSN address of the user and his profile in its location register.

The SGSN handles the inter-RNC mobility of a host, while the GGSN handles inter-SGSN mobility. Both the SGSN and GGSN are involved in the security functions and the generation of CDRs for billing purposes. All GSNs are connected via an IP-based GPRS backbone network. Within this backbone, the GSNs encapsulate the PDN packets and transmit them using the GPRS Tunneling Protocol (GTP). In UMTS, a tunnel is created between the GGSN and SGSN. A mobile node (also known as a mobile station in telecoms terminology) can move between base stations and base station controllers without changing the SGSN, by moving one leg of tunnel. It can move to an access network controlled by a different SGSN by moving both legs of the tunnel without disturbing the data session [New04].

GPRS allows a single mobile station to transmit on multiple time slots of the same Time Division Multiple Access (TDMA) frame. This results in a very flexible channel allocation scheme, whereby one to eight time slots per TDMA frame can be allocated for one Mobile Node (MN). Moreover, uplink and downlink channels are allocated separately, which efficiently supports asymmetric data traffic such as Web browsing. Before a mobile node can start using GPRS services, it must register with a SGSN of the GPRS network. The network checks the user authorization, copies the user profile from the HLR to the SGSN and assigns a temporary identity to the user.

To exchange data packets with external PDNs, a mobile node must apply for one or more addresses used in the PDN, e.g. an IP address in the case the PDN is the Internet. This address is called a PDP address. For each session a PDP context is created which describes the characteristics of the session. It contains the PDP type, the PDP address of the mobile station, the requested QoS, and the address of the GGSN that serves as the access point to the PDN. This context is stored in the MN, the SGSN and the GGSN. With an active PDP context a MN is able to send and receive data packets. A mapping between the MNs PDP address and International Mobile Subscriber Identity (IMSI) allows the GGSN to transfer packets between the PDN and MN. A user may have several simultaneous PDP contexts active at a given time [BVE99].

2.2.2 3GPP2

A simplified architecture for a cdma2000 network as defined by the Third Generation Partnership Project 2 [3GPT01, 3GPT00, 3GPT03] is shown in Figure 2-3. It is similar to the GSM/UMTS architecture. The main difference between the cdma2000 and UMTS architectures is that in a cdma2000 network there is a Packet Data Serving Node (PDSN). The PDSN establishes, maintains and terminates link layer session to mobile nodes and routes data packets to and from a MN. It is roughly the functional equivalent of the SGSN and

GGSN in a GPRS/UMTS network [VLLX02, PCCA04]. In 3GPP, GPRS-based mobility was already defined, so IP network enhancements were considered on top of GPRS. In contrast to this, the 3GPP2 has opted to make use of the IETF Mobile IP [Per96a, Per97] protocol for network-layer mobility support [MH00, PD00].

One of the main reasons for adopting Mobile IP was its global acceptance and the ease with which it would allow interworking and roaming with other IP networks. The PDSN provides the Foreign Agent (FA) functionality in 3GPP2, while the Home Agent (HA) is a separate entity. A tunnel is created between the MN and the PDSN in order to transport IP packets between the two. The MN can move between base stations and base station controllers without interrupting the data session [New04]. The 3GPP2 architecture also makes use of AAA protocols and infrastructure to authenticate roaming data users [Per00]. This allows the visited network to query the home network for authentication credentials and ensures payment for services rendered. This is in addition to existing HLR/VLR based authentication for wireless voice services. More details on Mobile IP and AAA can be found in Section 2.4.

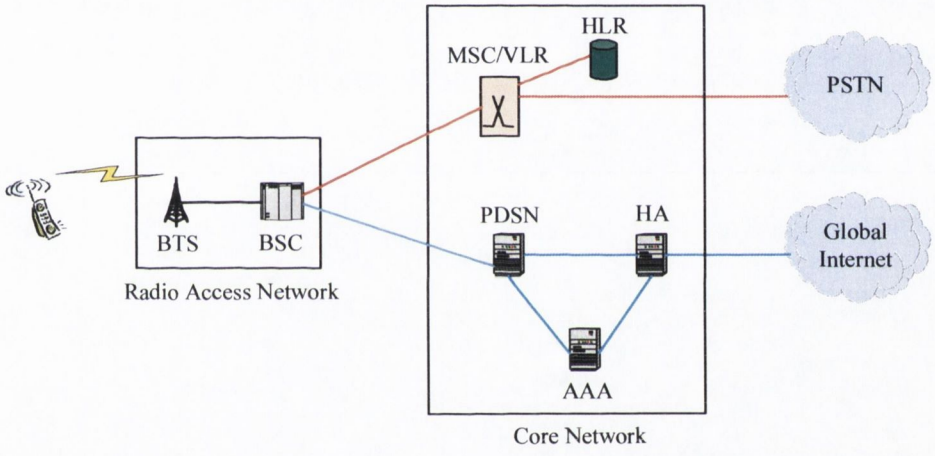


Figure 2-3 3GPP2/cdma2000 Network Architecture

The 3GPP2 has an evolutionary approach to the development of the cdma2000 network. Initial rollout has commenced with the introduction of cdma2000 1x, which is based on the cdmaOne standard and supports data rates up to 144Kbps. The next step is cdma2000 1xEV-DO (data only), which introduces a new air interface and supports up to 2.4Mbps on the downlink and 153Kbps on the uplink. It is also known as High Rate Packet Data (HRPD) and provides an *always-on* connectivity in a wide-area mobile environment. However, simultaneous voice over 1x and data over 1xEV-DO is difficult due to the use of separate carriers. Finally, cdma2000 1xEV-DV (data and voice) aims to introduce an *all-IP* architecture for radio access and the core network, and promises data rates up to 3Mbps.

2.3 Billing in Telecommunications Networks

All network operators, whether they are fixed, mobile, or value-added service providers have a requirement to generate revenue for the resources that they provide. The act of recording resource usage for the purposes of trend analysis, auditing, billing, or cost allocation is called *accounting*. Accounting in today's telecommunications networks is a complex process. It consists of a number of sub-processes, namely metering, pricing, charging and billing. Metering is the process of measuring and collecting resource usage information in the form of a Call Detail Record (CDR), related to a single customer's service utilization.

Pricing is the process of determining a cost per unit or *tariff* and is dependent on a number of factors such as the time of day, destination of the call (local, national and international), subscriber type and charging plan. The charging process applies the appropriate tariffs to translate the customer's resource usage information into an amount of money that he has to pay. This amount is then used by the billing process to generate an invoice for payment [PBSP01, KKAM+04].

There are a number of problems with existing telecommunications billing procedures, namely the cost of the actual billing process and the high level of fraud [Mep00]. Firstly, the cost of providing and maintaining a telecommunications billing system is quite high. It can be anything up to 50% of the total infrastructure investment and annual revenue of the network operator [CH00]. According to Jupiter Communications it costs an operator between $2.50 and $3 on average, to produce a paper bill for a residential customer, and more for business customers. Online billing and electronic payments could cut these costs in half, especially as half the calls to customer support centers are billing related. Secondly, the high level of fraud is an impediment to the development of next-generation networks. Recent reports indicate that between 1-3% of an operator's revenue is lost due to fraud each year, of which 24% is related to roaming fraud [Llo03]. This amounts to anywhere between 12-60 billion US dollars annually [BWorld, DIDATA03].

In traditional telecommunication environments, charging and billing for telephony services is based on a contract between the end user and the network operator. The essence of this agreement is that the user gets access to the network services and will pay for those services according to the bills produced by the operator. The bills may contain both fixed and usage-dependant charges. To compute the usage-dependent charges, the operator has to monitor all the calls placed and received on the network. For each call placed by the user, the switches in the network generate one or more CDRs. Millions of CDRs are stored on disk or magnetic tape and are used as the raw data for the billing system. Thus, the implementation of billing for telephony services is based on an offline billing system and on having CDRs generated in the network.

The problems of fraud and lost revenue are further exacerbated in mobile telecommunications networks where there may be roaming users associated with third party NOs. Bilateral agreements are required between the operators to allow users to avail of services while roaming in foreign networks. From the users perspective he must place total trust in the accuracy of the operator generated CDRs. This is particularly important in the case of roaming users where there may be multiple NOs involved in routing a call. Complicated tariff structures make it difficult for a user to recall their precise usage, let alone dispute it at a later stage. Subscriber based billing with unlimited credit is too open to abuse in mobile networks. With large numbers of mobile users and independent network operators, these problems will only escalate in the future. In order to facilitate the understanding of the charging, accounting, and billing processes in telecommunications networks, a brief description of the overall architecture is provided, together with a presentation of the network components involved in these functions.

2.3.1 Fixed Telecommunications Billing

In the PSTN as it exists in the United States, each network operator, the Local Exchange Carrier (LEC), Inter-Exchange Carrier (IEC), and possibly international carrier produces a CDR when a call is completed. Figure 2-4 adapted from [PPYS+99] depicts the overall CDR generation process. CDRs provide detailed information such as the source and destination parties, the duration of the call, QoS and other details. More than one CDR can be generated for a single chargeable event, e.g. because of its long duration, or because more than one party is to be charged. The CDRs are stored in a file at the local switch and periodically sent to a centralized billing system, usually at another location. The size of a CDR can vary between 20 and several hundred bytes.

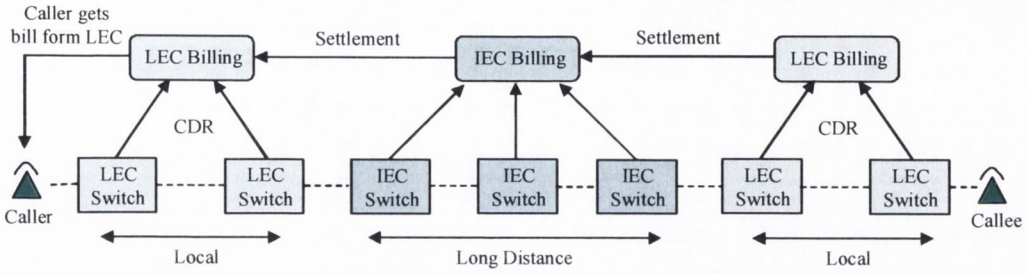


Figure 2-4 PSTN Billing

Although CDRs are in the process of being standardized, their contents can vary from operator to operator. This is partly due to the fact that the switching equipment from different manufacturers produces data in different formats. The only proof that a call took place is the CDR, and sophisticated duplication schemes are used to ensure that this data is not lost. The raw data generated by the switches is also kept for the purposes of settling disputes. However there are no mechanisms in place to ascertain the authenticity of the data, and the call records can either be denied by the customer or falsified by the operator. According to the Federal Communications Commission (FCC) of the United States, telephone company billing complaints jumped to 5,523 in the first three months of 2003, up 65% from a year earlier [FCC03].

2.3.2 Billing in GSM

The first GSM networks became operational in 1992 and initially consisted of only a handful of network operators. Today the Global System for Mobile Communications consists of more than 600 network operators with 74.9% of the world's digital market [GSMA04a]. In 1987, network operators from fifteen countries signed a Memorandum of Understanding (MoU) in Copenhagen, which aimed to promote and encourage the worldwide adoption of GSM. Other working parties were also established to solve the problems associated with building and administering a worldwide system based on independent networks. These groups were later transformed into the GSM Association [GSMA].

One of the first tasks of the GSMA was to address the issue of cross border roaming and billing procedures. The GSMA estimates that more than 20,000 individual roaming agreements are in place between its GSM operators worldwide, with more being added every day. GSM is founded on the concept of roaming, allowing subscribers from other networks to use their mobiles when they visit other countries or networks. However behind the simple objective of global roaming lies a complex process that gathers information about each call, about each caller, and takes a standardized approach to the charges being incurred.

The Transferred Account Data Interchange Group (TADIG) within the GSMA was given the task of implementing the interchange of billing data between different network operators by defining and implementing the Transferred Account Procedure (TAP) [GSMA04b]. TAP is the mechanism by which operators exchange CDRs of roaming users and allows roaming partners to bill each other for the use of networks and services through a standardized process. Much of the traffic carried by a GSM Public Land Mobile Network (PLMN) either originates or terminates in another network. The operator of the local fixed network charges the wireless operator for each call that terminates at one of its fixed-line subscribers. The same applies in the other direction where the GSM operator charges the fixed-line operator for each call made to a mobile number from a landline. Therefore GSM network operators and their local fixed-line counterparts usually negotiate an interconnect agreement to make charging as simple as possible. Other fixed-line international operators have similar agreements amongst themselves.

Thus in order to place a call for example from an Irish PLMN to a fixed phone in the US, it is not necessary for the two operators to directly negotiate a pricing contract. The Irish PLMN operator negotiates a price with an Irish PSTN operator, who in turn negotiates a terminating charge with its US counterpart. The Irish PSTN operator passes the call costs back to the Irish PLMN who recoups the cost of the call directly from its subscribers. This form of inter-administration accounting covers the division of revenue between both fixed and mobile networks. It does not, however, cover the costs incurred by foreign subscribers whilst roaming in other networks. TAP is required for inter-PLMN accounting for roaming mobile users. The latest version, TAP3, caters for variable length records and allows individual erroneous CDRs to be rejected. Figure 2-5 illustrates the collection and exchange of information required to support TAP.

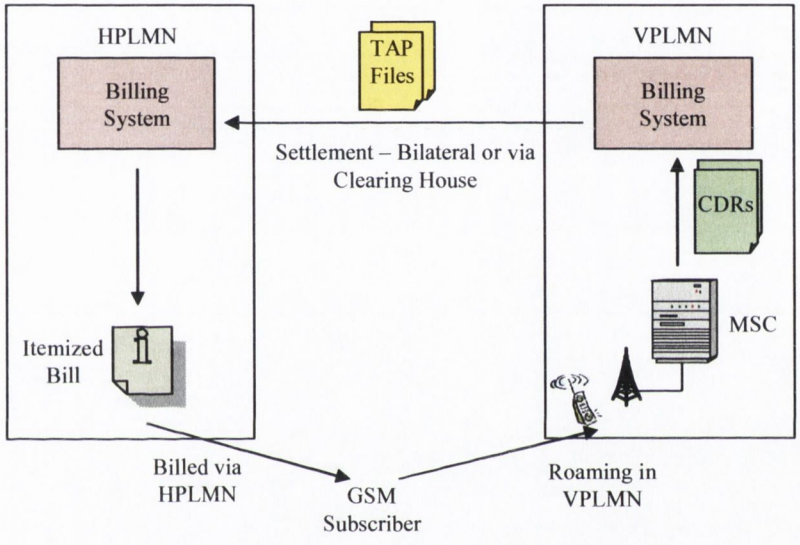


Figure 2-5 Generation and Exchange of TAP Records

In the case where a French subscriber calls a US number while roaming in an Irish PLMN, this call is routed via an Irish fixed network operator. The Irish fixed NO will still charge the Irish PLMN for the leg of the call placed to the US number. The Irish PLMN does not receive any revenue directly from the roaming mobile user. In order to recoup the costs incurred by the call, the Irish PLMN must charge the Home PLMN (HPLMN), in this case the French PLMN, to cover the costs incurred by the French mobile subscriber.

The details of the calls made by a subscriber roaming in a Visited PLMN (VPLMN) are recorded by the serving MSC. Each call produces one or more CDRs which are transferred on a regular basis to the billing system of the VPLMN for pricing. The CDRs produced on behalf of roaming subscribers are converted and grouped in files under the TAP format. The transfer of TAP records between the home and visited mobile networks may be performed directly or via a clearinghouse. Invoicing between the operators normally happens on a monthly basis. On reception by the HPLMN, the TAP record is converted into an internal format and added together with any CDRs produced by the subscriber while in the home network [3GPP03b].

In recent years network operators have started to introduce *prepaid* solutions. One of the advantages of prepaid subscriptions is that it helps to reduce the risk of unpaid accounts. However, roaming mobile users add to the complexity of prepaid schemes, as the majority of prepaid mobile solutions are based on temporary

accounts maintained in the home network. The *hot billing* approach is one solution which allows for the real-time collection of data and the transport of CDRs from the MSC in the visited network to the Prepaid Service Center (PSC). The balance in the customer's account is decremented according to the CDR. As a customer uses up the prepaid credit, the HLR and the Authentication Center (AuC) are updated to allow or prevent further service access. In the hot billing approach, sending real-time CDRs on a per second basis to the PSC may incur heavy overhead for the network [LCR00].

2.3.3 GPRS Billing

Second-generation networks mainly catered for voice telephony services. As a result, many wireless carriers, familiar with traditional telephony, have implemented wireless billing systems based on models that bill for *voice minutes* using CDRs. GPRS packet transmission on the other hand offers more user friendly billing than offered for circuit-switched services. Circuit-switched services billed based on the duration of the connection are unsuitable for applications with bursty traffic needs. The user must pay for the entire airtime, even for idle periods when no packets are sent e.g., when the user reads a Web page. In contrast to this, with packet-switched services billing can be based on the amount of transmitted data. The advantage for the user is that he can be online over a long period of time, but will only be billed on the transmitted data volume.

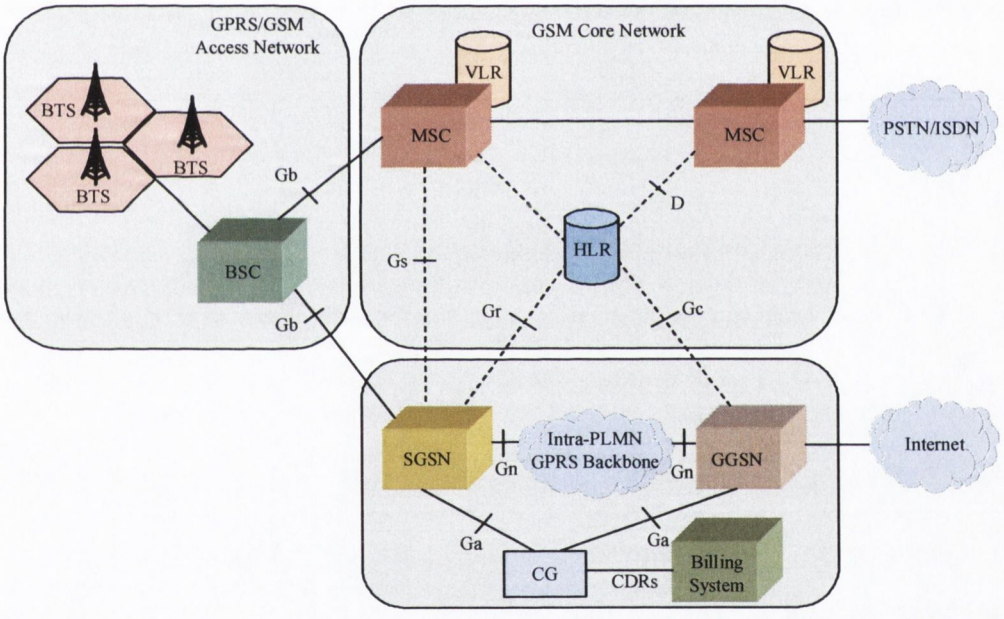


Figure 2-6 Mobile Data Billing

A GSM voice customer generates on average three CDRs per day, while a GPRS data customer is expected to generate at least ten times as many CDRs [SITI01a]. This increased amount of data raises many collection and mediation issues. Most of the information that will be used for billing purposes in GPRS is generated from the PDP context which is the underlying concept supporting the always-on feature. The PDP context establishes a virtual path (GTP tunnel) between the SGSN and the GGSN, and defines the different characteristics of the data session. The mobile user is allocated an IP address and connects to the appropriate GGSN giving him access to the appropriate content or service. The SGSN collects charging information for each MN related to the radio network usage, while the GGSN collects charging information for each MN related with external data network usage.

There are two main types of CDRs generated, one for the SGSN and one for the GGSN for each PDP context. A third record is provided for mobility management in the SGSN. The SGSN may also provide two SMS related records in case of short message delivery. The GPRS standard introduces new Charging Gateway Functionality (CGF), which provides a mechanism to transfer charging information from the SGSN and the GGSN nodes to the network operators Billing System (BS). The Charging Gateway (CG) concept enables an operator to have just one logical interface between the CGF and BS [ETSI00a]. Figure 2-6 shows the GPRS charging logical structure.

As the market for mobile voice saturates, operators are increasingly looking to mobile multimedia services in 2.5/3G networks to increase their revenue sources [New04]. At the present time GPRS users in Europe are being charged according to the volume of data being exchanged, i.e. operators are charging subscribers for access to the network and use of network resources regardless of what is being performed online. A major concern for the operators is that end users are not being charged for the value of the services they receive. Instead of just being charged for the transport of datagrams, the user should be charged for the contents value. There is a growing consensus that *content-based billing* is needed in next-generation mobile networks [SITI01b]. Content-based billing will enable the network operators to determine the type of data being transmitted over their networks and to increase their revenue streams [PBSP01].

The range of possible charging methods lies between two extremes, namely *flat-rate* charging and *usage-based* charging [Gin00]. The advantage of flat-rate charging is its simplicity, as it does not depend on the usage of the offered services. The disadvantage is that heavy users are charged the same as the occasional ones. Usage-based charging on the other hand involves the capturing and counting of the number of packets exchanged in a session and allows for better use of limited capacity and resources. The advantage of this method of charging is that the absolute usage of the network and services can be metered, calculated and billed, as long as packet information can be calculated efficiently. The disadvantage is that the charging process may be more expensive to implement than some of the services that are offered. Combinations of the various charging models can also be applied by the network operator in some situations. Other charging models such as Paris Metro Charging and Expected Capacity Charging are also possible [CH00].

2.3.4 Billing in a 3G Environment

The 3G/UMTS charging logical architecture shown in Figure 2-7 is similar to the one employed in GPRS networks. As opposed to a single CDR being generated by the MSC in GSM networks, various CDR types are required for the charging mechanisms to be employed for UMTS services. In UMTS networks, charging records can be generated by more than one SGSN for a PDP context due to routing area updates. In addition, different records are required from the SGSNs and GGSNs. Thus for a single PDP context, charging records are needed from both the SGSN and GGSN. A unique Charging ID (C-ID) combined with GGSN address is needed to enable correlation of CDRs produced from the same PDP context [3GPP03b, 3GPP04].

There are three different types of CDRs generated for each PDP context. An S-CDR is opened for each activated PDP context at the SGSN. It is used to record usage of the radio interface as well as general network resources. An M-CDR is also produced at the SGSN and is used to collect charging information related to the mobility management of a UMTS mobile. Each UMTS *attach* procedure produces an M-CDR even in the event of no data transfer. A G-CDR is produced at the GGSN and is used to collect charging information related to packet data information for a mobile node. Details of data traffic between the GGSN and the external PDN are recoded in the G-CDR [Oye01].

Together the CDRs provide information about the radio resource usage, the amount of data sent and received, and the source and destination addresses. The CGF provides a mechanism to transfer charging information

between from the SGSN and GGSN nodes to the billing system. It is up to the NO to decide how often CDRs are transferred from a GSN to the CGF. The CGF acts as a buffer for real-time CDR collection and may also perform specific activities such as the consolidation of CDRs in order to reduce the load on the billing system. The billing system is where the appropriate tariffs are applied to the recoded usage by the MN.

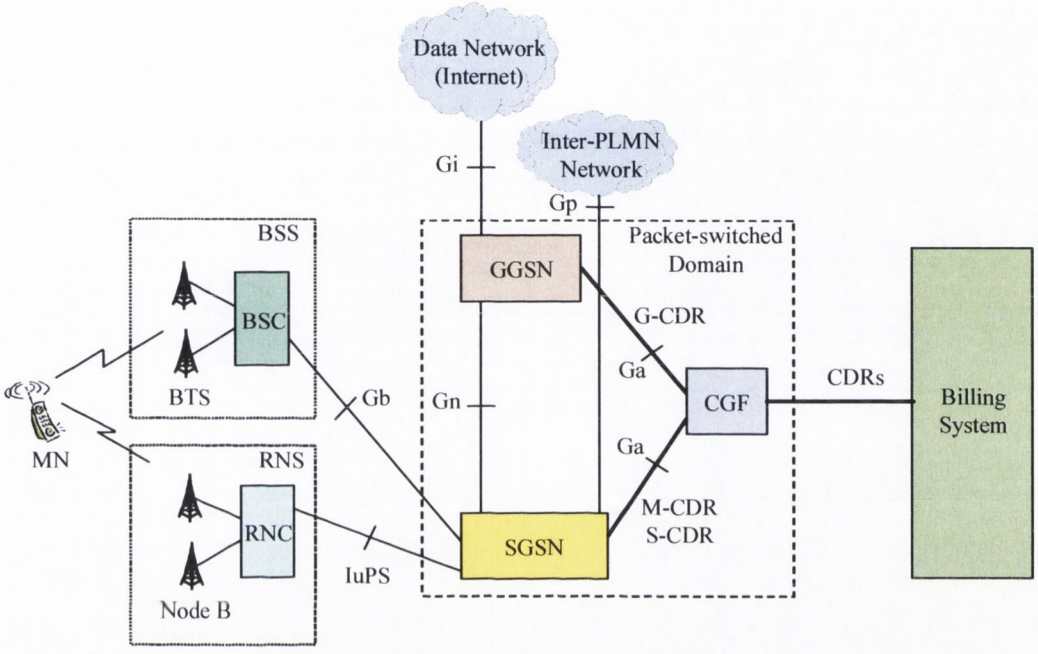


Figure 2-7 3G Packet Domain Charging Logical Architecture

In 2G networks, the network operator acted as an access network provider (e.g. GSM), and the exclusive service provider (e.g. voice telephony). However in 3G networks, these roles may be further subdivided into the access network operator (e.g. UMTS operator) and service provider (e.g. location management server). The access network operator role may be further decomposed into a radio access provider and core network provider. In addition to the above entities, there may also be content providers that will generate content e.g. text, audio and video, and negotiate distribution agreements with service providers [GKFK01]. Each of these new entities will need to be remunerated for their part in the service provision process. With multiple independent operators and providers, billing becomes a complex process and may require multiple trust relationships. The opportunity for mistakes and fraud in the system also increases. However from the user's point of view, regardless of who provides the services, the user should still only receive a single, itemized and accurate bill.

2.4 Mobile and Wireless Internet

Two technologies that have profoundly changed people's attitude to communication and information access in the last decade are the Internet and mobile communications. The Internet has become such an essential part of people's everyday life that they want to have it ready to use not only on their desktop machines but also in their mobile devices. Hence the development of a mobile wireless Internet is inevitable. However, at the present time, the approaches taken by the telecommunication providers and the Internet community to the development of a future mobile wireless Internet architecture are quite different. The evolution of cellular mobile systems towards support for Internet-style data services has already been highlighted in the previous

sections. It can be observed from the discussion in Sections 2.2 and 2.3 that the various 3G protocol architectures are far from the elegant simplicity of the classical Internet approach.

At the same time, there has been increased interest in the Internet research community to evolve the current Internet architecture to support wireless access links, QoS and mobility. Interest in this approach has been further boosted by the arrival of Wireless LAN (WLAN) and Wireless Personal Area Networks (WPANs) such as Bluetooth [BT]. WLANs and WPANs will enable people to easily create networks within their homes, share connections, and form closed user groups. This will enable large numbers of small independent network operators and service providers to establish pico-cellular networks. These NOs and VASPs in turn will require efficient authorization and accounting procedures to be in place. However, at the present time there is still no clear or consistent picture of what such a mobile wireless Internet, also frequently referred to as an *all-IP* next-generation network, should look like. In this section, the various technologies which will play a major role in shaping the future mobile Internet are explored in more detail.

2.4.1 Mobile IP

The IETF Mobile IPv4 protocol [MIP, MIP4] or Mobile IP for short is a well-known approach for mobility support in IP networks [Per02a, Per02b]. It enables a mobile node to maintain its existing transport layer connections when it moves from one sub-network to another. Mobile IP is a network layer protocol which is completely transparent to the higher layers and does not require any changes to the existing Internet hosts or routers. This is in contrast to the 2G and 3G architectures which solve the mobility problem at the link layer [AXM04, Edd04, PCCA04]. In normal IP routing, packets are routed from a source to a destination on a hop-by-hop basis, where a router makes a routing decision based on the network part of the address. Thus an IP address is used for routing and also specifies a point of attachment for a node on the Internet [For03].

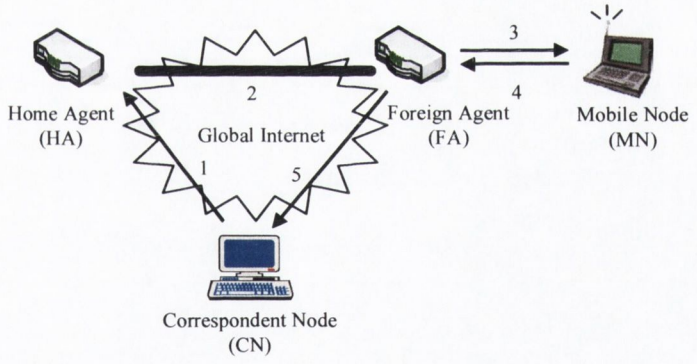


Figure 2-8 Mobile IP Network Architecture

For a host to continue to receive packets while it is roaming in a foreign network it needs to keep its IP address. Mobile IP solves this problem by adopting a *two-tier* addressing approach. In a Mobile IP network a roaming MN has two network addresses, one for identification and the other for routing. The first is known as the *home address* and remains unchanged regardless of where the node is attached on the Internet. The second is known as the Care-of Address (CoA) and changes at each new point of attachment in the network. The CoA may be a Foreign Agent care-of address, which is the static IP address of the FA in the visited network, or a Colocated Care-of Address (CCoA), which is a temporary IP address assigned to the MN.

A CCoA may be acquired through an auto-configuration process, such as the Dynamic Host Configuration Protocol (DHCP). The CoA may be shared by many roaming MNs in a visited network, whereas the CCoA

can only be used by a single MN at a time. A mobile node, using its home address, appears to be able to receive packets in a foreign network through a node in the home network known as the Home Agent (HA). The HA maintains a mobility binding that maps the MNs home address to a CoA. In Mobile IPv4, the router in the visited network that cooperates with the HA to deliver datagrams to a mobile node is known as the Foreign Agent (FA).

When a host roams into a foreign network, it obtains a new CoA which it registers possibly by way of the FA, with its home agent. The HA acts as proxy and attracts or intercepts datagrams that are destined to the home address of any of its registered mobile nodes. Whenever the HA receives datagrams addressed to the MN's home address, it tunnels them to the FA using the care-of address stored in its routing tables. Using IP-within-IP [Per96b], the home agent (tunnel source) inserts a new IP header (tunnel header) in the front of the IP header of the received datagram. The FA decapsulates the packets to reveal the MNs original home address and forwards them to the MN. Since the packets arrive at the mobile node with their home address as the destination, they are processed correctly by the upper layers. Figure 2-8 depicts the overall process of delivering datagrams to a roaming MN. Normal IP routing is used in the reverse direction for delivering datagrams from the MN to the Correspondent Node (CN). The CN can be a mobile or fixed node.

The MN sends a registration message to inform the HA of any change in its CoA or to renew a mobility binding. Registration messages need to be authenticated by the HA, prior to any updates to the CoA for a mobile node. Unauthenticated signaling messages can allow a malicious host to trick a MNs home agent into adding a false CoA for the node, and result in datagrams being redirected to an unknown network by the HA. Registration in Mobile IP must be made secure so that fraudulent registrations can be detected and rejected. Thus one of the primary issues that will determine the wider acceptability of the Mobile IP protocol is the secure and authenticated exchange of signaling data. Also, though the Mobile IP protocol provides an elegant solution for node mobility when the MN moves infrequently, this is not sufficient for pico-cellular environments, which require *fast handoffs*. In such environments, Mobile IP has shown to introduce significant latency as the registration messages may travel large distances before packet redirection occurs.

2.4.1.1 Route Optimization

The base Mobile IP protocol employs asymmetric routing, whereby datagrams from a CN to a roaming MN are routed through the HA. However, datagrams from the MN to the CN are routed directly to their destination (Figure 2-8). This asymmetric routing is far from optimal, especially when the MN and CN are in close physical proximity to each other. The extreme case is when the CN is part of the foreign network in which the MN is roaming, but due to asymmetric routing the datagrams must be routed via the distant home network.

This is known as the *triangular routing problem* in Mobile IP and can lead to high latency in the delivery of datagrams to their destination. Changes are required to the correspondent nodes to eliminate this problem. The basic idea of route optimization is to provide the CN with an up-to-date mobility binding of the current CoA of a MN. This allows the CN to bypass the HA and send encapsulated datagrams directly to the MN [PJ01]. As before the CN needs to verify the authenticity of all the binding updates that it receives. However the lack of robust security procedures and efficient key exchange protocols within the Internet make the implementation of the above difficult.

2.4.1.2 Regionalized Registration

Route optimization solves the problem of preventing encapsulated datagrams from being routed via the HA. However in highly mobile environments, large amounts of signaling traffic will still be generated between the visited and home networks by the Mobile IP registration process. If the distance between the visited

network and the home network of the MN is large, the signaling delay for these registrations may be long. In [GJP03], the authors present a regional registration scheme for Mobile IPv4 that makes use of a hierarchy of foreign agents and allows a MN to register locally within a visited domain. This reduces the number of signaling messages that have to be sent to the home network when the MN moves between FAs in the same visited network. It also improves handover performance as it reduces the latency associated with registering with the HA in a distant network.

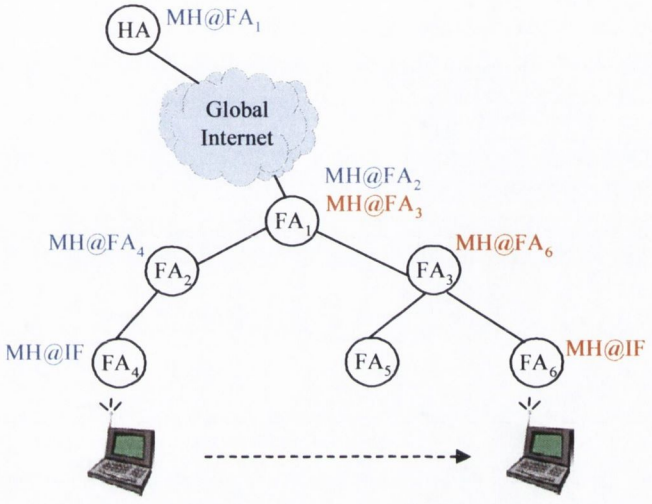


Figure 2-9 Hierarchical Foreign Agents

When a MN first arrives in a new domain it performs a registration with its home network. The home agent registers a CoA for the MN, which is usually the address of the Gateway Foreign Agent (GFA). When the MN moves between FAs under the same gateway, localized or regional registration can take place while the CoA at the HA remains the same. If there are a number of levels of FAs in the hierarchy then the agent advertisement message contains the addresses of the FAs between the leaf node and the GFA. Each intermediate FA also maintains a visitor list of foreign MNs that are currently roaming within the network.

During handover, regional registration messages travel only as far as the *crossover* FA, while the remainder of the path to the gateway remains the same. The crossover FA is the foreign agent lowest in the hierarchy which is part of both the old and new paths to the MN. As before all intermediate nodes must authenticate any regional registration messages that they receive. A registration key may be distributed to the MN and to the domain in which it is currently roaming. The key is used to prove the authenticity of registration messages generated by the MN. The MN also has a pre-configured secret with it's HA, while the FAs within a domain may share security associations.

Figure 2-9 adapted from [Per97] shows a hierarchy of foreign agents arranged in a tree topology within a domain. The mobile node performs a registration with its home agent via the path $FA_4 \rightarrow FA_2 \rightarrow FA_1$ using regular Mobile IP signaling. The home agent registers the publicly routable address of FA_1, the GFA, as the new CoA for the MN, while FA_2 and FA_1 add an entry into their visitor list. When the MN later moves from FA_4 to FA_6 it performs a regional registration operation in the local domain. This traverses the path $FA_6 \rightarrow FA_3 \rightarrow FA_1$ where the message is discarded by the gateway node. A new entry is added to the visitor list in FA_3 while the visitor list entry for the mobile node at the crossover router FA_1 is updated from MN@FA_2 to MN@FA_3. The CoA entry in the home network remains the same as before. Regional Registrations for

Mobile IP can also be classified as a micromobility scheme. Further enhancements for fast handoffs and paging support have also been proposed [SS02].

2.4.1.3 Mobile IPv6

Mobile IPv6 [MIP6, JPA03] makes use of the same basic network entities as Mobile IPv4, with the exception that there is greatly reduced need for foreign agents to be present in the network. The IPv6 protocol includes many features for streamlining mobility that are missing from IPv4 [For03, Tan03]. It also extends the address space to cater for the huge demand for IP addresses in the future. In Mobile IPv6, a MN can configure its care-of address by using the Stateless Address Autoconfiguration [TN96] and Neighbor Discovery [NNS96] protocols. Thus foreign agents are not required to support mobility in IPv6 [Per98].

Also, support for route optimization is a fundamental part of the protocol rather than a set of extension as in Mobile IPv4. Binding updates can be sent in normal datagrams from the MN to the CN using the destinations options in IPv6. Just as with Mobile IPv4 binding updates need to be authenticated. In order to comply with the IPv6 specification, each node is required to implement IPv6 authentication header processing. In Mobile IPv6 only the MN is authorized to provide binding updates to its CNs and does so whenever it changes its point of attachment in the network.

2.4.2 SIP Mobility

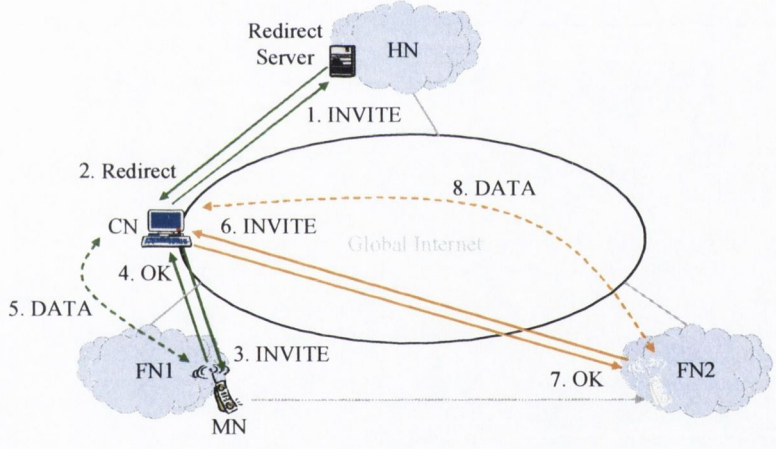


Figure 2-10 SIP-based Mobility

The Session Initiation Protocol (SIP) is an application layer control protocol that can be used for establishing, modifying and terminating sessions between users [SIP1, SIP2], and has been standardized by the IETF [RSCJ+02]. SIP has also become the protocol of choice to support IP multimedia services in both cdma2000 and UMTS networks [FLP04, HMP03]. It provides for user location services, call establishment, and call participant management for multimedia conferencing and Voice over IP (VoIP) services [SR99, SR00]. SIP is a text-based client-server protocol that defines a number of logical entities, namely user agents, redirect servers, proxy servers and registrars. The main function of the user agents is to initiate and terminate requests. Examples of SIP user agents are Internet telephones and conferencing software. Redirect servers respond with the location of another SIP user agent or server to which the user should send the next request. Proxy servers on the other hand are application-layer routers that forward SIP requests and responses. Finally, registrars keep track of users within their assigned network domain. A typical SIP server may implement the proxy, redirect and registrar server.

The SIP protocol supports both *personal* and *terminal* mobility by allowing users to maintain a single externally visible identifier regardless of their network location. SIP end points are identified by email like addresses which have the form *sip:alice@domain.com*. A SIP user agent typically registers its current network address with their local registrar. The SIP registration mechanism can be considered the application-layer equivalent of the Mobile IP registration mechanism. Whereas Mobile IP binds the MNs long-lived home address with a CoA, SIP binds a user-level identifier to a temporary IP address or a host name. A SIP redirect server has properties resembling those of a HA in Mobile IP with route optimization, in that it tells the caller where to send the invitation [SW00, BWDD+03, KGDD02].

Figure 2-10 shows how terminal mobility can be achieved using the SIP protocol. A MN that is roaming in the Foreign Network (FN1) registers its current location with its home registrar. When a Correspondent Node (CN) sends an INVITE to the MN, the redirect server in the Home Network (HN) redirects the INVITE to FN1. The MN responds with an OK message and data transfer can then commence. If during the session the MN moves from FN1 to FN2, it sends a new INVITE directly to the CN. Once the CN responds with an OK message data transfer can resume. However, SIP-based mobility management applies only to real-time communications over UDP. Transparent terminal mobility is not supported by SIP as it breaks the TCP connection [WS99, PCAG+04].

SIP must also deal with the issues of authentication of signaling messages and data confidentiality. Security in SIP is provided by either end-to-end or hop-by-hop protection. End-to-end mechanisms involve SIP user agents and are supported by features such as SIP authentication and SIP message body encryption [SVP02]. Hop-by-hop mechanisms on the other hand are used to secure signaling messages between two successive SIP entities and make use of a network-level security protocol such as IPsec [KA98] or Transport Layer Security (TLS). The IPsec protocol is discussed in more detail in Section 2.4.4.

2.4.3 Micromobility Architectures

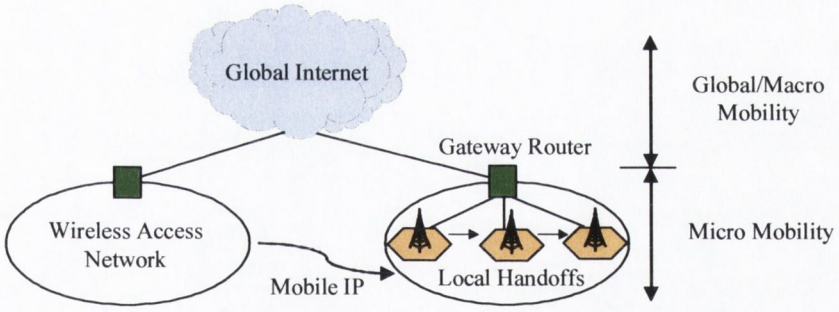


Figure 2-11 Intra- and Inter-domain Mobility

Mobile IP solves the *macromobility* problem in Wide Area Networks (WANs) by updating the care-of address at the HA, whenever a MN changes its point of attachment on the Internet. However Mobile IP suffers from a number of limitations which make it unsuitable for cellular networks and real-time traffic support. With Mobile IP, any change in a MNs point of attachment results in a signaling message being sent to the Mobile IP agent in the home network, even though most of the path between the MN and the HA remains unchanged. Mobile IP does not distinguish between different forms of mobility. For example, the same mobility procedures are applied regardless of the fact that a MN may be moving a short distance between two base stations or registering from a distant domain.

In addition, the base Mobile IP protocol does not have any provision for *paging* and expects a mobile node to update the HA on every move. Paging in cellular networks on the other hand facilitates efficient power management at the mobile node by allowing the MN to update the network less frequently, i.e. only when the MN moves between location or paging areas. The drawback associated with paging is that the network has only approximate location information of the whereabouts of a MN in the network when it is in idle mode. Prior to delivering any incoming packets the network has to perform a limited broadcast to determine the correct wireless access point.

Rapid proliferation in the number of wireless devices and networks has led to considerable research into *micromobility* protocols, both in industry and academia. The main aim of these protocols is to minimize the number of signaling or control messages that have to be carried over the core network to support host mobility. Micromobility architectures support mobility within a local network or domain. With micromobility protocols, local handoffs result in signaling messages being generated which are limited to the administrative domain in which the MN is currently roaming.

These signaling messages are intercepted by a *gateway node* or router which usually also acts as the CoA for the MN. As long as a MN remains within the administrative domain of a single network operator there is no need to update its route entry in the home network. Only when a MN moves between administratively separate domains is there a need to employ the Mobile IP protocol. Figure 2-11 illustrates the intra- and inter-domain mobility scenarios. A number of micromobility architectures have been proposed in recent years [CGKW+02, RB03]. In the following sections an overview of some of the better-known schemes is provided.

2.4.3.1 Cellular IP

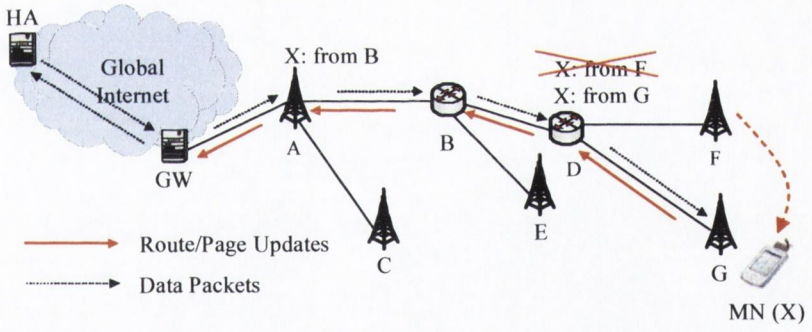


Figure 2-12 Mobility Management in Cellular IP

Cellular IP (CIP) is a micromobility protocol based around cellular telephony concepts such as passive connectivity, paging, and support for fast intra-domain handover for mobile nodes [CGKV+00, CIP]. The designers of CIP note that in future IP-based networks there will be a large number of mobile nodes, most of which remain passively connected to the network for extended periods of time. During this time a MN will only be required to send periodic signaling messages informing the network of its presence. However this implies that the network has only an approximate location for roaming nodes. Only when a MN is in active communication will the network need to find the exact location of the node by paging it. As with other micromobility schemes, Cellular IP efficiently caters for fast local mobility and relies on Mobile IP to solve the macromobility problem.

A CIP network consists of a number of nodes interconnected via wired links as shown in Figure 2-12. The leaf nodes are usually radio end points or base stations. The point of attachment to the Internet is a host known as the Gateway node (GW), which embodies both the home and foreign agent functionality. The GW is responsible for filtering out all signaling messages that are specific to the CIP network. Mobile nodes use the IP address of the GW as their Mobile IP care-of address. Cellular IP distinguishes between *idle* and *active* MNs and maintains two types of caches to hold hop-by-hop mappings for the same.

A *page cache* maintains a mapping for MNs that are not actively transmitting or receiving data but want to be reachable for incoming packets, whereas a *route cache* maintains mappings for only those nodes that are currently receiving or expecting to receive data. Paging cache entries have a longer timeout value than their counterparts in a route cache. Not every CIP node is required to maintain a page cache. However it is recommended that they all maintain route caches. Paging caches can be placed at strategic points within the network so as to maximize network efficiency.

The initial registration message transmitted by a MN is routed towards the gateway on a hop-by-hop basis. Each intermediate CIP node creates a *soft-state* mapping of the IP address of the MN and the neighbor that forwarded the packet. Subsequent data packets are used to refresh the existing cache entries which are valid for a system specific time known as the *route timeout* period. As long as the MN has data to send, the CIP nodes along the path to the GW keep an up to date mapping for the MNs point of attachment on the network. In cases where a MN does not have any data to send but wishes to maintain a valid route cache entry, it periodically sends a *route update* message towards the GW. Only route update control messages can be used to establish or refresh a route cache entry, while data packets can only refresh an existing route cache entry.

A Cellular IP network can be divided up into paging areas. Idle nodes need only report their position to the network when they move between paging areas or to periodically update page cache entries in order to remain reachable. A mobile node will periodically send a *page update* message towards the gateway. Each intermediate page cache along the path creates or updates an entry for the MN, while data packets can only be used to update an existing cache entry. In addition, a route update message can also update a corresponding paging cache entry for the MN. However page update control messages cannot refresh a route cache entry. Cellular IP also supports soft handoff by making use of *layer-2 triggers*, which notify the access points prior to actual handoff.

2.4.3.1.1 CIP Security

Page and route update messages can be used to create entries within the CIP caches which can result in changes to the routing of packets within the network. It is therefore of vital importance that each CIP node, prior to acting on any signaling information, authenticates all such messages. Unauthenticated signaling messages can be used to impersonate another node and create Denial-of-Service (DoS) attacks. A malicious host could generate false signaling messages and trick the node's home agent into adding a false CoA for the node. This would result in packets destined for a node being routed incorrectly by the HA to an unknown distant network. To prevent unauthorized entries being established in a cache, a CIP node must authenticate all signaling messages. This implies that there must be a pre-established security association between a MN and the CIP nodes with whom the MN is communicating. Also for seamless handover to occur there must be a security association with all other CIP nodes in the domain, or a means of transferring session keys to those nodes in a timely manner.

Cellular IP employs a fast session-key management scheme that allows authentication of control packets by CIP nodes. All the CIP nodes have knowledge of a *shared network key*. A session key for a MN is calculated by creating a hash (using MD5) of the concatenation of the mobile node's IP address (IP_{MN}), a random value

assigned to the host (R_{MN}) and the shared network key (K_{Nwk}) to produce $K_{Session} = H(IP_{MN} \| R_{MN} \| K_{Nwk})$ [OPT01]. The random number can be assigned by the gateway node and is carried in all signaling messages generated by the MN. The session key is calculated by the gateway node using the MD5 hash function when a MN first contacts the Cellular IP network to perform global Mobile IP authentication. A mobile node in addition has a cryptographic encryption key pair. One called the *public key* and the other called a *private key*. The public key is widely published while the private key is kept secret. Appendix A provides the reader with a detailed overview of hash functions and public-key cryptography.

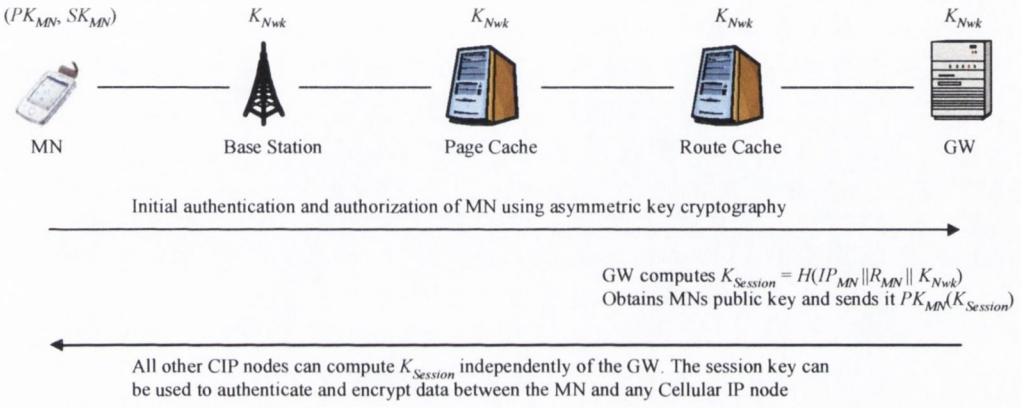


Figure 2-13 Key Management in Cellular IP Networks

The GW encrypts the session key with the public key of the MN (PK_{MN}) which it obtains from a Trusted Third Party (TTP) and forwards it to the MN. All other CIP nodes can independently compute $K_{Session}$ as they can obtain the IP address and random number from the signaling messages generated by the MN. The payload of a signaling message carries an authentication hash which consists of the session key, a timestamp and the packet contents. The authors of Cellular IP suggest that in order to improve security of the system, the shared network key should be changed periodically. Figure 2-13 shows the overall process of generating and distributing the session key.

There are a number of drawbacks with authentication scheme employed in Cellular IP. All CIP nodes have knowledge of the shared network key (K_{Nwk}). Repeated use of the key will lead to the decrease in the effective security of the key. However there are no mechanisms in place in the CIP protocol to automatically update the network key. Data packets can refresh a cache entry but do not need to be authenticated by a CIP node. This implies that a malicious node can inject data packets and keep cache entries alive to disrupt traffic flows. Also, there are no mechanisms in place to account for network usage within a CIP network. Once a node stops transmitting signaling or data packets the cache entries expire and are subsequently deleted by the CIP node. There will be no further record of the MN ever being present on the network.

2.4.3.2 HAWAII

HAWAII which stands for Handoff-Aware Wireless Access Internet Infrastructure is another micromobility scheme that provides efficient intra-domain handover [RPST+00, RPTV+99]. HAWAII defines the concept of a *domain*, which is essentially a wireless access network under the control of a single authority. Furthermore it divides the domain into paging areas and supports paging of nodes using multicast addressing. It allows MNs to retain their network address while moving within a domain which enables better QoS support. HAWAII divides the network infrastructure into a number of hierarchically arranged domains where each domain is connected to the Internet via a *domain root router*. Each host in a HAWAII network has a

unique IP address and a home domain. A domain can further consists of several routers and base stations as shown in Figure 2-14. Routing between the domain router and a mobile node is accomplished by setting up special dynamically established paths.

On power up the MN sends a Mobile IP registration request message to its nearest base station which adds a forwarding entry for the node. The base station initiates a HAWAII power-up message towards the domain root router. Intermediate routers will also add a forwarding entry (used during the reverse path) for the node. This entry consists of a message number, an IP address, an interface number and a multicast address, corresponding to the paging area to which the base station belongs. Multicast addresses are assigned to paging areas in HAWAII and all base stations within a paging area join that multicast group.

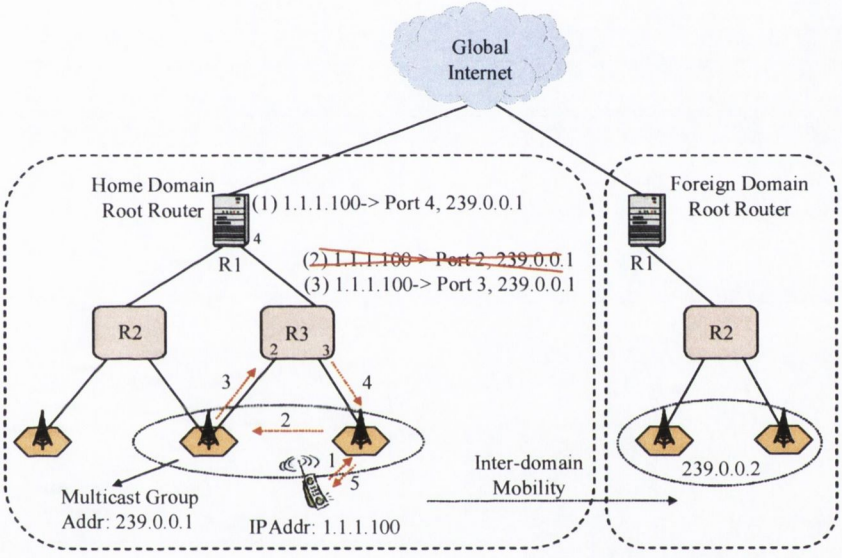


Figure 2-14 Intra- and Inter-domain Mobility in HAWAII

For intra-domain mobility HAWAII employs two separate path setup schemes. In the *forwarding path setup* scheme which is designed for TDMA type networks, the new base station sends a HAWAII handoff message to the old base station which informs the relevant routers of the new path. In the case where a mobile node can listen on multiple radio channels simultaneously e.g. CDMA networks, the *nonforwarding path setup* scheme is employed. The scheme does not require forwarding of datagrams from the old base station to the new one but an immediate switch at the first crossover router. For inter-domain mobility the MN acquires a colocated care-of-address in the foreign domain. Datagrams arriving in the home network are tunneled using the CCoA to the MN. As before, inter-domain mobility in the foreign network does not result in any updates in the home network.

HAWAII has to deal with the same security issues as any of the other regionalized IP mobility schemes, such as the authentication of signaling messages in the network. Security associations are required between the MN and HA, HA and FAs, and FAs and MN. The authors of HAWAII propose the use of the IETF AAA protocols for distribution of security keys between the entities concerned. The AAA protocol and current its implementations are described in detail in Section 2.4.4, highlighting their heavyweight nature and their unsuitability for micromobility environments.

2.4.3.3 TIMIP

Terminal Independent Mobility for IP (TIMIP) is another micromobility scheme which is based on many of same principles of CIP and HAWAII [GEN01]. However unlike those protocols, TIMIP can be implemented on mobile terminals without having to make modifications to their IP stacks. The authors of TIMIP present this feature as one of the strong selling points of the proposal, as replacing the protocol stacks on all legacy nodes can become an expensive task. TIMIP makes use of *layer-2 handoff* mechanisms at the wireless Access Points (APs) and avoids the need for special IP-layer signaling between the MN and the AP. Thus all IP signaling is completely implemented in network nodes and transparent to the IP layer of the mobile terminals. Authentication functions are implemented at the application layer.

A MN has to register itself with the Access Network Gateway (ANG) which authenticates it and broadcasts its layer-2 link layer address to all APs in the network. Data packets are used to keep alive soft-state routing entries in the access routers in the network and route update messages travel only as far as the crossover router. Data packets destined for a MN in the same TIMIP domain as the correspondent node travel only as far as the Access Router (AR) whose routing table has an entry for the destination. Data packets only reach the ANG in the worst case scenario. This is an improvement over the Cellular IP scheme where all packets must be routed through the gateway, even if the destination is located in a source next to the destination.

2.4.3.4 IDMP

The Intradomain Mobility Management Protocol (IDMP) is a lightweight micromobility protocol for managing mobility within a domain [DMDM+02]. IDMP is an extension of the base intra-domain protocol used in TeleMIP [SMAD00]. Unlike other micromobility proposals, IDMP makes use of *two CoAs*, and does not assume the use of Mobile IP for macromobility management. An IDMP network consists of a Mobility Agent (MA) which is the equivalent of a Gateway Foreign Agent (GFA), and one or more Subnet Agents which are equivalent to a FA in the Mobile IP protocol. On entering a new IDMP domain a MN obtains two CoAs. The first is known as the Local Care-of Address (LCoA) and identifies the MNs point of attachment to the subnetwork. The MN informs the mobility agent of its current LCoA. The second is the Global Care-of Address (GCoA) and can be the address of the MA. It does not change as long as the MN remains within the same administrative domain.

IDMP aims to improve intra-domain handover latency by eliminating the delay component associated with updating the LCoA at the mobility agent. It does this by making use of *layer-2 triggers*. This requires either the MN or base stations to indicate to the MA that there is an imminent change in connectivity. In response to this message the MA will multicast all incoming packets to all the subnet agents in a given set. Each of these subnet agents buffers the packets, and the appropriate agent immediately starts delivering them to the MN when it obtains in a new LCoA without having to wait for the MA to receive an update. Paging in IDMP assumes that the subnet agents are grouped into paging areas. MNs in the idle state do not need to obtain a new LCoA if they remain in the same paging area, even though they may have changed subnet agents. A limitation of IDMP is that it supports only a two-level hierarchy. According to the authors, IDMP shares the same security considerations as Mobile IP.

The workings of a number of micromobility protocols have been discussed in this section. It can be observed that there are a number of similarities in the design and operation of these protocols. It can be further seen that in each of the above protocols, there are certain key nodes or routers that maintain a route cache which holds soft-state mappings of the current location of mobiles within the access network. Many of the schemes rely on a hierarchical tree-like wireless network. Regardless of how the route cache is structured, a router must authenticate all signaling messages that it receives from a MN prior to any cache modifications. Also, a mobile node usually has a Security Association (SA) with its home network and must establish a temporary

SA in the access network. This may involve the use of a trusted key-distribution server or a dialog with the home network. The reader is directed top Appendix A of this thesis for a definition of a SA. To aid fast handoffs and maintain QoS, the requirement to contact a remote entity such as the home agent must be kept to a minimum. Many of the schemes also make use of layer-2 triggers and paging techniques to improve handoffs and power management in mobile nodes. Finally, there is a gateway or domain root router which connects the access network to a core network such as the Internet. This node can be used to collate accounting information of network usage by roaming MNs.

2.4.4 Billing and Security for Internet Services

The IETF has been working on the definition of a general Authentication, Authorization and Accounting (AAA) infrastructure for network access. Authentication involves validating the end user's identity prior to permitting them network access. Authorization defines what rights and services the end user is allowed once network access is granted. Last but not least, accounting provides a methodology for collecting information about the end user's resource consumption, which can then be processed for billing [AAA].

RADIUS [Met99, RWRS00] and Diameter [CLGZ+03, RHKS02] are two of the most widely used implementations of the AAA protocols today. With the increasing popularity of mobile devices, a need has been generated to allow users to access network resources in a foreign domain. Service provision in a visited network requires authorization, which leads directly to authentication of the user's credentials, and in turn to the accounting of network resources. Hence work has been ongoing within the IETF to support AAA services over emerging network technologies such as Mobile IP. Finally, the IP Security Protocol (IPsec) working group has been working on a standard to provide various security services for traffic at the IP layer [IPsec].

2.4.4.1 RADIUS

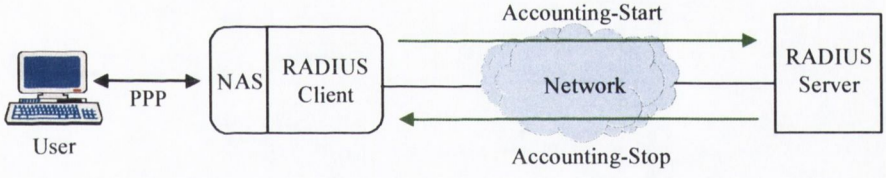


Figure 2-15 RADIUS Accounting

The most widely deployed AAA protocol is the Remote Dial In User Service (RADIUS), which was standardized by the IETF in 1997 [Met99, RWRS00]. RADIUS was developed to provide authentication and accounting services to Network Access Server (NAS) devices for dial-up connections. A NAS operates as a client of RADIUS and communicates with a RADIUS server over the network. A RADIUS server is responsible for receiving user connection requests, checking username and passwords for a matching entry stored in its database, and returning configuration information such as the types of services the user can access, in order for the client to deliver the requested service to the user. All communications between a RADIUS client and server are authenticated through the use of a shared secret. User passwords are sent encrypted between the client and the RADIUS server.

In addition to authentication and authorization, RADIUS was extended to provide accounting functionality [Rig97]. Figure 2-15 shows a RADIUS client and server which have been configured with the accounting extensions. The NAS forwards an *Accounting-Start* message to the RADIUS server describing the type of service being delivered and the user to whom it is being delivered. The client collects information about the session, the number of input and output octets and the session duration. At the end of the service delivery the

client will generate an *Accounting-Stop* message and sends it to the server. In each case an acknowledgement is sent back by the server.

2.4.4.2 Diameter

The RADIUS protocol was designed to support small numbers of users requiring simple server-based authentication. However with the advent of laptops and low-cost handheld devices, users are demanding services they have access to in their home network while roaming in foreign networks. Service providers must now provide AAA services to thousands of concurrent users, accessing network services over wireless and fixed communications links. With this in mind the IETF has developed the Diameter base protocol [CLGZ+03, RHKS02]. Diameter is an extensible peer-to-peer protocol which may be used on its own for accounting purposes only, or can be used with a Diameter application such as Mobile IPv4 to support inter-domain mobility [CJPH04]. A Diameter client is a device on the edge of the network, such as a NAS or FA that performs access control. The protocol also supports the use of *brokers* which facilitates roaming Mobile IP users. The Mobile IP AAA trust model is discussed in the following section.

2.4.4.3 AAA for Mobile IP

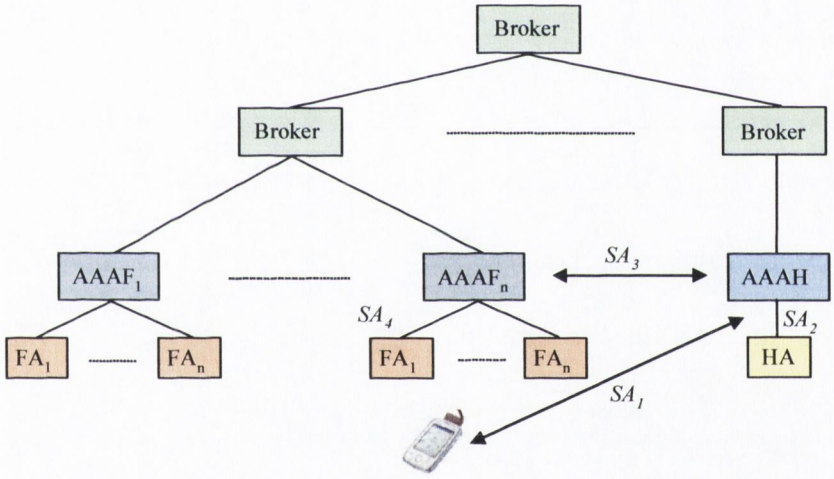


Figure 2-16 AAA Trust Model for Mobile IP

The Mobile IP and AAA working groups in the IETF have been looking at defining requirements to allow for the authentication and collection of accounting information of network usage by mobile nodes [GHJP00, Per00, PC04, SBGP+03]. Figure 2-16 shows the entities involved in authenticating and registering a MN in a foreign network using the AAA infrastructure. A foreign domain contains one or more AAA servers (AAAF) and multiple foreign agents. The FAs interact with a mobile node to authenticate its credentials. A foreign agent has a security association with its local AAA server, which in turn may have further security associations with other AAA servers.

If the AAAF cannot verify the credentials of a mobile node it can contact the MNs home AAA server (AAAH) with whom it must share a SA. A security association at a very minimum consists of a shared secret between two entities. The specification recommends that SAs between the various entities in the network should be set up using the IPsec [KA98, TDG98] protocol. AAA servers identify clients and subsequently their home domain by using the Network Access Identifier (NAI) which is of the form *user@realm* [Abo99]. A mobile node can identify itself by including the NAI along with the Mobile IP registration request.

In the AAA trust model for Mobile IP, a mobile node shares a security association SA_1 with the AAA server in its home domain. The AAAH in turn shares a security association SA_2 with the home agent. It is also necessary for the AAAH and the AAAF to share a security association SA_3, in order that the AAA server in the foreign domain can verify the credentials of roaming mobile node. Finally, the FA must share a security association SA_4 with the AAAF, in order for it to allocate local resources to a mobile node. For scalability reasons the concept of *brokers* (AAAB) is employed, which means that a foreign domain does not need to keep security associations with every possible home domain. The use of brokers in the system requires that the two administrative domains have security associations with the broker. The broker then becomes privy to all security exchanges between the two domains and also has to be trusted.

Once a mobile node has been authenticated three session keys are generated by the AAAH. Each session key that is generated by the AAAH is generally distributed to two entities. The method by which the key is encoded is dependent upon the security association between the entities. The Mobile-Home key $K_{MN,HA}$ is shared between the mobile node and the home agent. It is securely transported using the security association SA_2 for the HA and SA_1 for the MN. For mobile nodes currently roaming in a foreign network, this key has to be transported via the AAAF and the serving FA in the foreign network. The Mobile-Foreign key $K_{MN,FA}$ is shared between the MN and the FA. It is securely transported using SA_3 for the FA and SA_1 for the MN. The AAAF forwards the key to the correct FA using the security association SA_4. Finally the Foreign-Home key $K_{FA,HA}$ is shared between the FA and the HA. It is securely transported using SA_3 for the FA and SA_2 for the HA. Once the session keys have been distributed, there is no need to invoke the AAA protocols until the keys expire. During intra-domain handover the new FA will contact the AAAF and obtain the session keys $K_{MN,FA}$ and $K_{FA,HA}$, which were previously assigned to the old FA.

From the above discussion it becomes increasingly clear that the overheads incurred in setting up the AAA security associations and the transferring of cryptographic material can be quite substantial. In micromobility environments where a mobile node may change its point of attachment within the access network frequently this may lead to degradation in the QoS, as routers in the new path will need to be informed of the existing SAs prior to making any routing decisions. In addition, inter-domain handover requires that the MN obtain a new set of session keys from the AAAH server and have the keys distributed to all entities concerned.

2.4.4.4 IPsec

The IP Security Protocol is an IETF standard [KA98, TDG98] which provides authentication and privacy services at the network layer. Some of the security services offered include access control, protection against replay attack and confidentiality. These objectives are met through the use of the Authentication Header (AH) and Encapsulating Security Payload (ESP) protocols. The authentication header is used to protect the authenticity and integrity of an IP packet with a keyed cryptographic hash value [MS95]. The AH protocol inserts an additional header between IP and the transport layer headers that includes some authentication data. The encapsulating security payload provides confidentiality by encrypting IP packets or their payloads. The protocols may be applied alone or in combination with each other and support two modes of operation. The *transport* mode provides protection primarily for the upper-layer protocols e.g. the payload of an IP packet, whereas in *tunnel* mode, IP encapsulation is used to provide protection to the whole packet [Opp98].

IPsec also makes use of the concept of a security association, which is used to exchange information about cryptographic keys, algorithms and parameters to be used between communicating entities. A SA in IPsec is a one-way relationship between a sender and a receiver. If a peer relationship is needed then two SAs are required. A SA is uniquely identified by a triple consisting of a Security Parameter Index (SPI), an IP destination address and a security protocol identifier (AH or ESP). IPsec assumes that there are pre-existing security associations between entities that wish to use the protocol.

It recommends the use of the Internet Key Exchange (IKE) protocol for this purpose, which allows both parties to agree upon cryptographic algorithms and parameters and to perform a key exchange [HC98]. The protocol is quite complex and consists of two sub-protocols namely ISAKMP [MSST98] and OAKLEY [Orm98]. A key exchange in IKE is a three step process, in which the first phase consists of the exchange of cookies that help in preventing DoS attacks. In the second phase, both parties perform a Diffie-Hellman key exchange to mutually compute a session key. Finally, in the last phase each party authenticates the Diffie-Hellman parameters to protect against the man-in-the-middle attack by exchanging digital signatures.

The IP Security Protocol specifications and architecture are complex in nature, and non-trivial to implement for large-scale networks. Making a host IPsec compliant requires changes or extensions to the existing IP protocol stack. It is not an end-to-end protocol and requires pair-wise security associations to be setup between the various entities in the system. IPsec does not define a key exchange mechanism and an additional protocol such as IKE has to be employed. The IKE specifications too have been criticized for being far too difficult to understand [PK00]. In the following sections two technologies which have the potential to revolutionize the development of next-generation mobile networks are examined, namely Wireless LANs and Ad Hoc Networks. The IKEv2 draft [Kau04] currently under construction is an attempt to simplify the standard, remove those requirements not needed, and incorporate within this one new standard IPsec functionality currently contained within other documents.

2.4.5 Wireless LANs

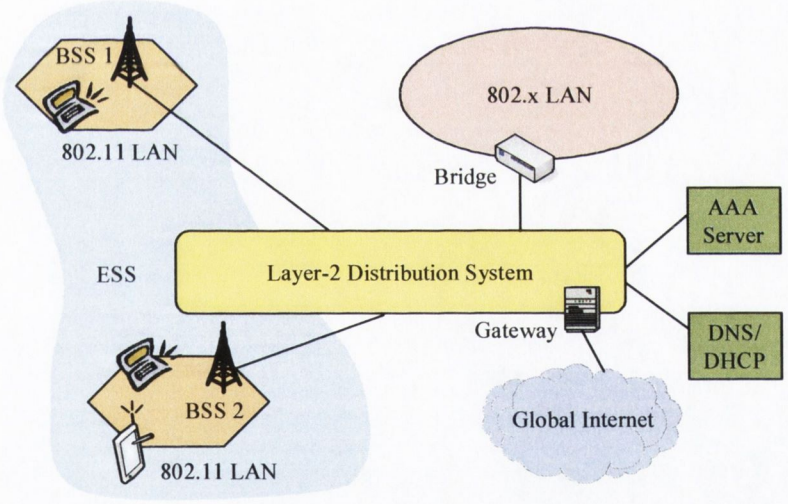


Figure 2-17 Infrastructure-based 802.11 LAN

Wireless Local Area Networks (WLANs) were primarily designed to provide high-speed connectivity to personal computers such as laptops and PDAs in corporate and home environments. The main attraction of WLANs is their flexibility, ease of installation, and cost-efficiency compared to traditional wired networks. They can extend access to local area networks such as corporate Intranets as well as support *broadband* access to the Internet. To date, public wireless access has typically been provided by cellular network operators. WLAN based systems are now emerging as a new means of public wireless access. WLANs based on the IEEE 802.11 standards [IEEE802] enable wireless networks that support data rates of up to 54Mbps over areas of a few thousand square meters. HIPERLAN/2 is another WLAN standard which is being

standardized by the ETSI Broadband Radio Access Networks (BRAN) project [BRAN]. It operates in the 5GHz band and has built in support for QoS and security. WLAN installations in public spaces such as airports, hotels and shopping malls are also referred to as *hot spots* [Var03].

Wi-Fi (short for wireless fidelity) is the popular name for WLANs based on the IEEE 802.11 standards. In the past few years Wi-Fi has emerged as the dominant standard for wireless LANs and has created broad interest in the public media. There are a number of 802.11 standards that operate in the Industrial Scientific and Medical (ISM) band at 2.4GHz and also at 5GHz. The ISM band is designated for unlicensed commercial use. The first of these was the 802.11 standard which allowed a maximum data rate of 2Mbps, whereas the current 802.11b standard allows data rates of 11Mbps. The emerging 802.11g standard provides a maximum data rate of 54Mbps and makes use of Orthogonal Frequency Division Multiplexing (OFDM) to achieve higher data rates, and is backward compatible with the 802.11b standard. In addition to WLANs, devices such as microwaves, garage door openers and Bluetooth equipment also operate in this band. WLAN users can therefore experience significant interference from other devices. However, the cost of establishing a Wi-Fi hot spot is significantly lower than that of setting up a cellular node using licensed spectrum [Boingo03].

IEEE 802.11 wireless networks operate in one of two modes: *ad hoc* or *infrastructure*. In ad hoc (peer-to-peer) mode nodes communicate with each other directly. However if a node wishes to communicate with nodes outside of the network then a routing protocol has to be employed. This mode can be considered to be a *pseudo ad hoc* mode. General mobile ad hoc networks are discussed in more detail in Section 2.4.6. In infrastructure mode each node communicates with an Access Point (AP). The AP acts as an Ethernet bridge and forwards packets to the wired and wireless networks. The MNs and the AP which are within the same radio coverage form a Basic Service Set (BSS). A Distribution System (DS) connects several BSSs thereby extending the wireless coverage to form an Extended Service Set (ESS).

More than 36 million Wi-Fi cards were shipped in 2004 [ITFacts], and it is estimated that that there will be in excess of 100,000 hot spots within the next five years [Nic03]. However for the continued popularity and growth of WLANs, seamless roaming between different networks is required in order to provide ubiquitous wireless broadband access [HL02]. At the present time there are dozens of players in the market that will provide hot spot coverage from small Internet Service Providers (ISPs), to large telecommunications companies like T-Mobile USA. The number of new entrants is growing at a rapid pace and each provider has its own business and billing plans.

A number of approaches have been adopted by providers to facilitate limited roaming for users between their networks. Large companies are in the process of building hot spot networks with national coverage. Examples of such Wireless Internet Service Providers (WISPs) or Hot Spot Operators (HSOs) are Wayport and Cometa Networks. Aggregators like Boingo or iPass on the other hand do not build networks but instead establish business relationships with WISPs and HSOs to share in the revenue stream. They sign up subscribers and offer them a single account which gives them wireless access through the aggregator's partner networks. However for large scale roaming to succeed, providers must resolve differences in their business models, billing and authentication mechanisms.

2.4.5.1 WLAN Security

Many organizations are in the process of deploying 802.11 based WLANs as an extension to their existing wired infrastructure. In some cases this can open a *back door* into the organizations network and allow access to hosts even behind the organizations firewall. The 802.11 standard has limited support for confidentiality through the Wired Equivalent Protocol (WEP). 802.11 based WLANs are vulnerable to the *parking lot*

attack, whereby an attacker can sit in the car park and snoop traffic on the organizations network, as in most cases even the limited WEP protocol is not enabled [ASWZ02].

WEP is based on the RC4 stream cipher and requires a shared symmetric key for encrypting traffic between the AP and wireless devices in a BSS. However the key sizes used in WEP are only 40 or 104 bits and are open to cryptanalysis. In addition, the protocol concatenates the key with a 24-bit Initialization Vector (IV) and sends this as cleartext to the receiver. A new IV is generated for each packet that is sent over the radio interface. However since the length of the IV is only 24 bits, it is likely to be reused over a short period of time. An attacker could collect an IV and use it to retrieve the encryption key [PD03, Wal00]. The reader is once again redirected to Appendix A for an overview of the various cryptographic terms and notation used throughout this thesis.

802.11 WLANs are also vulnerable to the *link-layer address spoofing* attack. Each IEEE 802.11 Network Interface Card (NIC) has a unique 48-bit address assigned by the IEEE which is used for access control decisions. However WEP broadcasts the link-layer address in cleartext during packet transfers. An attacker can easily eavesdrop to obtain the link-layer address of a node and use this at a later stage to gain access to the network, as most open source drivers now allow a user to change their cards link-layer address [HA03].

There are a number of initiatives to address the security shortcomings of 802.11 LANs. The first is the 802.1x port-based access control mechanism which provides mutual authentication between the network and its clients. With 802.1x, the user is authenticated to the LAN through a RADIUS server in the network. It also addresses the issue of rogue APs that may pop up in the network. A second important feature of 802.1x standard is its support for frequent key exchange known as the Temporal Key Integrity Protocol (TKIP). Encryption will be addressed by the 802.11i task group and it is expected that it will use TKIP and the Advanced Encryption Standard (AES) for encryption services. An alternative proposal is to make use of Virtual Private Networks (VPNs) to create a secure tunnel between the user's device and destination [Var03].

2.4.6 Mobile Ad Hoc Networks

Ad hoc networks have become a topic of increased interest in recent years, due to the availability of inexpensive wireless devices and the desire for ubiquitous mobile communications [RR02]. Traditional mobile networks consist of nodes that communicate with fixed base stations and a mobile switching center in order to communicate with other mobile or fixed nodes. More recently *self-organizing* wireless networks have been proposed which require no fixed infrastructure to be in place for communications to occur. Such networks are referred to as ad hoc networks and can be standalone or used to extend the reach of the wired infrastructure such as the PSTN and Internet.

An ad hoc network is an infrastructureless mobile network which consists of a group of nodes that communicate with each other using multi-hop wireless links. Individual nodes are responsible for dynamically discovering other nodes in their neighborhood with whom they can communicate directly. A key assumption is that not all nodes in a network will be able to communicate with each other directly. Thus nodes must also act as routers to forward packets for others in the network. Another feature of ad hoc networks is the dynamic changes that can occur in the topology and link characteristics due to node mobility. Routing protocols for ad hoc networks must therefore be able to discover and maintain routes to other nodes using shared wireless links in the presence of rapid changes and hidden or exposed nodes. Finite power availability on mobile terminals and limited bandwidth in wireless environments also play a part in determining the development of such protocols. The IETF Mobile Ad Hoc Networks (MANET) working group is in the process of standardizing a number of routing protocols [MANET].

2.4.6.1 Routing and Security

The majority of routing protocols for ad hoc networks can be divided into two main categories, namely table- and demand-driven [RT99, MMDM04]. Table-driven or *proactive* routing protocols attempt to maintain a consistent up-to-date view of the current topology of the network. These protocols require each node to maintain routing tables and to respond to changes in the network topology by propagating updates throughout the network. Demand-driven or *reactive* protocols in contrast create routes only when required by the source or initiating node. When a node requires a route to a destination it initiates a route discovery process and subsequently uses route maintenance procedures to maintain the route as long as it is required or becomes unavailable. A number of proactive and reactive routing protocols are being considered by the MANET working group at the present time. A brief overview of some of the more prominent ones is provided below.

The Optimized Link State Routing (OLSR) protocol is a proactive protocol and provides the advantage of having routes immediately available in each node for all destinations in the network [CJ03]. OLSR is an optimization on a pure link state protocol. The optimization is based on Multipoint Relays (MPRs) which are selected nodes in the network that are only allowed to forward broadcast messages during the *flooding* process. Flooding is the simplest form of routing where a node broadcasts a message to all its neighbors. If a node other than the destination receives the message for the first time, it re-broadcasts the message to all its neighbors. This reduces the number of retransmissions of broadcast control messages in the network.

OLSR has two types of control messages, namely *Hello* and *Topology Control* (TC) messages [BMA02]. Neighbor Discovery is the process by which a node detects the nodes with which it has a direct link. The node broadcasts Hello messages at periodic intervals (e.g. every 2 seconds), containing the list of neighbors known to the node and their link status. The Hello messages are received by all one-hop neighbors but are not forwarded. This enables each node to discover its one- and two-hop neighbors. Each node then selects its own set of MPRs from its set of one-hop neighbors. Each node in the network also maintains topological information about the network by means of TC messages via the MPRs at periodic interval (e.g. every 6 seconds). The TC messages are flooded to all nodes in the network and take advantage of the MPRs to reduce the number of retransmissions.

The Dynamic Source Routing (DSR) protocol is a source-initiated on-demand routing protocol for multi-hop wireless networks. Each node maintains a route cache that contains the source routes of which the mobile is aware. The protocol is referred to as on-demand or reactive, as route discovery is only initiated when a node needs a path to a destination. The initiator broadcasts a Route Request (RREQ) packet with the target node as the destination along with a unique identifier. Intermediate nodes that receive the RREQ check their route cache to see if they have a valid route to the destination. However if no route exists, the node appends its own address to the route record and forwards the packet. To limit the signaling traffic in the network, a node only forwards a RREQ if it has not already seen the request and its address does not appear in the route record. A Route Reply (RREP) is generated when a route request reaches the destination or an intermediate node with a valid route. Assuming that the radio links are symmetric the destination node simply reverses the route record and unicasts the RREP to the source. A node that initiates route discovery may receive more than one RREP and uses the route with the shortest number of hops [JMH03].

The Ad Hoc On-Demand Distance Vector (AODV) protocol is another reactive protocol and does not require maintenance of routes to destinations that are not in active communication. Moreover, AODV provides loop-freedom that is accomplished through the use of *sequence numbers*. Each mobile node maintains its own sequence number that it increases monotonically each time it learns of a change in the topology of its neighborhood. This sequence number ensures that the most recent route is selected whenever the route discovery process is executed [PRD03].

The AODV protocol uses RREQ messages flooded through the network in order to discover the paths required by a source node. An intermediate node that receives a RREQ replies to it using a RREP message only if it has a route to the destination, whose corresponding destination sequence number is greater or equal to the one contained in the RREQ. Otherwise, the intermediate node broadcasts the RREQ packet to its neighbors until it reaches the destination. The destination unicasts a RREP back to the node from which it received the RREQ. As the RREP is propagated back to the source, all intermediate nodes set up forward route entries in their tables. Route maintenance is accomplished through the use of Route Error (RERR) packets. RERR packets are generated at a node when the data link layer encounters a fatal transmission problem. Following the reception of a RRER message a node initiates a route discovery to replace the failed paths.

Most of the current routing protocols implicitly trust all the participants in the network to cooperate in the forwarding of datagrams. This approach works well in closed user groups such as a military or emergency services network, where the users are motivated to work with each other to achieve a common goal. However in situations where the nodes may have affiliations with different network operators or are under the administrative control of individual users, the cooperative nature of the network may break down. These users may not necessarily wish to cooperate in the relaying of packets on behalf of other users in the network. One of the major concerns of such users will be to preserve the limited battery life of their mobile devices, and the relaying of packets directly impacts on this. Work has been done to *stimulate cooperation* in such self-organized mobile ad hoc networks [HGBV01, LPW03, SBHJ, ZCY03].

Ad hoc networks are envisaged as complementing cellular networks and wireless LANs in the future for providing *last-mile* access to the wired infrastructure [TO03b]. As always, authentication and accounting will play a major role in determining the success of such a network. Thus there must be mechanisms in place to *compensate* the nodes involved in the relaying process and for any value-added services that they provide. However it is not always possible to reach the fixed network or a TTP to verify user identities or payment tokens that may be presented to a node. In the latter case, the cost of contacting a TTP to verify payment tokens may outweigh the actual benefits gained.

Ad hoc networks are also vulnerable to attack from outside sources due to their dynamic topology, distributed cooperation and the use of broadcast wireless links. Active attacks against the network can be mounted by a malicious node masquerading as a legitimate node to advertise false routes and disrupt traffic flows in the network, e.g. the *black hole* and *routing table overflow* attacks [DLA02]. Nodes in an ad hoc network can be compromised or hijacked to launch internal attacks against the network. More sophisticated attacks can be mounted whereby a node selectively drops datagrams and points to collisions on the transmission medium as the probable cause. Routing security plays an important role in the security of the entire network. Therefore mechanisms are required to monitor such malicious activity in order to be able to identify nodes that misbehave and subsequently blacklist them from the network [MGLB00].

Lack of centralized monitoring or management points makes the task of identifying misbehaving nodes difficult. Due to the dynamic nature of the network it is difficult to establish long-lived trust relationships in ad hoc networks, and security solutions with static configurations are not adequate. Also the limited capability of mobile devices in terms of processing and power requirements means that use of heavyweight cryptographic algorithms for authenticating a nodes identity and securing routes becomes impractical [BBCG+01]. Finally, an ad hoc network may consist of hundreds or even thousands of nodes. Therefore it is imperative to employ lightweight and scalable security mechanisms to address the above problems.

2.5 3G and Mobile Internet Interworking/Integration Issues

Two very distinct approaches to providing mobile wireless communications in current and next-generation networks have been examined in the previous sections. The traditional telecoms operator approach has been to build and operate large regional or country-wide cellular networks, usually at a considerable financial cost. The NO provides location management functionality for registered MNs within its network, and usually has extensive MoUs with other operators to provide roaming services. In recent years cellular networks have evolved from providing pure circuit-switched voice services, to providing seamless data and voice services using packet-switched technologies. One of the main driving factors is the phenomenal growth of the Internet and the desire to access network services anywhere, anyplace and anytime from a user's terminal. To this end, cellular network operators have made large investments in new spectrum, and have been evolving their existing 2G networks to provide higher data rates, and always-on connectivity to deliver 3G networks. However the rollout of 3G networks has been delayed and in their place network operators have been offering *3G-like* (2.5G) services.

In contrast, the advent of Mobile IP, wireless LANs, and ad hoc networks has created a new type of independent network operator, one that does not require a large financial outlay to quickly build and operate pico- and micro-cellular wireless networks. Such networks offer high-speed access of up to 54Mbps to the Internet. Cellular network operators are not oblivious to this fact and have developed interworking strategies to integrate these new technologies into their existing telecommunications infrastructure. To provide a seamless user experience between these varied systems will require work in a number of areas such as roaming, efficient terminal and user authentication, and integrated billing. Some of these issues are briefly highlighted in the following sections.

2.5.1 Interworking Between 3G and WLAN Systems

3G networks will provide for wide-area network coverage with speeds of up to 2Mbps, whereas WLAN systems give users high-speed access of up to 54Mbps over a geographically smaller area. Integrating the complimentary nature of these two technologies will combine the strengths of each, and result in wide-area system capable of providing users with ubiquitous wireless access [DTNA+03]. However there are a number of challenges to be overcome before this can happen, including seamless vertical handovers across WLAN and 3G radio technologies, security, common authentication, and a single itemized bill for voice and data services offered by the NO and VASPs or "One Stop Billing" [Sal04, KKAM+04].

There are a number of ways in which cellular networks and wireless LANs can be integrated. A simple criterion could be based on the ownership or management of the WLAN, which leads to two configuration scenarios. The first is where the network operator owns and manages the WLAN, while the second scenario is where the WISP or enterprise is the owner. Operator-owned WLANs will allow cellular operators to reduce the load on their limited 3G wireless spectrum by economically offloading data traffic to WLANs in hot spot coverage areas. Additionally, operators have existing authentication and billing mechanisms which they can leverage in the WLAN space [KH03]. From the WISPs perspective, partnering with other WLAN and Cellular operators will give them access to a large customer base and roaming agreements, and will allow them to increase their revenue streams [SSM04].

The 3GPP and 3GPP2 standards bodies have been working on specifications to make WLANs an integral component of their total service offering to their cellular subscribers [AHP03]. Two candidate architectures namely *tightly coupled* and *loosely coupled* interworking have been considered by them [BCHL+03]. These architectures are characterized by the amount of interdependence between the two systems. With tight coupling, the WLAN is connected to the 3G core network SGSN in the same manner as other radio access

networks (RAN/UTRAN), and is considered like any other routing area in the system. All traffic originating in the WLAN passes through the core network before reaching the external PDN. Tight coupling also makes use of 3G mobility management protocols and offers seamless handovers. In addition, tight coupling offers the reuse of 3G AAA procedures and the same level of security in WLAN and 3G environments. Tight coupling is primarily tailored for operator-owned WLANs and cannot easily support third-party WLANs.

With loose coupling, the WLAN is deployed as an access network complementing the 3G networks. The WLAN traffic does not pass through the 3G core network. To reuse 3G authentication procedures, 3G interworking WLAN terminals will require access to the smart card and UMTS Subscriber Identity Module (USIM) applications. This approach also supports integrated billing. The loose coupling approach makes use of IETF protocols for AAA and mobility. Interworking will have to be performed between the AAA client in the access point and the AAA server and HLR in the 3G core network [SFP02]. Mobile IP is required for supporting mobility across domains. However there is no need to introduce 3G entities into the WLAN network, as is required with the tight coupling approach.

2.5.2 Integration of Mobile Internet and Ad Hoc Networks

Recently, issues of how to integrate wireless ad hoc islands with other wireless networks and the core IP network have gained momentum [MRPS04]. In [TSC03], the authors propose integrating mobile ad hoc networks with the existing IETF Mobile IP protocol to create an all-IP wireless environment. They propose extending traditional IEEE 802.11-based APs to incorporate the flexibility of MANETs. This will enable mobile nodes to act as routers to extend the coverage of foreign agents in the network. Unlike the Cellular IP and HAWAII micromobility protocols which restrict mobile nodes to reside within one hop from a base station, the proposed solution allows MNs to be multiple hops from a base station. If a FA crashes, a MN can rely on MANET routing capabilities to connect to neighboring FAs.

A MANET is connected to the Internet via a gateway node which forwards data packets and relays them between the MANET and Internet. To support Mobile IP, each GW also acts the FA and periodically broadcasts *agent_advertisement* messages to announce its services. A MN within the service ranges of multiple GWs can choose the closest one as its default gateway. MANET routing protocols are used to support intra-MANET communications. If a MN does not have a route to a destination within its routing table, it will forward the packet to the local MANET gateway. The GW then uses IP routing to forward the packets to the correct destination node. When a MN roams away from its home network, Mobile IP will be used to forward packets between MANETs [TSC03].

2.6 Summary

Taking into account the discussion in the previous sections, Figure 2-18 presents a more detailed network architectural view of a future integrated mobile telecommunications network compared to the initial view presented in Figure 1-1 of Chapter 1. It can now be seen that extensive interworking will be required between the existing and new entities that will have to be introduced into the network. The figure also shows a scenario where a roaming user moves from a 3G cellular network into a WLAN environment, and eventually becomes part of an ad hoc network. Seamless roaming between these networks will require the Mobile IP and AAA protocols such as RADIUS/Diameter to be employed, in addition to existing telecommunications authentication and accounting mechanisms. Also if existing billing techniques are employed, then numerous CDRs of varying types will be generated by each operator or service provider involved in the call. In some cases the user will require multiple trust relationships and receive separate bills from each operator. As highlighted earlier, the MN is not involved in the CDR generation process and the mobile users must rely solely on the operators to bill them for the correct amount.

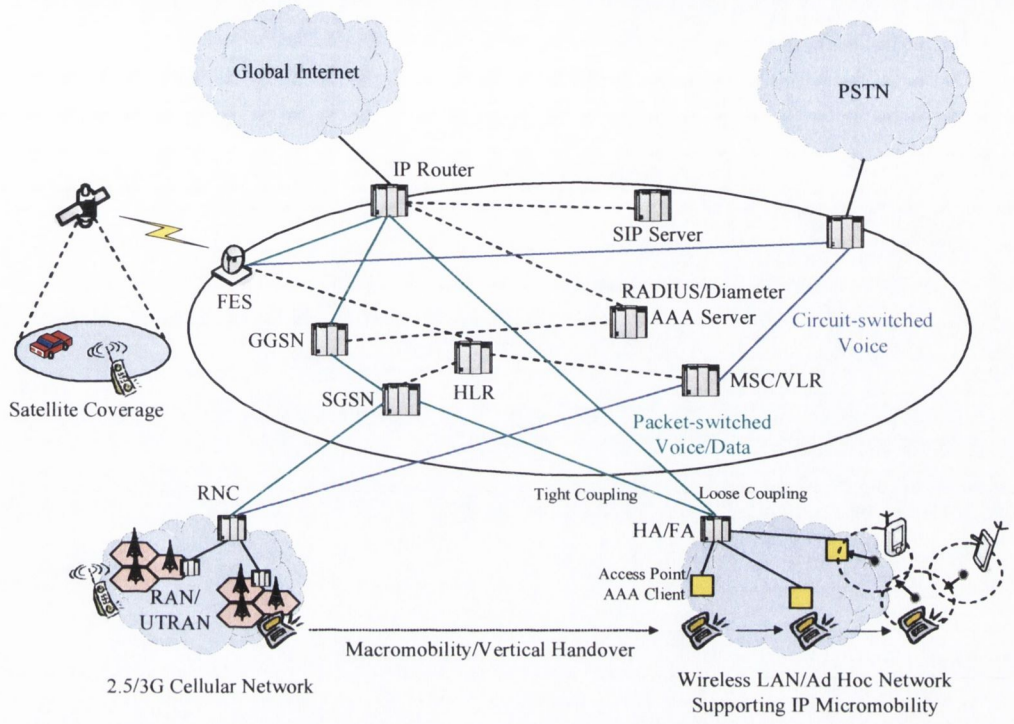


Figure 2-18 Network Architecture View of Next-Generation Mobile Networks

Technologies such as Mobile IP, WLANs and ad hoc networks will play and important part in the overall development of an *all-IP* next-generation network architecture. They will also allow for large numbers of independent network operators and service providers to enter into the telecommunications domain which is presently controlled by a small number of large corporations. How authentication, accounting and billing agreements can be maintained between such large numbers of NOs and VASPs remains an open issue.

The current technological advances in mobile telecommunication networks and the Internet will no doubt alter the existing business models. In traditional billing systems the MN plays no part in the billing process. Such billing systems allow the total amount to be paid afterwards, but cannot ensure payment or provide incontestable charging. In mobile telecommunications networks, user have been mainly billed based on their subscription and call duration, while the charging, billing and accounting schemes used in the Internet have been quite simple or completely non-existent in some cases. From the operator's perspective there is a need to move from simple flat-rate pricing schemes to usage-based models, and to charge end users for the real-value of the service provided. However, the AAA protocols currently being proposed for next-generation networks are complex and heavyweight in nature.

Next-generation networks based on independent operators and service providers will require new lightweight and scalable authentication and billing strategies, which will guarantee payment for services rendered. In the next chapter, existing electronic payments schemes are examined in search of techniques to advance field of efficient authentication and accounting procedures in next-generation networks. Chapter 4 will present a lightweight solution to address the twin problems of authentication and accounting in next-generation mobile networks.

3 Electronic Payment Systems

"Everything ... must be assessed in money; for this enables men always to exchange their services, and so makes society possible."

Aristotle (384 – 322 B.C.)

3.1 Electronic Payments

Payment in its most primitive form involves *barter*: the face-to-face exchange of goods and services for other goods and services between two parties. However this form of payment suffers from the problem that each party must have exactly what the other wants. For example, a person wishing to exchange food for a bicycle, must first find another person who is both hungry and has a spare bicycle. Consequently, over the centuries, bartering arrangements have been replaced with various forms of money. The earliest money was called *commodity money*, where physical commodities such as corn, salt or gold, whose values were well known, were used to effect payment [OPT01].

The next step in the progression of money was the use of tokens such as paper notes, which were backed by deposits of gold and silver held by the note issuer. In highly stable economies, governments are trusted to issue tokens without the need for commodity backing. This form of money is known as *fiat money*, and has value due to the fact that issuing authority is widely trusted. Cash payment is the most popular form of money transfer. Depending on the country, somewhere between 75% and 95% of all transactions are paid in cash, even though the value of these transactions for the most part are quite low [BIS00, AH05]. However with large amounts of cash, security becomes an issue and people start to avail of the services of financial institutions such as banks. If both parties to a payment transaction hold accounts with the same bank, then a payment can be effected by making a transfer of funds from one account to another. This essential mechanism is at the root of a wide variety of payment schemes facilitated by the financial services industry today [AJSW97].

Other forms of payment are cheques, credit transfers or bank giros, payment cards (credit or debit), and now electronic payments. An electronic payment system allows monetary value to be transferred from one entity to another across a computer network. The idea of paying for goods and services electronically is not a new one. However the arrival of the Internet has spurred the development of new technologies and business models to facilitate electronic credit and debit transfers by ordinary consumers. In many cases, these payment systems are the electronic equivalent of traditional payment instruments such as cash, cheques or payment cards, while others represent new forms of value representation and exchange.

Traditional means of payment suffer from a number of security problems. There are large costs are associated with the production, storage and distribution of cash and coins. Bank notes can be counterfeited and signatures on cheques can be forged. Electronic payment systems are also subject to the same security risks as traditional payment systems. In addition, since electronic tokens or instruments are just pieces of digital

data stored on a computer system, they can easily be copied and used repeatedly unless proper security checks are put in place [CS97]. Unlike traditional payment instruments, strong cryptographic algorithms and protocols must be employed to protect electronic payment transactions [Sch96].

The purpose of this chapter is to review the existing payment systems and to provide the reader with an appreciation of the relevant underlying cryptographic techniques. The chapter is organized as follows: Section 3.1.1 presents a generic network payment model, followed by a comparison of macro- and micropayment techniques in Section 3.1.2. A detailed look at some of the more prominent macro- and micropayment schemes takes place in Sections 3.2 and 3.3 respectively. Particular attention is paid to one-way functions and hash chain schemes as alternative lightweight mechanisms for supporting authentication and accounting in next-generation mobile networks.

3.1.1 Network Payment Model

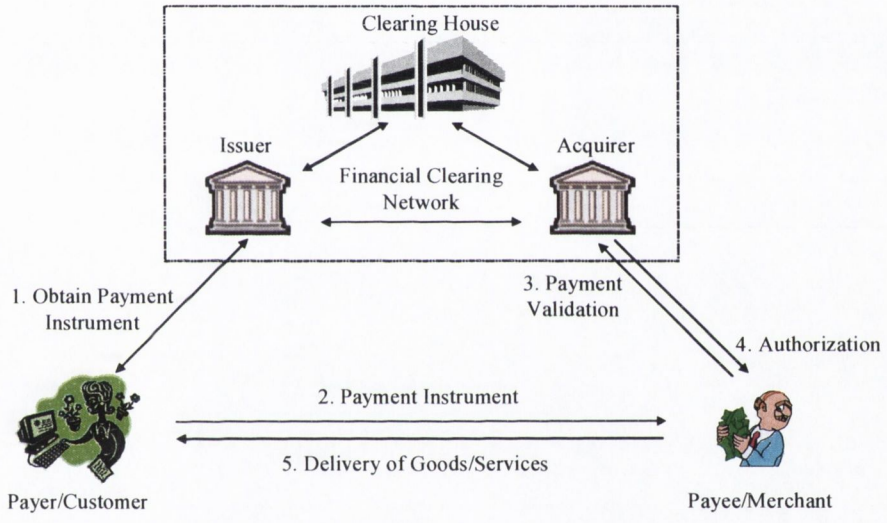


Figure 3-1 Generic Payment Model

Electronic commerce (e-commerce) involves at least three entities, a payer (customer or user), a payee (merchant or vendor) and a financial institution [Pan96]. The last role may further be subdivided into two parts: an *issuer* who issues the payer with the payment instrument or tokens, and an *acquirer* which accepts the payment instrument from the payer on behalf of the payee and authorizes the transaction. The actual flow of money in the system is from the issuer to the acquirer. In some cases there may be a clearing house in the middle connected via dedicated network, such as the Society for Worldwide Interbank Financial Telecommunications (SWIFT) to complete the transaction. The payer and payee are usually connected via an open network such as the Internet. Figure 3-1 gives an overview of message exchanges that take place in a typical electronic payment transaction.

3.1.2 Macropayment and Micropayment

Numerous methods for electronically paying for goods or services across a network have been proposed over the years. These systems are usually electronic equivalents of physical payment methods such as cash, cheques or credit cards. Such electronic payment systems, designed to purchase goods which may range from one dollar to thousands of dollars, are known as *macropayment* systems. Macropayment instruments have a minimum transaction overhead, as they require a real-time dialog with a bank or a Trusted Third Party (TTP)

in the network to verify the authenticity of the payment tokens. In addition, most systems make use of computationally expensive cryptographic operations such as public-key cryptography. These two factors combined, make the use of macropayment systems prohibitive for payments of a few cents or less [OPT01]. An overview of macropayment systems is presented in Section 3.2.

In contrast, *micropayments* are a family of payment systems which have been designed to allow repeated small valued payments e.g. one-tenth or one-hundredth of a cent in a single transaction. Examples of such transactions are, consulting an online encyclopedia, or reading a single news article from the online edition of a newspaper. The low-value per transaction also implies that the cost associated in verifying a micropayment token should be small. Thus a successful micropayment system must not involve computationally expensive cryptographic techniques. Most micropayment research has concentrated on repeated payments at a single vendor. Micropayment systems are examined in detail in Section 3.3.

3.2 Macropayments

Electronic payments systems based on cash, cheques and payment card systems are now examined. The majority of macropayment schemes are used for purchasing high-value goods, and require an online connection with a payment server in the network to authorize each payment. A number of offline schemes such as CAFE [BBC+94] and Mondex [Mondex] have also been proposed, which require no contact with a third party and only involve the payer and the payee. To prevent users from spending more money than they actually possess, offline schemes make use of *tamper-resistant* trusted hardware modules. Such modules are usually supplied in the form of a chip card or smart card. Some payment systems provide user anonymity and untraceability. Many of these schemes make use of heavyweight cryptographic ciphers such as public-key cryptography. The overview of macropayments is concluded by showing how a number of the payment techniques above have been adapted for mobile commerce solutions. Finally, the unsuitability of these schemes for paying for mobile communications and services in next-generation networks is highlighted.

3.2.1 Electronic Cash

There are a number of properties of cash that make it one of the most popular and widely accepted macropayment instruments. One of the reasons why it is so acceptable is that the physical handing over of cash completes the transaction and guarantees payment. In addition, there are no charges incurred in person-to-person transaction and it allows for the indefinite *transferability* of notes and coins. Finally, cash allows for payer *anonymity* and *untraceability*. Attempts to create electronic or digital cash payment methods have focused on subsets of the above attributes.

One of the first pioneering companies to launch an electronic cash payment scheme was DigiCash founded by David Chaum [Cha04]. DigiCash's *eCash* allows consumers to make anonymous payments of any amount on the Internet. It is an online software solution allowing payment for information, hard goods and even pay-out services, where a client might receive back a payment as part of the service. eCash is said to be fully anonymous because clients withdraw coins from a bank in such a way that the bank cannot know the serial numbers of those coins. The coins can be spent anonymously with a merchant, and even collusion between both the bank and merchant will fail to identify the spender [Cha85, Cha92]. Strong security is provided in the system through extensive use of symmetric and public-key cryptography.

Figure 3-2 provides an overview of the process of purchasing and spending eCash coins. Both clients and merchants have accounts at an eCash bank. Clients can withdraw eCash coins against their account and store them in their *cyberwallet* on their computer. The electronic coins used within the eCash system are unique in that they are partly minted by the client, before being signed by the bank using a cryptographic *blind signature* protocol. This is analogous to putting a coin and a piece of carbon paper into an envelope and

sending it to the bank. The bank signs the envelope with its private signature key without opening the envelope. This prevents the bank from seeing the serial numbers on the coins it is issuing. The bank decrements the client's account for the required amount and returns the envelope. The client opens the envelope to obtain the bank-signed eCash coins, which it can use to later pay a merchant for goods or services. On receiving eCash coins, a merchant must forward them to the minting bank to ensure that they have not already been spent. If the coins are valid, they will be deposited to the merchant's account. The merchant will then send the goods and issue a receipt for the transaction.

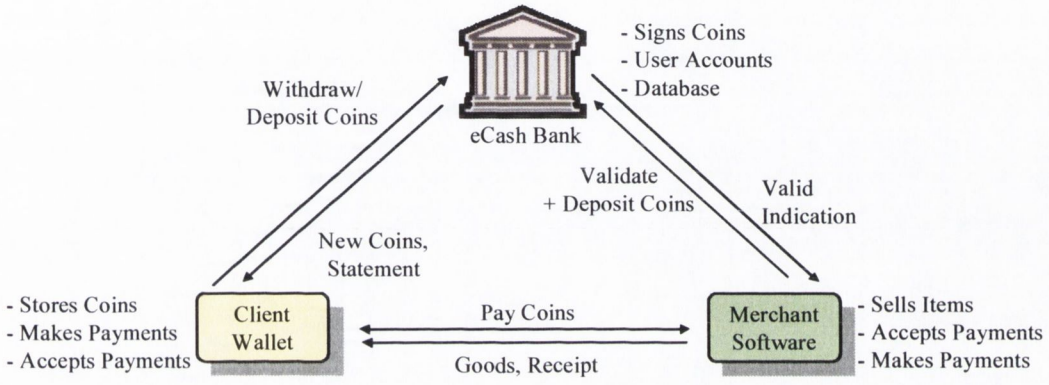


Figure 3-2 eCash Functional Model

To ensure that a coin is not spent twice, the minting bank must record the serial numbers of all coins that are deposited at the bank. This results in the maintenance of a very large database. User-to-user transfer of eCash is also possible, but requires the amount to be forwarded to the bank for verification. There is a fear that anonymous eCash will help hide the identity of criminal and will be used for money laundering, tax evasion and other crimes. For those reasons and others, DigiCash failed to attract a large enough customer base and filed for bankruptcy in November 1998, when the only US bank to use it, Mark Twain Bank, dropped its eCash offering.

CAFE (Conditional Access for Europe) [BBC+94] is another payment scheme based on the idea of untraceable electronic cash by David Chaum [Cha82]. CAFE allows the user to issue electronic coins up to a specified amount, and employs a trusted *observer* module within the user's smart card to ensure the security of the system. CAFE also allows for offline payments, thus reducing the communications overhead of maintaining a communications channel between the merchant and a central database.

Mondex, a subsidiary of MasterCard International is a *cash-like* system which makes use of a prepaid stored value card and allows card-to-card payments [Mondex]. A Mondex card can be inserted into an ATM and used to top-up the card from the user's account using a chip-to-chip dialog. Similarly, retailers equipped with a *value transfer terminal* can accept payment from Mondex users. Little is publicly known about the security features used in Mondex. Public-key technology of some form is used to secure the chip-to-chip dialog [Jon96].

The Mondex scheme is not the only system to use smart cards to effect cash payments. Since 1994 EMV, a consortium consisting of Europay (a group of European card issuing banks) together with MasterCard and Visa have been working on common specifications for integrated circuit cards, terminals to read the cards, and card applications [EMV00a, EMV00b, EMV00c, EMV00d]. In August 2002 MasterCard and Europay International merged into a single organization. The Common Electronic Purse Specification (CEPS) is

another globally interoperable electronic purse system which aims to be compatible with the EMV specifications [CEPS00a, CEPS00b, CEPS00c].

3.2.2 Electronic Cheques

In today's banking world, payment can be made to a third party from a customer's account via the customer's instructions in the form a cheque. Typically, a cheque authenticated with the consumer's signature is presented to the merchant, who may in turn endorse it with his signature, before presenting it to the bank for payment. Though paper-based payments using cheques have been falling, there is still a need for a cheque-like electronic payment system. The handwritten signature can easily be replaced with a cryptographic *digital signature* and the existing inter-bank funds transfer network can be used for clearing and settlement purposes.

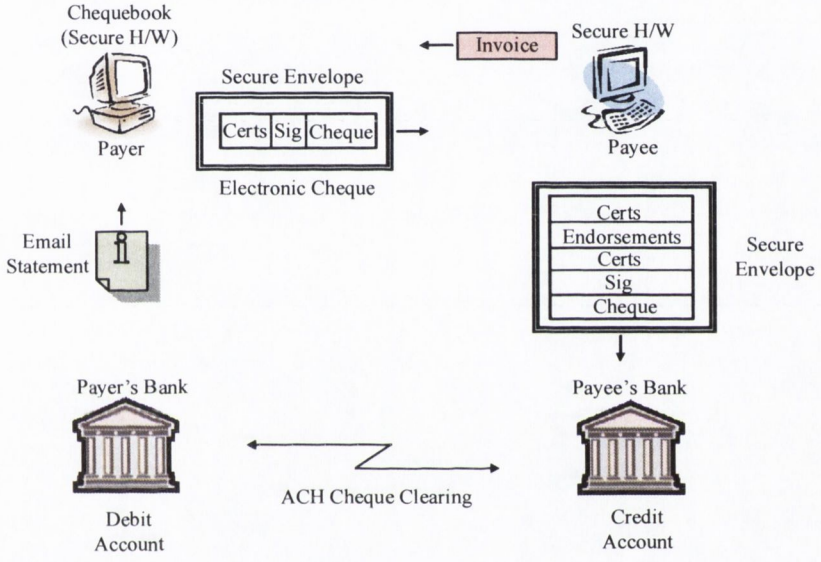


Figure 3-3 FSTC Electronic Cheque Scheme

The Financial Services Technology Consortium (FSTC) in the United States has demonstrated a prototype electronic cheque system that maps directly onto the conventional cheque model [FSTC]. The FSTC electronic cheque scheme is an offline scheme that makes use of tamper-resistant tokens. The payer uses his secure hardware to generate a digitally signed payment instruction or cheque that is transmitted to the merchant along with a digital certificate.

All users in the system are issued with a X.509 certificate by their bank, which is used to verify the payer's digital signature on the cheque. The payee endorses the cheque when it is received, again making use of a secure hardware device, before sending it to his bank. The cheque can then be settled through the existing Automated Clearing House (ACH) network. Figure 3-3 shows the overall cheque settlement process.

When two parties hold bank accounts at two different banks, payment can be made by directly transferring money from the payer's account at one bank to the payee's account at another. When both the payer and the payee hold accounts at the same centralized online financial institution, the transfer between accounts is as simple as subtracting one account and crediting another, without the need for a financial clearing network. This centralized account model has become popular on the Internet with over 20 different payment systems using this approach. Example of such payment systems are PayPal and Yahoo! PayDirect [OPT01].

3.2.3 Payment Card Systems

The first payment cards to be used were *charge cards*, introduced by the Western Union in 1914, and were generally limited to the local market or in-store. In 1958, Bank of America introduced what is today known as the *credit card*. Consumers were no longer tied to a single product or merchant and could make purchases at a wide range of outlets. The market for credit cards is currently dominated by two major players, namely Visa and MasterCard. Charge cards work in a similar fashion to credit cards, with the principal difference being that the entire bill must be settled at the end of the billing period. Finally, *debit cards* such as Maestro from MasterCard are a popular *pay now* product, which allow consumers to access funds in a demand deposit account to conduct a transaction at the Point-Of-Sale (POS). Credit and debit cards are the most widely used means of purchasing goods and services over the Internet [Visa03].

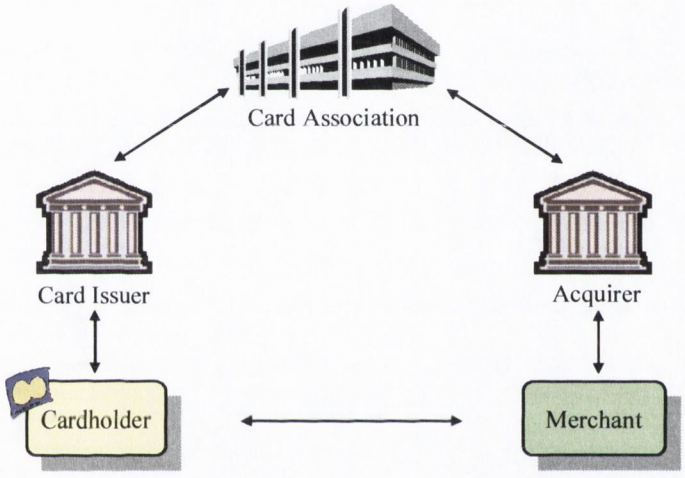


Figure 3-4 Entities Involved in a Credit Card Transaction

The First Virtual system was one of the earliest credit card based payment systems launched for the Internet [EPS1, EPS2]. The system was based on a *try before you buy* philosophy and was not entirely fraudproof. Both merchants and buyers were required to register with the First Virtual (FV) server before any transactions could take place. Buyers had to provide their credit card details and an email address during the registration process, while merchants had to provide their bank details to which any received payments would be lodged.

When a buyer makes a purchase from a FV registered merchant, an email is sent to him to confirm the transaction. The buyer's credit card is billed for the charges that have accumulated at the end of the billing period and the merchant's account is credited. The major advantage of the FV system was its simplicity. It did not employ any cryptographic protocols, so was not subject to export restrictions. However, the First Virtual system ceased operations in July 1998, due to its failure to build a sufficient customer base.

For many years now it has been possible to make payments with credit cards without requiring the buyer and merchant to be co-located. Credit card companies have for some time allowed orders to be taken either by post or by telephone. These orders are referred to as Mail Order/Telephone Order (MOTO) transactions. Using credit cards to make payments across computer networks has similar associated risks as are experienced with MOTO transactions. Attackers eavesdropping on network traffic may intercept messages and capture credit card details. One reason for keeping the credit card number secret from attackers is that, if

compromised, the card number, name, expiry date etc. may be used again by a fraudster to carry out further transactions.

One way around this is to generate a new disposable credit card number each time something is purchased. This system was first brought to the market by Orbiscom in a product called O-card in early 2000 [Orbis]. These generated numbers are from a range allocated globally to the card issuer, but are distinct from those used when issuing real credit cards. The merchant is unaware that there is anything unusual about this and processes the card in the normal way. Once a single transaction has completed, the card issuer will mark the number as invalid and refuse to process further transactions against that credit card number.

An alternative way to transfer credit card details across a network is to use the Secure Sockets Layer (SSL) protocol [KFK96]. SSL is a general purpose protocol designed by Netscape Corporation for encrypting any dialogue taking place between applications communicating across a *socket*, though its primary use to-date has been to enable secure credit card transactions on the World Wide Web (WWW). SSL also allows the merchant to be authenticated in order to prevent attackers from posing as legitimate merchants to capture credit card details. The merchant in an SSL exchange authenticates himself to the client by producing certificates which link his name to a public key. Merchants must apply for a X.509 certificate [ITU, PKIX, NISTPKI] from a public Certification Authority (CA) before engaging in any SSL dialogs. Most popular Web browsers are preconfigured to trust certificates from a number of such authorities. There is no requirement for the card holder to be authenticated. However SSL does have built-in support for mutual authentication of the communicating parties.

The main component of the protocol is the SSL *handshake* which is transparent to the application using it. The handshake protocol is responsible for authenticating communicating peers to each other. It is also entrusted with the job of negotiating encryption and message authentication algorithms along with the required keys. Once the handshake is completed, the two parties share a secret which can be used to construct a secure channel. A typical SSL session makes use of the RSA key exchange algorithm with only the server being authenticated. However the overhead associated with SSL handshake protocol can be quite substantial [APPS00].

The Secure Electronic Transactions (SET) protocol was developed by Visa and MasterCard and is the proposed standard for securing credit card transaction across the Internet. In February 1996, a set of documents specifying the SET protocols were issued [SET97a, SET97b, SET97c]. Each of the parties in SET transaction, with possible exception of the cardholder, is required to authenticate himself at some point in the payment process. In MOTO transactions, the cardholder forwards his credit card details to the merchant who in turn sends them to the acquirer to obtain clearance for the payment. The SET protocol on the other hand makes use of a *dual signature*, that allows both the merchant and the acquirer to independently verify the transaction details, without seeing the others data. Encryption is performed on parts of certain messages which can be selectively revealed to the parties as required. For example, the financial data about a credit card is not revealed to the merchant and data about the purchased product is concealed from the acquirer.

When SET was conceived in 1996, it was expected that it would have achieved widespread usage in a period of around two years. These projections turned out to be wildly optimistic. One of the first problems related to the complexity of the specification. The fact that SET is specified in three weighty volumes means that the software development and testing effort was considerable. Interoperability testing was also a significant hurdle. Another major barrier to rapid rollout was the need for a supporting CA hierarchy. If the certification is to mean anything, strict security policies must be in force to safeguard the certificates and keys over their entire lifetime from issuance to revocation or expiry. All these problems combined together have led to a

failure in the adoption of the SET protocols. As of 2002 the SET standard has been officially abandoned by both Visa and MasterCard International [EComm].

3.2.4 Mobile Commerce Payments

The growth of mobile commerce (m-commerce) follows the increasingly popular ownership and use of mobile personal devices, such as mobile phones and Personal Digital Assistants (PDAs), and the explosive growth of the Internet. Simply put, m-commerce is the ability to conduct business transactions and access services over mobile wireless devices. Users will increasingly initiate a wide range of business and financial transactions, such as online shopping and banking from mobile devices. They will also use their mobile devices for information browsing, such as obtaining weather updates, sport scores, transport schedules, and other services [Sen00, Var02]. In [Kar04], the author cites a number of studies that predict that 118 million Europeans, 145 million Asians, and 22 million Americans intend to use their mobile phones to pay for small purchases in the near future, and that global m-payment transactions will reach $11 billion by 2005.

Two main motivations for using mobile devices for m-commerce transactions are convenience and security. Mobile devices allow users to access services and authorize transactions when paying for goods anywhere, anyplace and anytime. From the merchant's point of view, they can use location-based advertising to target users and increase sales in the form of impulse buying. In addition, the existing security features of mobile devices can be used in m-commerce transactions. For example, mobile phones based on the GSM standard can take advantage of some of the unique security features offered by the technology.

Each GSM phone contains a personalized smart card based Subscriber Identity Module (SIM). The SIM securely holds details identifying the subscriber and his profile, together with a number of encryption keys that are shared with his network operator. When a user inserts the SIM card into a phone handset and enters the appropriate Personal Identification Number (PIN), the SIM activates, authenticates itself to the cellular network and negotiates a session key that is used to encrypt the content of any traffic traveling over the air from then on. This authentication and content protection can be very useful in making payments [Her03].

The authentication procedure allows the NO to link the identity of a user to an account held in the operator's network. For subscribers that have an accounting relationship with the network operator and are periodically billed, this account will have a name, contact details and billing history. For those increasingly numerous subscribers that have prepaid or pay-as-you-go accounts, the account will probably be anonymous and will have a fixed balance remaining to be spent. This puts the network operator in a very good position to act as a banker or mobile transaction provider in the process of making electronic payments. Many network operators already have existing business relationships (MoUs), and this can be further leveraged to make person-to-person payments in the future. Mobile phone operators view their involvement in this process as an important source of income in the years ahead.

One of the simplest forms mobile payments is to use vendor-specific premium numbers to bill the caller's account. Such schemes can be used for services such as dispensing soft drinks from a vending machine. This system in principle can also work with prepaid phones. However, it is not in the interest of competing local mobile operators to negotiate roaming agreements, which would allow their subscribers to use services of their competitors. Hence such systems are restricted in the sense that both the user and the merchant must share the same NO. To address this problem, third-party mobile payment systems which operate independently of any one network operator have emerged. Examples of such payments systems are Sonera MobilePay, GiSMo and the PayBox account based schemes. In such systems both the user and merchant maintain accounts at a central third-party transaction server [OPT01]. The MobilePay and GiSMo systems are however no longer operational. Numerous other mobile payment schemes are cited in [Kar04].

In the PayBox system, to initiate a payment the user informs the merchant of his phone number [PayBox]. The merchant in turn contacts the PayBox server and sends the user's number and the transaction amount. The digits are transferred over the GSM audio channel using the Dual Tone Multi-Frequency (DTMF) tones. The server obtains the merchant's identity using the GSM caller-identification service. The server checks the user's account and calls the user and narrates the transaction details over the voice channel. To confirm payment the user enters his four-digit PayBox PIN, and money is transferred from the user's account to the merchant's account.

The two main protocols used to secure online purchases with credit cards using desktop machines are the Secure Sockets Layer (SSL/TLS), and the Secure Electronics Transactions (SET) protocol. Work has been carried out to adapt these protocols for mobile devices. To this end the Wireless Application Protocol (WAP) forum has developed the Wireless Transport Layer Security (WTLS) protocol, and proposals have also been put forward for a mobile SET protocol [OPT01]. Depending on the capabilities of the mobile device all, some or none of the payment processing may take place on the device. There may be a Wireless Identity Module (WIM) present on the device or it may be integrated into the smart card. A gateway or *wallet server* may be present in the network that will process part of the payment transaction, thus relieving the burden on the mobile device. Figure 3-5 shows the entities involved in a mobile credit card system.

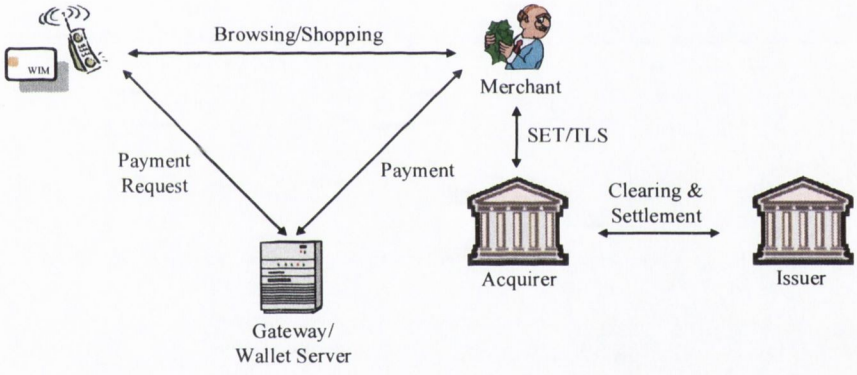


Figure 3-5 Mobile Payments

However, even with these enhancements, many of the same scalability problems and performance issues of traditional macropayment systems remain. There is still a requirement to contact an online third-party for payment authorization or verification for every single transaction. Many of the schemes still make use of expensive public-key cryptographic algorithms to protect the communication links and payment protocol. Furthermore, new delays and problems are introduced by the limited capabilities of the wireless link. In the following sections, a number of micropayment schemes are presented that help to alleviate many of the bottlenecks observed in macropayments schemes.

3.3 Micropayments

Micropayment schemes are used to effect low-value payments, probably less than $1, and in some cases as small as one-hundredth of a cent. Micropayment systems need to be efficient as they are used to make frequent payments, like paying for each tick of a phone call. Given these constraints micropayment techniques must be both inexpensive and fast. Achieving this requires certain compromises, such as relaxing the security requirements. In contrast to macropayment systems, micropayment schemes allow for offline verification in order to reduce the communications overhead, and make use of lightweight cryptographic

protocols to minimize the computational overhead on the user's device. However the amount of effort required by an attacker to defraud the system is still relatively high compared to the value gained, due to the very small amounts involved in each transaction.

The majority of micropayment schemes were designed to purchase small value items, such as reading a few articles from a newspaper on the Internet, playing an online game and so forth. In such scenarios it would not be efficient for customers to buy micropayment tokens from every vendor. In addition, users may not wish to have long-lived account-based relationships with individual merchants or service providers. Thus a number of the micropayment schemes such as Millicent [GMAG+95] and PayWord [RS96] make use of the notion of a *broker*. The broker acts as a middleman to issue and aggregate micropayments on behalf of merchants and service providers that have signed up to accept them [OPT01].

As outlined in Chapter 1, it is envisaged using micropayment techniques not just as a means for paying for low-value goods and services, but also for roaming mobile nodes to pay access network operators in real time for transporting voice and data packets across their networks. In the next section a detail examination of a number of hash chain schemes which can be used to efficiently pay NOs and VASPs in real time for service provision is carried out. In Chapter 4, it is shown how hash chains can also be used to authenticate signaling messages, such as route and location updates in next-generation mobile networks.

3.3.1 Hash Chain Schemes

Lamport [Lam81], proposed the repeated evaluation of a *one-way* function to generate a chain of values allowing many user authentications. A one-way hash function is one where it is easy to compute $y = f(x)$, but computationally expensive to reverse the transaction. A hash or message digest function takes as input a variable length message and often converts it to a smaller fixed-length output. The resulting digest is a strong *digital fingerprint* of the message. A good hash function should also be *strongly collision-free*, which implies that the probability of finding two messages with the same hash should be very low. Finally, even a small change in the input to the hash function should result in a significant change in the output, a phenomenon known as the *avalanche effect*. Examples of some well known hash functions are MD5 [Riv92a] and SHA-1 [NIST95].

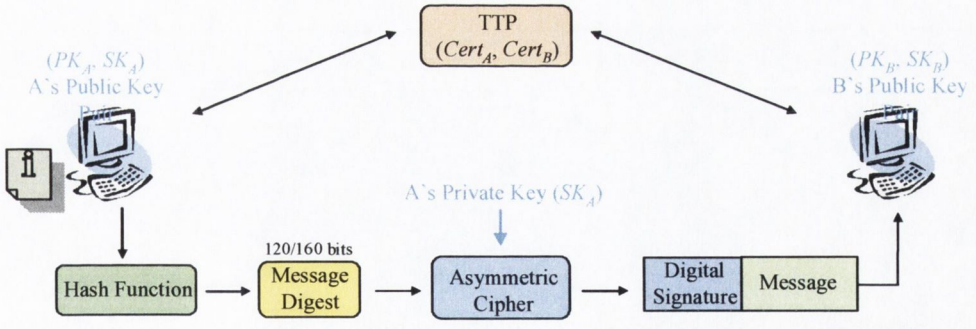


Figure 3-6 Digital Signature Generation using RSA

Hash functions can be used with symmetric block ciphers such as AES [NIST00a] to produce a Message Authentication Code (MAC). Other methods for producing message authentication codes are also possible [RSALab]. Hash functions are primarily used in conjunction with the RSA [RSA78] primitive to produce *digital signatures* as shown in Figure 3-6. In each case, rather than encrypting an arbitrarily large message, the message is first passed through a hash function to produce a 120/160-bit output, which is then encrypted with a shared secret key or a public key. The added advantage of using digital signatures is that they can be

used to link the identity of the sender with the message for *non-repudiation* purposes. The recipient can verify the signature by contacting an online TTP to retrieve the sender's certified public key. In practice the actual message should be padded and redundancy bits should be added prior as specified by the ISO/IEC 9796 signature process [MOV96]. Alternatively one can make use of the Digital Signature Standard (DSS) which has been standardized by NIST [NIST00b].

Public-key algorithms are useful for generating digital signatures and establishing session keys. However they are computationally expensive when compared to symmetric ciphers. Symmetric ciphers on the other hand are useful for bulk encryption. Finally, hash functions such as MD5 and SHA-1 are considered to be computationally more efficient than both symmetric and public-key algorithms. They can be used repeatedly to efficiently generate authenticated payment tokens. In [Mep00], it was shown that hashing is an order magnitude faster than symmetric encryption, three orders of magnitude faster than signature verification and four orders of magnitude faster than signature generation. A number of hash collision attacks were announced at the CRYPTO 2004 conference against the SHA, MD4 and MD5 algorithms. In February of 2005, three Chinese cryptographers showed that SHA-1 is not strongly collision-free [WYY05, NIST05a].

3.3.1.1 Hash Chains

A user generates a hash chain of length n, by applying a one-way hash function n times to a random value P_n, the *root* of the hash chain, to obtain a final hash P_0 the *anchor* of the chain. The user only needs to securely store the root of the hash chain on his device from which the rest of the chain can be recomputed. Since the size of a hash value is only 120 or 160 bits, it can be easily stored on small form-factor devices such as PDAs, mobile phones and smart cards. The anchor of the chain which accounts for no monetary value can be made publicly known to the recipient of the chain.

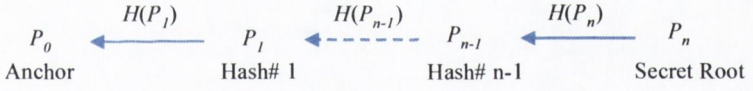


Figure 3-7 Hash Chain Generation

Each hash value in the chain can provide a single user authentication or payment token. For each authentication token the user releases the *pre-image* of the last hash value. For example, for the first token the user releases the hash value P_1. The receiver can apply the same hash function to the value P_1 to obtain the anchor P_0. Since the hash function is one-way only the user could have generated the hash value. Multiple tokens can be sent to the recipient by sending the appropriate hash value further up the chain. For example, in order to send five tokens one can send the hash value P_6. The recipient can recursively apply the hash function five times to the value P_6 to obtain P_1, the last verified hash value.

3.3.1.2 Micropayments Using Hash Chains

Hash values from a user-generated hash chain can be used as authenticated payment tokens. Some of the first hash chain based micropayment schemes were PayWord [RS96] and *i*KP micropayments [HSW96]. Figure 3-8 shows the overall payment process where a user generates a hash chain of length n. The user *commits* to a number of values such as the anchor of the chain P_0, the length of the chain, the value of each hash, and the vendor at which he wishes to spend the chain. Other values such as an expiry date associated with the chain are possible. The signed commitment $Sig_{User}()$ is a digital signature which consists of a message digest of all the above values encrypted with the secret key of the user, along with the original values in plaintext.

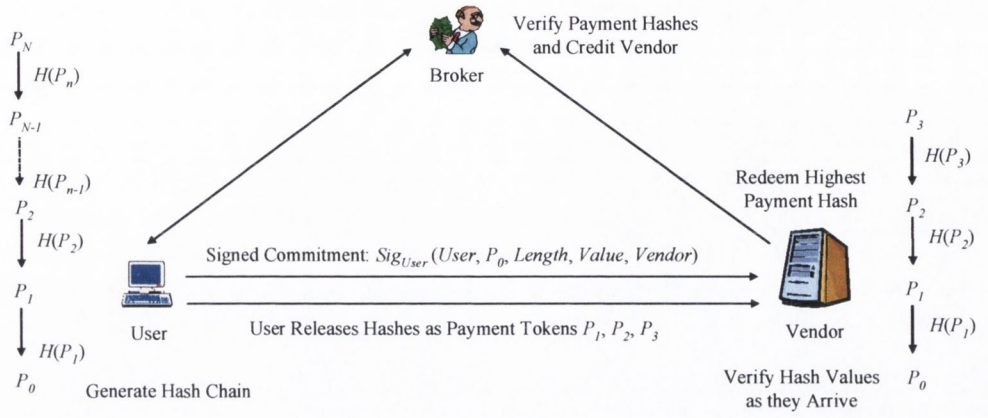


Figure 3-8 Micropayments Using Hash Chains

Prior to the payment, the user forwards the commitment to the vendor, who can verify its authenticity offline. For each micropayment the user releases the appropriate payment hash in the chain. The vendor can redeem the hashes at the broker with whom the user has an account at a later date, by presenting the highest payment hash along with the signed commitment. As is the case with gift vouchers or prepaid phone cards, the values in the chain can only be spent at the designated vendor. If the chain is lost or discarded, then the value associated with it will be forfeited. Unused hash values in a chain can be redeemed at the broker, once the lifetime of the chain has expired.

3.3.1.3 Hash Chain Trees

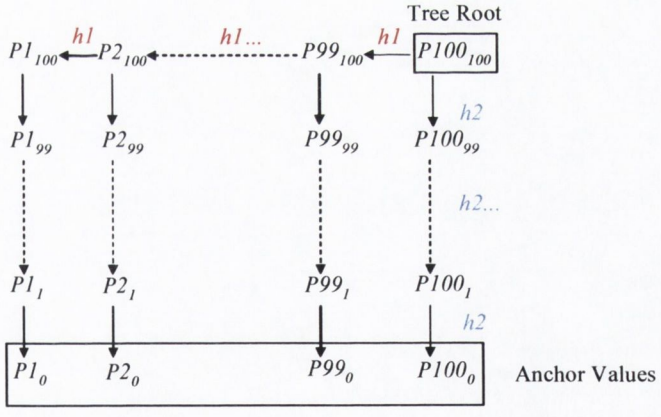


Figure 3-9 UOBT Generation

A number of extensions to the basic hash chain scheme have been proposed, such as the PayTree scheme [OPT01]. These schemes modify the chain structure to produce a tree or graph structure, and allow more efficient storage and computation of large numbers of hash values. One such scheme of particular interest is the Unbalanced One-way Binary Tree (UOBT) scheme [YHH99]. The UOBT is an efficient hash chain scheme, where the root of each chain is derived from another hash chain. The scheme is ideal when a large number of hash values are needed and the host device has limited storage capabilities. Only the *tree root* value has to be stored on the device to be able to reconstruct the entire UOBT. Appendix A of this thesis

provides the reader with a generic UOBT implementation along with the formulae required to compute any node in the tree given the tree root and the size of the tree. Figure 3-9 shows an example UOBT where $P100_{100}$ is the tree root.

The hash function *h1* (e.g. SHA-1) is repeatedly applied to this value to obtain the *backbone hash chain* ($P100_{100}$ ... $P1_{100}$). Each of these hash values is used as the *secret root* value for deriving the individual sub-chains by applying the hash function *h2* (e.g. MD5). For example, the value $P2_{100}$ is the root of the *P2* chain, which consists of the values $P2_{100}$, $P2_{99}$ and so on until the anchor of that chain $P2_0$ is reached. The result is a UOBT with a backbone chain length of 100, with each sub-chain also consisting of a 100 hash values and an overall hash chain tree consisting of 10,000 hashes. The signed commitment consists of a hash of the concatenation of the anchors of each of the sub-chains signed with the private key of the user or broker, to produce $Sig_{User/Broker}(P1_0, P2_0 ... P99_0, P100_0)$.

On small devices with limited storage such as a PDA, it may not be possible to store all the hash values of a long hash chain. On average the number of hashes performed for a chain length of n is $(n - 1)/2$. Therefore if a user spends a hash chain of length 10,000, the average computational overhead per payment (assuming no caching of intermediate values) excluding the initial signature, will be 4,999.5 hashes. If on the other hand a UOBT is used with a backbone chain length equal to the length of each sub-chain, it can be shown that the average computational overhead to compute the next hash is $n^{1/2} - 1$, where n is the number of values in the UOBT and $n^{1/2}$ is the square root of n. This gives an efficiency improvement from $O(n)$ to $O(n^{1/2})$. For example a 100x100 UOBT requires 99 hashes on average to compute a hash value. There is an initial overhead in transporting the set of anchors to the broker for the commitment to be signed. Also the merchant is required to store the set of anchors for verification purposes. However once the initial exchange has taken place, the number of cryptographic operations performed during the communications session are greatly reduced.

3.3.1.4 Optimal Hash Sequence Traversal

To compute the next value in a hash chain one of two techniques can be applied. The first is to compute the required hash value starting from the root of the chain each time. Alternatively one can pre-compute all the values in the chain and store them in memory. Both of these techniques require memory and computational complexity of $O(n)$. However for limited capability mobile devices, the generation or storage of very large hash chains becomes a problem.

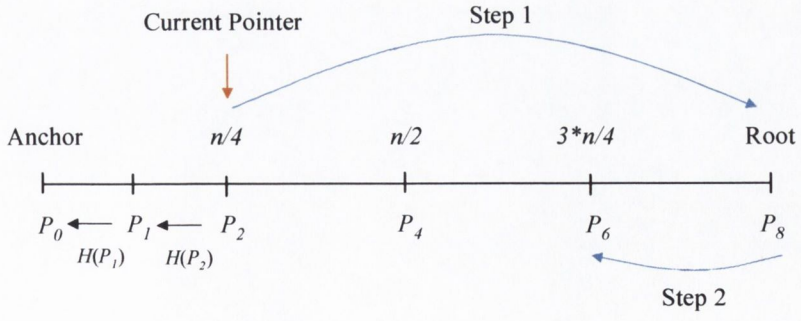


Figure 3-10 Optimal Hash Sequence Traversal

Jakobsson [Jak02] presents an optimal hash sequence traversal scheme which has an upper bound of memory and computational complexity of $O(log\ n)$. The user pre-computes all the values in the hash chain starting

from the root prior to using them on a powerful machine such as a desktop computer. However, the user is only required to store a number of strategic hash values or *pebbles* in the chain for later use on the mobile device, where a pebble stores the hash chain value for a position it is associated with. The location of the pebbles is modified over time. More specifically the proposed algorithm performs at most $log_2 n$ hash functions for each value that it outputs in the chain, where n is the total length of the chain. Also the algorithm requires at most $log_2 n$ memory locations of size $2log_2 n + 160$ bits, where a memory location consists of a hash chain value assuming SHA-1 and some state information. A chain of length 2^{32}, which requires 32 storage cells of size 224*32 bits or 896 bytes, will last for more than 68 years, if a hash value is released once per second.

Figure 3-10 illustrates an example of optimally outputting values in a hash chain of length $n = 8$, where the first value to be calculated is P_1. As a starting position the algorithm calculates and stores three pebbles. The first is the root of the chain at n and the second is the mid-point of the chain at $n/2$. The final value is located in the middle of the first interval at $n/4$. It can be seen that the maximum computational effort required calculating the next hash value to be outputted during the first $n/2$ rounds is $n/4$. After the current pointer reaches the first pebble at $n/4$ the algorithm moves it to the root of the chain in Step 1. The pebble is then gradually moved towards its ultimate destination of $3*n/4$ in Step 2, where each move costs one hash function evaluation. If it reaches its destination by the time the current pointer reaches $n/2$ the maximum computation effort required to output the remaining values remains at $n/4$. By adding more pebbles the computational costs can be further reduced. A more efficient version of the algorithm is presented in [CJ02], which halves the computational costs to $O(log\ n^{1/2})$ and further reduces the memory requirements. Unlike the UOBT scheme, only the anchor of the chain needs to be sent to the broker for the commitment to be signed.

3.4 Summary

Micropayment research has mainly focused on paying for low-value goods and services on the Internet. However micropayments are also seen as an enabler for efficient authentication of signaling messages and real-time payment for datagram transport in next-generation mobile networks. The main contributions of this thesis, namely three authentication and accounting schemes for next-generation mobile networks are presented in the next three chapters. In each chapter the system model and the main protocol goals are outlined. This is followed with the protocol design and implementation details. Finally, the experimental results are presented along with a summary.

Specifically, in Chapter 4, the use of hash chains is employed to develop a real-time authentication and accounting scheme for Mobile IP based micromobility access networks. The proposal eliminates the need for a mobile node to have a subscription or accounting relationship with a home network operator. In Chapter 5, the basic payment protocol is extended into a real-time multi-party micropayment scheme for packet forwarding in ad hoc networks. In both these schemes, a UOBT is employed for the efficient storage of hash values on the mobile device. Finally in Chapter 6, the use of hash chains employing the optimal hash sequence traversal technique by Jakobsson is used to provide authenticated routing and packet forwarding in ad hoc networks.

4 Accounting for Network Services in the Mobile Internet

"The best way to predict the future is to invent it."
Alan Kay

4.1 Introduction

The Internet has evolved from its humble beginnings of just four nodes comprising the ARPANET in 1969, into what is undoubtedly today the largest global communications network, with more than 300 million hosts [ISC04] and in excess of half a billion users worldwide [IWS04]. Today the Internet comprises some 50,000 regional, national and international networks, which are connected to the core network via dedicated high-speed links. The design of the Internet is such that it consists mainly of closed user groups, where nodes and users have long-lived trust relationships with the network operator. The accounting procedures provided in these networks are usually very limited and in some cases non-existent.

The Internet was originally designed as a network to cater for non real-time applications and fixed stationary nodes. However, today the Internet has evolved into a network that now has support for real-time applications, limited QoS, and host mobility using the Mobile IP protocol [Per02a, Per02b]. The Mobile IP protocol in turn has evolved from providing mobility support for portable computers, to support for wireless handheld devices with high mobility patterns. A new category of micromobility protocols has been proposed to deal with the increased signaling loads that will be generated with large populations of such devices on the network [CGKW+02, DMDM+02, GEN01].

The huge influence of the Internet and its associated applications on global communications has been such, that telecommunications operators are rushing to provide seamless access to the Internet through their existing and next-generation cellular networks. Chapter 2 investigated the various approaches that are being adopted to incorporate Internet technologies into existing 2.5G and 3G networks. The impact of such data services on CDR generation and inter-operator billing strategies was examined. The concept of an all-IP next-generation network was also highlighted and how this would influence on the overall network infrastructure. With the availability of low-cost wireless hardware anyone with a wireless access point has the potential to become a network operator. For roaming to occur across these networks, a potentially large number of inter-operator billing agreements will be required. Consequently the billing methods of Chapter 2 with their implicit trust relationships become increasingly inadequate in such environments.

It is envisaged that next-generation all-IP networks will consist of large numbers of independent NOs and VASPs, who will charge users for access to services in the fixed network, but may not necessarily wish to maintain long-term contractual relationships with them. These access networks operators may also employ a variety of micromobility protocols for fast handover support within their networks. With large numbers of

roaming users, NOs and VASPs, it is no longer safe to assume a trust relationship between them to guarantee payment for usage of network resources. Instead a real-time payment solution that pays the access network operator in real time for packet delivery is proposed. This will eliminate the need for a MN to maintain a subscription with a home network operator for location management purposes. From the NOs point of view, the proposed solution eliminates the need to maintain trust relationships with individual users and the overhead associated with postfact billing. It also eliminates the need for complex MoUs between the NOs and VASPs [TO03a].

In Chapter 3, a survey of the state-of-the-art in electronic payment systems was carried out. It was found that traditional payments systems were not adequate for mobile environments due to their dependence on an online connection to authorize the payment instrument, and their use of heavyweight cryptographic primitives. A new category of micropayment protocols were introduced which allows the recipient to verify payment tokens offline and as they are received. Thus the real-time payment protocol for next-generation networks proposed in this chapter will make extensive use of micropayments based on hash chains.

In Section 4.2, the entities in the system model are introduced by presenting an example scenario where a mobile user roams into a new access network. The requirements for a protocol which aims to eliminate the shortcomings of present authentication and accounting strategies are outlined in Section 4.3. In Section 4.4, the various aspects of the proposed authentication and accounting protocol are described in detail. Section 4.5 presents the implementation details, followed by the experimental setup and results in Section 4.6.

4.2 System Model

A survey of existing telecommunications based mobile networks and their evolution towards next-generation networks in Chapter 2 highlighted the complexity that will be introduced into the network entities in the system in order to provide IP-based services. It was also noted that with the advent of cheap wireless technologies such as WLANs, many of the barriers to becoming an independent network operator had been dismantled. The Internet as it stands today is highly optimized for data services. It however lacks robust support for mobility, QoS and AAA services. There are presently a number of initiatives underway to address these problems. In particular, the Mobile IP protocol and the various micromobility architectures will address the problem of seamless mobility on the Internet. In contrast to the approach adopted by the of the telecommunications operators, which propose migrating the existing core network entities to support IP-based services, the protocol proposed in this chapter makes use of lightweight authentication and accounting procedures to transform the existing Internet infrastructure into a mobile communications network. Thus allowing anyone with the appropriate hardware and high-speed link into the Internet to become a network operator, and generate revenue for services rendered.

The system model envisages that in next-generation networks there will be a large number of independent NOs, who will provide users with wireless access to other fixed or mobile nodes on the network. These access networks will have dedicated high-speed connections via a Gateway router (GW) into the core IP network. Small or medium sized networks may consist of a number of radio cells and may support micromobility protocols for efficient handover within the access network. Each user in the system will be uniquely identifiable by his NAI which is of the form user@realm [Abo99].

A user who arrives in a new access network can immediately start availing of services in the core network, once he has registered himself with the NO, and bought operator specific payment tokens from a trusted Broker (BK). In order to be able to receive incoming calls a user must register his Care-of Address (CoA) with a Location Management Server (LS). The LS keeps a mapping of the user's NAI to his current CoA. Figure 4-1 depicts the scenario where the MN (usr1@ls2.net) moves from the domain wisp.airport.com to the

domain wisp.motel.com, which results in an update at his preferred location server (LS2). A correspondent node (joe@ls1.net) can query LS2 to obtain the MNs current CoA in order to communicate with him.

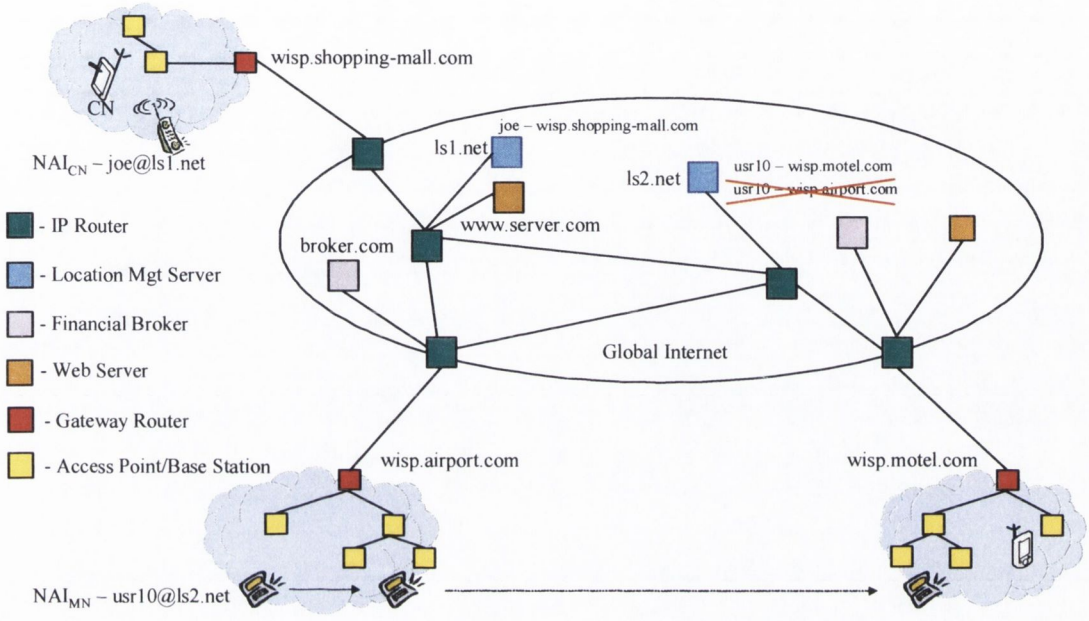


Figure 4-1 Network Model

A NO generates revenue by charging for usage of network resources from roaming mobiles and may also provide other value-added services. Existing credit based mobile billing systems trust users to pay their bills based on strong identity verification and credit history checks. They also place total trust in the operator to bill the user for the correct amount but provide no mechanisms to prove the authenticity of the CDRs generated by the operator. Unlimited credit with postfact punishment is too open to abuse in mobile networks. With a large population of mobile users and independent NOs, it is desirable to remove the need to trust them and thereby minimize fraud in the system. Micropayment technology is employed which allows the routers within the access network to authenticate datagrams prior to making a routing decision, and the NO to be paid in real time for service provision. This eliminates the need for any long-lived contracts between the MN and the operators.

A roaming user registers with the NO to obtain a unique network address for the access network that he currently finds himself in. This address serves as the new CoA for the user while he remains in the domain of the access network. The user subsequently purchases operator specific payment tokens from his broker via the gateway node as payment for using resources in the access network. The GW authenticates the broker commitment and broadcasts the payment details to the routers within the access network. In order to be able to receive incoming calls, the MN sends an authenticated update of its new care-of address to its chosen location management server.

The MN also periodically attaches the correct number of payment tokens that the LS requires to keep the CoA information alive within its database. In addition to the existing fixed servers in the network, there is a new category of mobile users, and thus data calls are handled differently from voice calls. In the case of a data call, the MN obtains the IP address of the destination node by querying a Domain Name Server (DNS), and forwards datagrams towards the destination via the GW node in the access network. For voice calls, the

MN first pays its location management server to obtain the current CoA for the correspondent node and then forwards datagrams to the CN via the gateway.

In each case, the MN attaches the required amount of payment tokens along with each datagram to enable them to be transported through the access network. The hash values also perform an authentication function, and are used by the routing nodes within the access network to update their *soft-state* route mappings for roaming mobile nodes. The payment parameters are removed by the gateway node prior to releasing the datagrams into the core network. The service provider periodically deposits the highest payment hash in each chain used by a MN with its broker to redeem payment for network usage.

4.3 Protocol Goals

The deficiencies in the billing procedures in existing mobile networks have been outlined. A vision of the architecture of future Mobile IP based access networks has also been presented. The design goals of the proposed authentication and accounting solution for Mobile IP based next-generation networks are outlined below:

Real-Time Payment – A mobile user should be able to pay in real time for network usage in an access network without the need to contact a home network operator. Further, if the location management functionality can be provided as a paid service by a VASP, then subscription with a home network operator can be completely removed. By removing the need for subscription based billing and location management, the requirement for a *home network* can be eliminated.

Authentication and Router Updates – Users should not have to maintain a separate security relationship with each access network that they may use from time to time. In addition, signaling messages transmitted by a mobile in order to add or update the soft-state routing entries within the routers in the access network must be quickly and efficiently authenticated. This should be done without the need for extensive key exchanges between the various entities or the need to contact a third party.

Lightweight Cryptographic Procedures – Authentication and accounting related cryptographic data that accompanies datagrams in the access network should be kept to a minimum. This will lead to fast processing of datagrams prior to routing and minimize storage requirements in intermediate nodes. Nodes outside of the access network should not have to understand the mobility or payment messages. Packets that traverse the core network should appear as normal IP datagrams to intermediate routing nodes.

Offline Payment Verification – There should be no requirement to maintain an online connection to a third party to verify the validity of payment tokens by an accepting entity. The payee should be able to present the payment tokens at a later date to a broker and be guaranteed payment. There should be multiple brokers in the system which allows payment tokens to be redeemable at the recipient's broker, who in turn trusts the issuing broker.

Identified Payment Tokens – The payment tokens should redeemable only by the specified entity. This will prevent double-spending of tokens by cheating nodes or eavesdroppers in the network.

Personal Mobility – A user should be reachable using a globally unique identifier and should be able to obtain an access network specific address on demand. The payment tokens and associated security keys for a user can be stored on a smart card. This allows them to be used in any mobile device owned or rented by the user. However the smart card should be used purely as a secure portable device, and there should be no dependency on the cryptographic hardware to maintain the security of the system.

Location Privacy – The location of a mobile user should be known to as few entities as possible in the system. One should be required to pay a location management server to obtain the current CoA for a user. This will minimize the number of unsolicited messages being sent to a node.

In brief, the proposed solution for next-generation mobile networks wishes to remove unnecessary trust from the system for roaming and billing purposes, reduce the online communications overhead of contacting a home network, and allow real-time payment for usage of network resources anywhere by anyone who holds valid payment tokens.

4.4 Protocol Design

In the following subsections the various procedures which form part of the overall authentication and accounting process for the proposed system model are discussed. This section begins by detailing the roles of the various entities in the system, the requirements for each entity in order to engage in the protocol, and any assumptions that have been made in order for the correct operation of the protocol.

4.4.1 Roles, Requirements and Assumptions

The main entities involved in the various protocol exchanges are the mobile and correspondent nodes, the base stations, routers and gateway node which are part of the access network operator's infrastructure, and the financial brokers and location management servers. The roles and requirements for each of the above entities are outlined below:

Mobile Node – Each MN in the system is required to generate or securely obtain a public key pair and to subsequently obtain a public-key certificate for the same from a Trusted Third Party (TTP) in the network, along with the public-key certificate of the TTP. The Brokers or Location Management Servers in the network can perform such a function. The public-key pair is used for performing encryption, decryption and digital signature operations. The above implies that the MN must be able to efficiently perform public-key operations. The same requirements apply to CNs in the system.

Access Network Operators Infrastructure – The network operator's infrastructure consists of Base Stations, Routers and a Gateway Node. Each of these fixed network entities maintains a Authentication Cache (AuthCache) which is used to store a number of parameters for each active MN in the access network. The GW node is required to obtain a public-key certificate from a TTP and must be able to perform encryption, decryption and digital signature functions. The base stations and routers in the access network are required to possess the public-key certificate of the GW and must be able to efficiently perform hash operations using any of the well known hash functions such as MD5 and SHA-1, as well as public-key operations using the RSA algorithm. The reader is directed to Appendix A of this thesis for more details.

Location Management Server – The LSs in the network maintain location management information for registered MNs. A MN is required to pay the LS for the privilege of maintaining its current CoA. A LS also possess a public-key pair and obtains a public-key certificate from a TTP in the network. Location management servers can also perform the role of a TTP for MNs in the system. A LS will also possess the public-key certificate of TTPs higher up in the trust hierarchy. A LS must be able to perform public-key encryption, decryption and digital signatures operations using the RSA algorithm. It must also be able to perform hash function operations using the MD5 and SHA-1 algorithms.

Broker – Financial brokers in the system are trusted entities that are required to generate broker signed commitments which certify MN generated hash chains. The signed broker commitment allows the GW node

in the access network to be confident that the MN is presenting valid payment tokens and that it will subsequently receive payment for network provision. A broker must possess a public-key pair and obtain a public-key certificate for the same from a TTP. It must be able to perform public-key encryption, decryption and digital signature operations. It must also be able to perform hash operations. A broker can also perform the role of a TTP to certify the public key of a MN in the network.

The protocol assumes that there is a globally trusted Public Key Infrastructure (PKI) and associated hierarchy of Certification Authorities (CAs) in place. The top level CAs in the system will certify entities such as the brokers, location management servers and the network operators in the system. The BKs and LSs in turn can certify individual users. It is assumed that these CAs are highly trusted entities and implement strict security policies in relation to storing root keys, payment information and certifying individuals or entities in the system.

4.4.2 Registration in the Access Network

On entering or waking up in a new access network, a mobile node's first task is to obtain a network specific address. It first obtains the address of the gateway node which is periodically advertised (along with the public-key certificate of the GW node) by the base stations in the network and sends a Mobile IP Registration-Request (RREQ) to it. The MN identifies itself by attaching a digitally signed copy of its NAI and nonce, and uses a null IP address as its initial source address [Per02a]. The RREQ is encrypted with the public key of the GW node to prevent unauthorized access to the message details across the air interface and is of the general form $(0, IP_{GW_{ID}}, Registration\text{-}Request, PK_{GW_{ID}}(Sig_{USR_{ID}}(NAI_{MN}, NONCE), Cert_{USR_{ID}})$. A mobile node is issued with a NAI when it establishes an association with a location management server. It may additionally get its public key certified by the LS or any other widely known TTP.

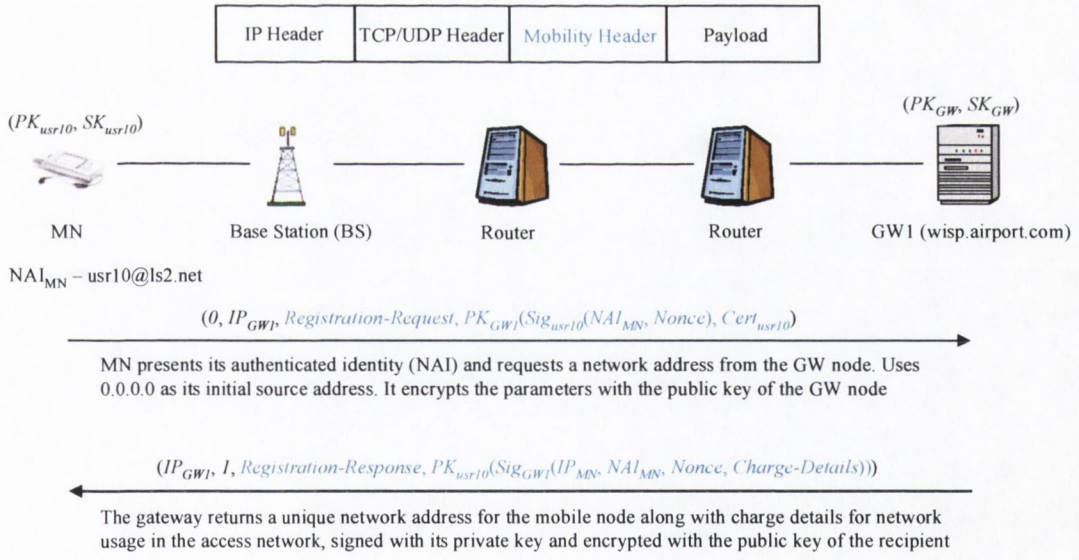


Figure 4-2 Registration Procedure and Datagram Format

The GW returns a unique network specific address along with its charge details for network usage, signed with its private key in the Registration-Response (RREP) message. The RREP is encrypted with the public key of the MN to prevent the details from being overheard across the air interface and is of the general form $(IP_{GW_{ID}}, 255.255.255.255, Registration\text{-}Response, PK_{USR_{ID}}(Sig_{GW_{ID}}(IP_{MN}, NAI_{MN}, NONCE, Charge\text{-}Details))$.

The charge details at a very minimum may specify the charge for carrying data and voice traffic to and from the core network. Figure 4-2 shows the datagram format which contains an additional header that is referred to as the *mobility header*, and the message exchanges between the mobile and gateway nodes. The mobility and payment related fields are kept within the mobility header and are required only by the nodes within the access network. The mobility header is removed by the gateway node prior to any datagrams leaving the access network.

4.4.3 Payment Chain Purchase

Prior to making any calls, a mobile node must purchase access network specific payment tokens from a broker whom the network operator must also trust. The MN generates a UOBT of the desired length from a secret tree root to obtain for example the set of anchors ($P1_0$ ... $P100_0$). The secret roots of the UOBT from which the individual sub-chains are generated do not leave the users device during the chain purchase protocol. The payment chains of the UOBT can only be spent at the specified NO and have no monetary value until committed to by a broker. To obtain this commitment, the MN attaches a macropayment for the broker in the Purchase-Request message. The request is sent to the gateway node in the access network which forwards it to the identified broker (BK) without charge and monitors the reply.

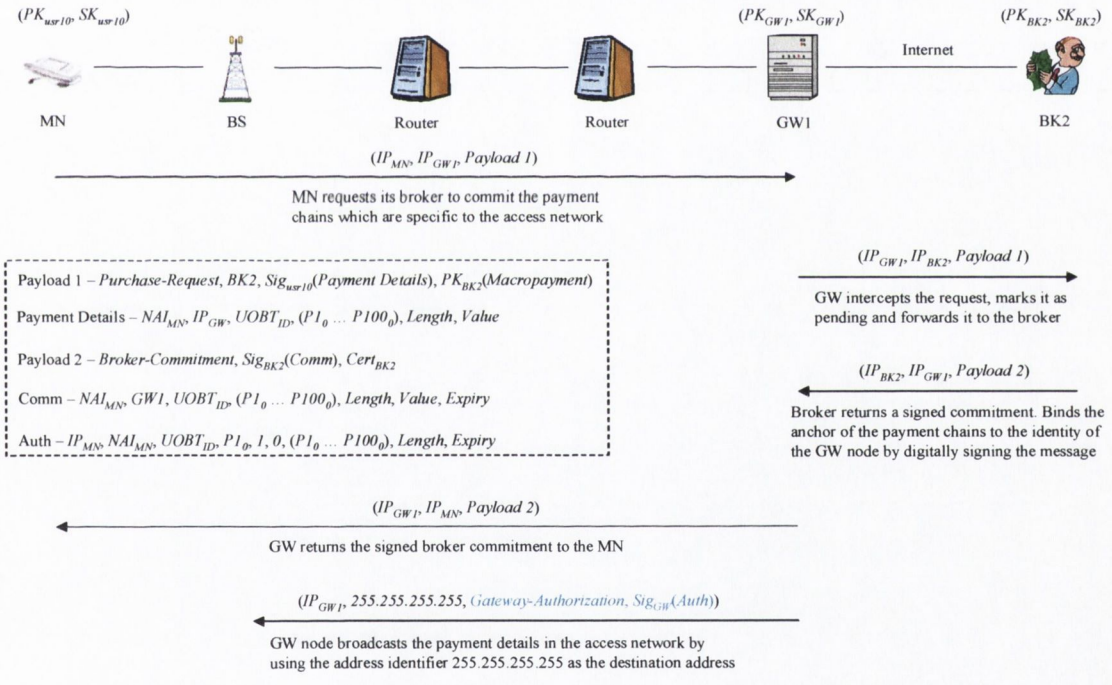


Figure 4-3 Payment Chain Purchase

The message consists of the payment details such as the NAI of the user, the identity of the NO where the chains are to be spent, the UOBT identifier along with the set of anchors, the length of each chain and the value of a payment token in the chain. The payment details are signed with the private key of the user and are of the general form $Sig_{USR_{ID}}(NAI_{MN}, IP_{GW}, UOBT_{ID}, (PX_0 \ldots PY_0), Length, Value)$, the authenticity of which can be verified by the broker. PX_0 ... PY_0 are the anchors of the sub-chains of the UOBT. The macropayment is encrypted with the public key of the broker $PK_{BK}(Macropayment)$, which ensures that no unauthorized

entity is able to access the payment details during transit. Figure 4- 3 shows the payment chain purchase protocol in detail.

The broker commits to the hash chains or promises to honor their value by digitally signing the payment chain commitment *Comm*(), consisting of the payment details sent by the user and an expiry date associated with the chains. The signed broker commitment is of the general form $Sig_{BK_{ID}}(NAI_{MN}, GW_{ID}, UOBT_{ID}, (PX_0 \ldots PY_0), Length, Value, Expiry)$. The commitment shows that each payment hash from the chains represents a prepaid value redeemable at the broker. The commitment is returned to the user via the gateway node in the access network. The GW verifies the broker commitment and broadcasts the relevant payment details *Auth* to all internal routing nodes in the access network.

The payment details are stored in an Authentication Cache (AuthCache) in each router in the access network. Hash tokens are released subsequently to the network operator by the MN and serve as payment throughout the duration of a call. Since the hash tokens are generated using a one-way function they can serve the dual purpose of authentication. They are used by the routers in the access network to authenticate signaling messages and datagrams prior to updating their route cache entry for a mobile node. If the MN already possesses valid payment tokens for the access network, it may present these to the gateway node which can authenticate the same and broadcast the relevant details to the routing nodes within the access network.

4.4.4 Location Management

In order to receive incoming calls, a user must periodically update his care-of address details at his designated location management server. Other users in the network wishing to contact the user can identify the serving LS from the user's NAI, and obtain the current care-of address for the user. Figure 4-4 shows the sequence of steps involved in the process. The MN sends a message to its location server via the gateway node in the access network. The mobility header contains a payload identifier (Data), the UOBT identifier ($UOBT_{ID}$), a hash token (PX_Y) as payment for network usage in the access network, the sub-chain number X and the hash number Y.

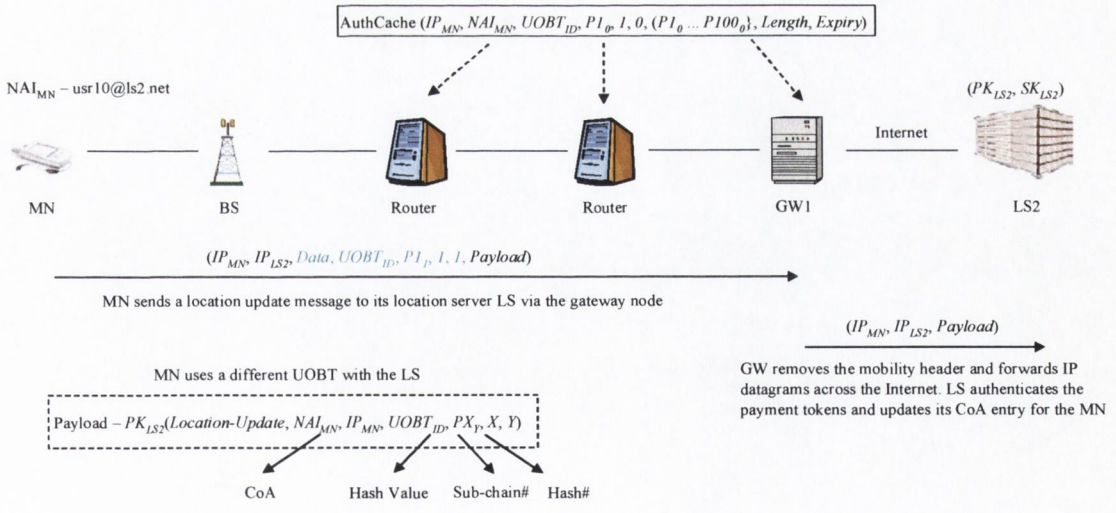


Figure 4-4 Updating the Care-of Address

Intermediate routing nodes are able to verify the authenticity of the hash token by hashing back to the last hash value stored in their authentication cache. A valid hash value allows them to add to or update both their

authentication and route caches. The AuthCache always contains the last valid hash token that was received by the node. This aids in fast verification of the hash tokens, as a node is not required to hash back to the anchor of the chain each time. The gateway node removes the mobility header and forwards the payload to the specified destination address. Note that the datagram appears as a regular IP datagram to routers in the core network. The contents of the payload are decrypted by LS2 to reveal a Location-Update message which contains the new CoA for the user. The location update message is of the general form $PK_{LS_{ID}}(Location\text{-}Update, NAI_{MN}, IP_{MN}, UOBT_{ID}, PX_Y, X, Y)$, where PX_Y is the hash value, X is the chain number and Y is the hash number. The user also attaches the required hash tokens as payment to the LS for hosting this information. The payment tokens are from a separate UOBT that the user purchased previously in order to spend at the LS, and are also used to authenticate the update request.

The requirement to store and verify the hash values in intermediate nodes in the access network is not mandatory if the NO is willing to update the soft-state routing entries without authenticating the same. It is however a requirement to capture the hash details at the GW node for accounting purposes. If the use of the optimal hash sequence traversal scheme outlined in Chapter 3 is employed, then the storage requirements at each intermediate node would only be 2*32 bytes (the anchor and last hash token in the chain), assuming that the SHA-256 algorithm is used. Thus it is recommended to make use of an AuthCache in all routers in the access network.

4.4.5 Call Delivery – Data

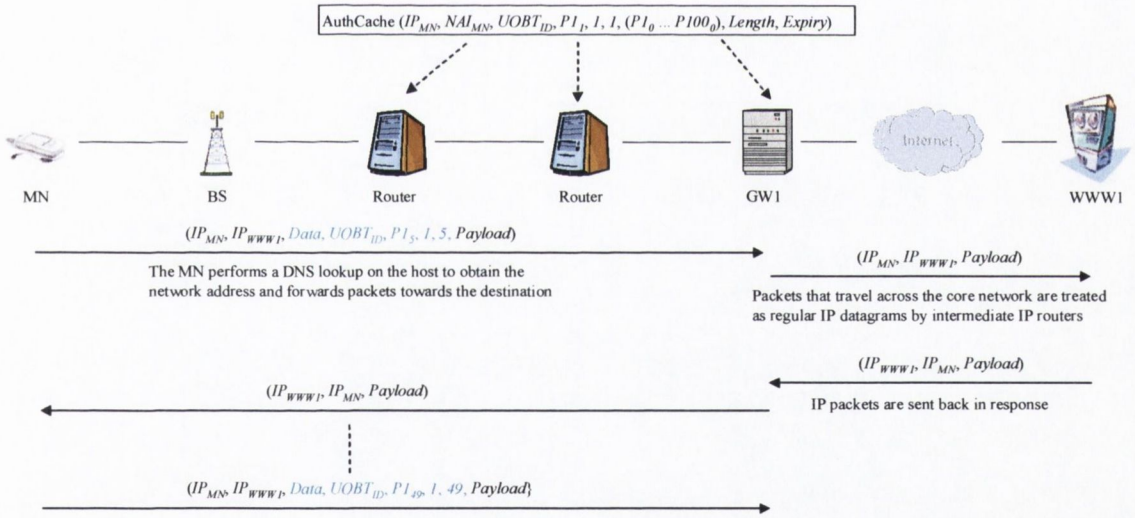


Figure 4-5 Accessing a Web Server

From time-to-time a node may wish to contact other fixed nodes in the core network, such as a Web server or a VASP. The address of such servers is well known and remains unchanged over a period of time. The MN may query a DNS to obtain the IP address of the server and may need to pay the NO for the same. Once the MN has the network address of the server, it attaches the required payment tokens along with each datagram, for them to be transported through the access network towards the destination. At a very minimum, the charge details could be a fixed value that is supplied by the GW node when the MN first registers in the access network. However in practice, more sophisticated flexible charging schemes may be required by the network operator. Intermediate routing nodes on the path to the GW can authenticate the hash tokens, by hashing back to the last stored hash value in their authentication cache.

The gateway as before removes the mobility header prior to forwarding the datagrams. In order to pay the NO multiple hash tokens, the MN does not have to attach multiple hash values. Instead it can just attach the correct hash value further up the chain. By sending $P1_5$ after $P1_1$, it is the equivalent of sending four payment hashes $P1_2$ to $P1_5$, since they can be obtained from $P1_5$ by repeatedly applying the correct hash function. Also, since the datagrams are handled in the normal manner by intermediate routers in the core network as well as the destination node, existing applications can be used without modification.

The charge details that the MN negotiated with the NO for accessing data services in the core network allow for a negotiated amount of data to be delivered to the MN in response to a request. However the gateway node can at any stage demand that the MN release further payment tokens, if the traffic levels exceed the agreed contract details. Datagrams on the reverse path are delivered without modification to the MN. After a call finishes, the MN can use the unspent hashes in the remaining chains on other calls. During a call the routers in the access network ensure that the hashes sent with a datagram have not already been used.

4.4.6 Handover and Authentication

During handover a MN changes its point of attachment from one base station to another in the access network. New nodes in the path to the GW must hash back to the anchor of the current chain, while existing nodes are only required to hash back to the last value stored in their authentication cache. Figure 4-6 shows the message exchanges when the MN moves between base stations. Prior to moving, the MN was using the first sub-chain ($P1$) and the last hash value it released was the fifty-third hash ($P1_{53}$) in that chain. When the MN changes base stations, it transmits a datagram with the hash value ($P1_{57}$) from the $P1$ chain.

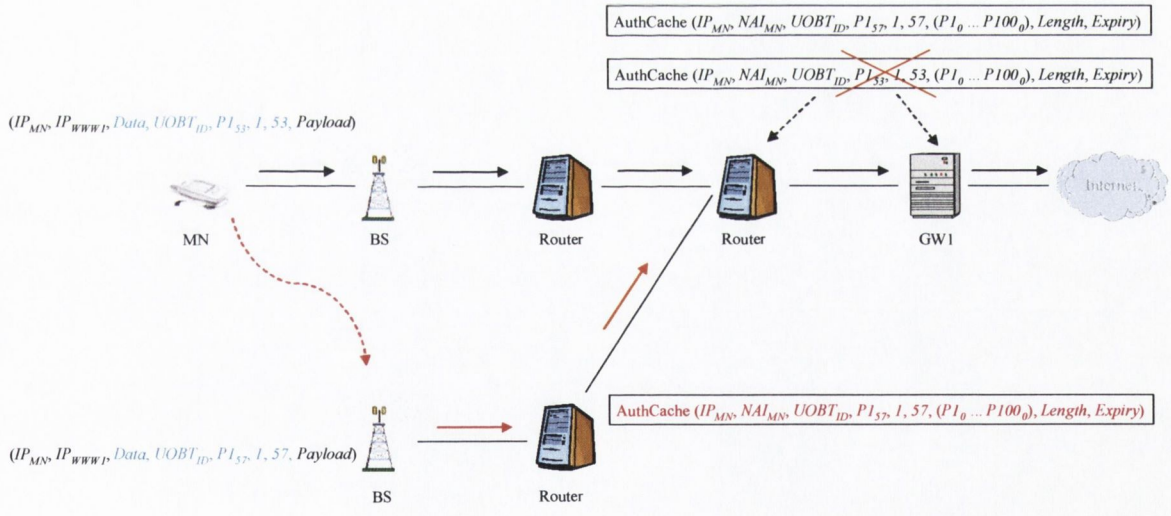


Figure 4-6 Handover in the Access Network

Any new routers in the path towards the gateway node verify the hash value by hashing back to the current sub-chain anchor ($P1_0$) and consult their authentication cache. Existing routers in the new path to the gateway only need to hash back to the last stored hash value ($P1_{53}$) in their AuthCache. It has been shown that an ordinary desktop machine can perform over a hundred thousand hash operations per second [Mep00]. Therefore there should only be a minimum amount of delay in setting up the new path in the access network. In case a MN has no data to send but wishes to keep its route entries alive in the access network, it can forward the required number of hash tokens with a network specific keep alive message.

4.4.7 Call Delivery – Voice

Whenever a user wishes to make a voice call to another user in the network, the MN must first obtain the current care-of address for the destination node. The MN does this by requesting its location management server (LS2) to obtain the care-of address from the correspondent node's location server. The CNs location server (LS1) can be identified from the NAI of the correspondent node (e.g. joe@ls1.net).

The MN transmits a datagram destined for its location server (LS2) via the GW node, the payload of which contains a Location-Request message of the general form $PK_{LS_{ID}}$(*Location-Request*, NAI_{MN}, NAI_{CN}, $UOBT_{ID}$, PX_Y, *X*, *Y*). As far as the GW is concerned the MN has transmitted another data packet and it needs to validate the attached payment tokens. The GW removes the mobility header and forwards the packet towards the destination. The mobile node's location management server decrypts the contents of the payload to reveal a Location-Request message from the MN, along with the payment tokens required by it to complete the operation.

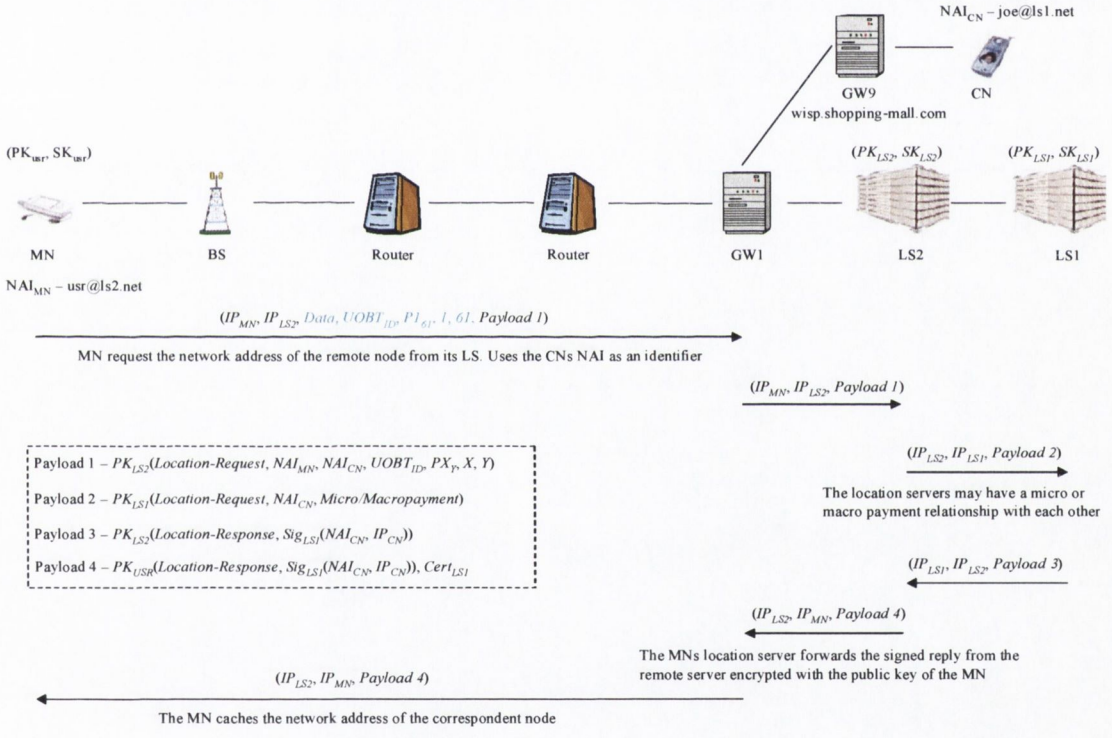


Figure 4-7 Location Query

The location server uses the NAI_{CN} field in the message to obtain the address of the location management server that hosts the CoA information for the CN. It sends its own location request message to LS1 asking it for the current care-of address for the CN along with a micro or macropayment for LS1. This message is of the general form $PK_{LS_{ID}}$(*Location-Request*, NAI_{CN}, *Micro/Macropayment*). LS1 responds with a Location-Response message of the general form $PK_{LS_{ID}}$(*Location-Response*, $Sig_{LS_{ID}}$(NAI_{CN}, IP_{CN})) which is forwarded to the MN. The details are signed with the private key of LS1 and encrypted with the recipient's (LS2) public key, so that intermediate nodes are not be able to obtain any details of the whereabouts of the mobile node (CN). LS2 decrypts the message and then re-encrypts response from LS1 with the public key of the MN, thus

protecting the whereabouts of the CN from nodes in the access network. This last message is of the general form $PK_{USR_{ID}}(Location\text{-}Response, Sig_{LS_{ID}}(NAI_{CN}, IP_{CN}), Cert_{LS_{ID}})$.

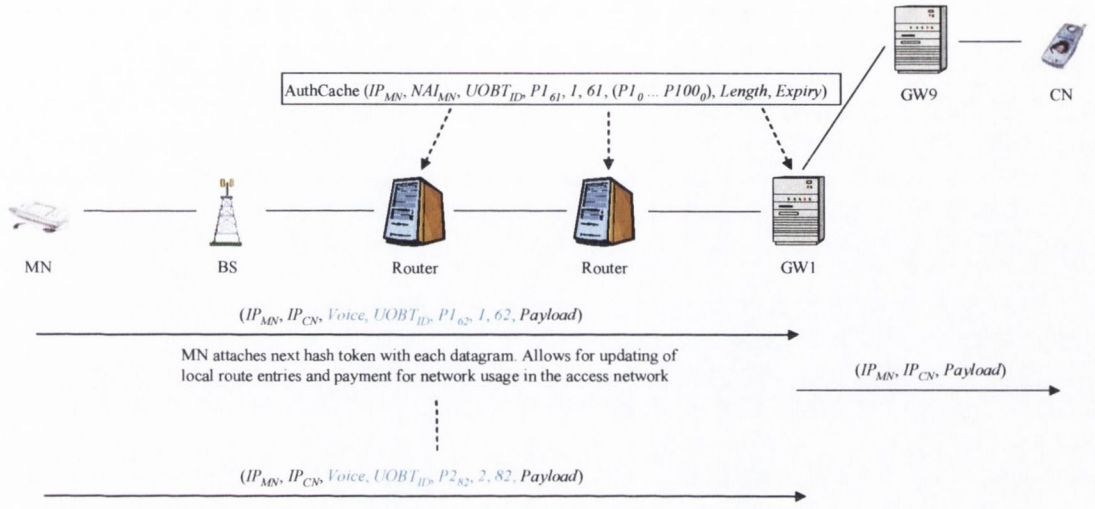


Figure 4-8 Making a Voice Call

Once the MN has the network address of the CN, it can forward datagrams directly to the CN via the gateway node, as long as it pays the NO for network usage in the access network. The MN indicates in the mobility header to the gateway node that it is sending voice data in the payload field to correspondent node, and attaches the required number of payment tokens with each datagram. Intermediate routing nodes are able to verify the authenticity of the datagrams prior to making a routing decision. If the MN exhausts the current chain it can immediately start using the next available sub-chain of the UOBT.

4.4.8 Broker Clearing

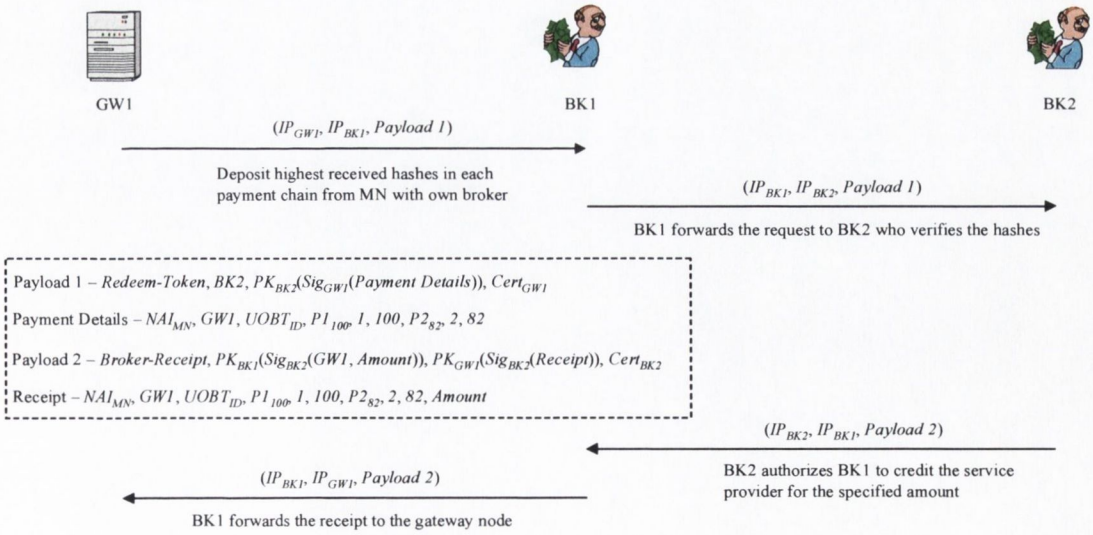


Figure 4-9 Redeeming Payment Hashes

Periodically, a service provider contacts his broker and deposits payment tokens that it has collected for usage of network resources by mobile nodes. He encrypts the payment details with the public key of the broker which issued the payment chains. This message is of the general form (*Redeem-Token*, BK_{ID}, $PK_{BK_{ID}}(Sig_{GW_{ID}}(NAI_{MN}, GW_{ID}, UOBT_{ID}, PX_Y, X, Y))$). The NOs broker (BK1) forwards the message to the issuing broker (BK2) and marks the transaction as pending. The actual payment details are encrypted with the public key of BK2 and not accessible to BK1. BK2 verifies the payment tokens and checks them for double-spending.

A reply is sent back to BK1 which authorizes him to credit the NOs account for a specified amount. The reply from BK2 also contains a payment receipt which is encrypted with the public key of the NO, and is forwarded to the NO by BK1. This message is of the general form (*Broker-Receipt*, $PK_{BK_{ID}}(SIG_{BK_{ID}}(GW_{ID}, Amount))$, $PK_{GW_{ID}}(Sig_{BK_{ID}}(NAI_{MN}, GW_{ID}, UOBT_{ID}, PX_Y, X, Y, Amount))$, $Cert_{BK_{ID}}$). The brokers in the system have accounting relationships and periodically transfer funds between each other to settle user accounts. The value of unspent hashes can be reclaimed by a user once the chains have expired and the spent hashes have been deposited by the service provider.

4.4.9 Discussion

The proposed solution provides a means of allowing real-time payment for the provision of network services and authenticated router updates in the access network. The need for billing procedures from the access network has been eliminated. The notion of a home network has also been made redundant, which is in contrast to the AAA schemes that are currently being used or are being proposed for next-generation networks as highlighted in Chapter 2. The resulting solution also differs from the multi-party payment protocol proposed in [Mep00] as it is assumed that the NO has a high-speed connection into the core Internet and has a traffic contract negotiated with its upstream provider. This eliminates the need for the MN to pay multiple operators in the path to the destination node and further simplifies the accounting procedures.

The proposed solution also radically alters the current business model and paves the way for a new generation of independent small and medium sized NOs and VASPs. Large numbers of new entrants into the market will lead to more choice for end users, new innovative services, and more competitive tariffs. The proposed solution however does introduce a new entity into the system, namely the broker. A broker infrastructure will have to be created to allow users to purchase tokens at their preferred broker with whom they may have an accounting relationship, and to be spent at any NO from where they wish to access network services. The system will also require inter-broker settlement procedures. A good candidate for this role could be the banks which already have a large user base.

A micropayment scheme using hash functions and offline broker contact allows the solution to be efficient and scalable. To aid performance, digital signatures are only used at call setup time and to generate the broker commitment. Subsequent authentication and accounting procedures are achieved using only hash tokens. The protocol is generic enough to be easily integrated to any of the micromobility schemes that have been studied in Chapter 2, and makes use of an additional mobility header that is access network specific. The GW node in the access network removes this header prior to any datagrams being forwarded onto the core network. On the reverse path the datagrams are delivered without modification to the MN. This ensures that existing Internet applications can work without any modifications. It is recommended that each router in the access network maintain an authentication cache which it must consult prior to making any modifications to its routing table. However this is at the discretion of the NO. Registering the care-of address at a location management server is also an optional feature, which may or may not be used by all users in the system. A user may only wish to access data services or make outgoing calls from his MN, in which case he is not

obliged to register his current CoA with a location management server. Only if a user wishes to be reachable for incoming calls, does he needs to update his care-of address periodically at a LS and pay for the privilege.

Hash chains are of a finite length and there is a possibility that a node may run out of hash values during a session. In the case of a UOBT if there are unused sub-chains, then the user can switch to the next sub-chain immediately. If there are no further sub-chains available then this can result in the dropping of a connection. This is particularly true for real-time communications such as voice telephony or video conferencing, where the source may transmit a large number of datagrams during a session. However, this applies to all payment protocols that have a fixed amount in the user's purse.

When a mobile node moves between access networks any ongoing calls may be dropped. This is due to the fact that the MN will need to register in the new network and obtain another IP address. In addition, it will require new provider specific payment tokens to pay for network usage. If on the other hand a MN already possesses valid payment tokens from a prior visit, it may use them instead. If the user was involved in a non real-time session such as accessing a Web server, then there may only be a small delay before the connection can be resumed. In some cases the delay may be so small that the user may not notice that a new connection has been established in the background. For voice calls however, a new connection will have to be setup with the destination node.

With the exception of purchasing payment chains and location management procedures, the overhead incurred by including the mobility header in each datagram is 16 bytes for the hash value (assuming the use of MD5 to generate the sub-chains) and another 8 bytes to represent the payload type, UOBT identifier, sub-chain number and hash number. Thus a total of 24 bytes is added as part of the mobility header within the access network, though these figures are eventually implementation dependent.

4.5 Implementation Details

In order to validate the proposed authentication and accounting protocol for next-generation networks a number of options were considered. These ranged from implementing a real testbed, to developing a simulated network. The first option required configuring wireless hardware and software modules to achieve the required network configuration. Since the protocol was designed to provide authentication and accounting services in a micromobility environment, this would further require the need to implement a basic micromobility protocol and extensions to the IP protocol stack. This option was not pursued further as it demanded a considerable amount of development time and subsequent deployment effort.

Instead a decision was made to make use of network simulator *ns-2* [NS] to test the protocol theories. The terms *ns-2* and NS will be used interchangeably to refer to the network simulation software in the rest of this thesis. The Columbia IP Micromobility Software suite [CMIS] was used as the micromobility protocol. In particular, the Cellular IP (CIP) component of the CMIS software suite was extended to implement a micropayment solution [CIP]. The OpenSSL libraries were used to implement the cryptographic functionality [OSSL]. The resulting modified protocol is referred to as *MobPay*. The features of each of these packages are briefly described in the following sections, prior to describing the implementation details. It should be noted that the MobPay prototype developed in this chapter is a purely proof of concept implementation. A detailed security evaluation will be required prior to deploying the protocol in a live network scenario.

4.5.1 *ns-2* – The Network Simulator

NS is a discrete event packet-level simulator designed specifically for networking research and is considered the simulator of choice in the network research community. NS provides substantial support for simulating routing, multicast and IP protocols such as UDP, TCP, and RTP over wired and wireless links. NS began as a

variant of the REAL network simulator, which was intended for studying the dynamic behavior of flow and congestion control schemes in packet-switched networks. *ns-2* is currently being developed by the Virtual InterNetwork Testbed (VINT) group in the University of Southern California [VINT]. The work is supported by a number of groups including the Defense Advanced Research Projects Agency (DARPA) [DARPA]. While considerable effort has gone into developing it, *ns-2* is still a work in progress. *ns-2* is open source and freely available to download form the ISI website [NS].

The simulator framework uses a split-language programming approach. The core of the simulator (i.e. the low-level event processing, per-packet actions such as forwarding etc.) is written in C++ to allow for fast simulation of large scenarios. OTcl, an object oriented version of Tcl [TCL], is used for the control structure and the description of the simulation scenarios. Also the scheduling of events and the dynamic configuration of network components during simulation is usually done in OTcl. Users can setup an actual scenario by writing a simulation script in OTcl. In this script the user defines the network topology and exactly what events should occur during the simulation. The script is then fed into the OTcl interpreter which builds a corresponding simulation for the C++ objects in the NS simulator library. Users can also specify in the OTcl script if any events are to be traced. An entry will then be made to a trace file anytime that event occurs. When the simulation has finished executing, the Network Animator (Nam) tool can be used to visualize the simulation [Nam].

Although NS has built-in support for the Mobile IP protocol, it does not have any micromobility protocol support. NS can however be extended by patching in the CMIS micromobility suite. The CMIS implementation is an extension for the *ns-2* network simulator based on version 2.1b6. The CMIS v.10 release includes *ns-2* implementations of the Hierarchical Mobile IP, Cellular IP and HAWAII micromobility protocols. The CMIS libraries are also open source and available to download from the Columbia University website [CMIS].

4.5.2 OpenSSL Cryptographic Toolkit

OpenSSL is an open source cryptographic toolkit written in the C programming language. OpenSSL provides routines for cryptographic primitives utilized in implementing the Secure Sockets Layer (SSL) protocol, as well as a full-strength general purpose cryptographic library. It has support for public-key ciphers such as RSA, and message digest functions such as SHA-1 and MD5. The OpenSSL libraries run on a variety of platforms and are downloadable from the OpenSSL website [OSSL].

The OpenSSL library (version 0.9.7a) cryptographic primitives were used for generating the UOBT hash chains and implementing the micropayment protocol. The library was also used for generating public-key pairs for the various nodes in the system. A certification authority was created specifically for the simulation and was assigned a self-signed 2048-bit certificate. The CA in turn certified the public keys of the BK, MN and GW nodes to produce X.509 compliant digital certificates. Appendix F provides more details on how to use the OpenSSL API for generating digital certificates.

4.5.3 CMIS – Cellular IP Implementation

A typical Cellular IP network consists of a hierarchy of CIP nodes, with one or more Base Stations (BSs) as the leaf nodes [CGKW+02]. CIP nodes maintain a route and page caches. The GW which is also a CIP node acts as the interface to the Internet. As the MN roams in the network it receives datagrams from a CN which resides on the Internet.

A brief walk-through the original Cellular IP simulation is as follows:

- A MN enters the access network and begins communicating via BS1.
- Prior to the MN receiving any data packets from the CN, it must obtain a care-of address in the access network. To do this it sends a location update packet.
- The location update packet, updates the routing caches in all the Cellular IP nodes on the path to the GW (CIP Node 3 and CIP Node 1).
- One second into the simulation the CN is instructed to start sending datagrams to the MN. These travel through the Internet via the GW which then delivers them to the MN via CIP nodes 1 and 3.
- Depending on transmission characteristics (UDP or TCP), the MN sends periodic updates towards the GW to keep the soft-state cache entries alive.
- As the simulation progresses, the MN moves at a constant speed of 20 m/s in the direction of BS4. Three handovers occur from BS1→ BS2, BS2→ BS3 and BS3→ BS4. At time 30.0s the MN reverses its direction and another handover occurs from BS4 → BS3.

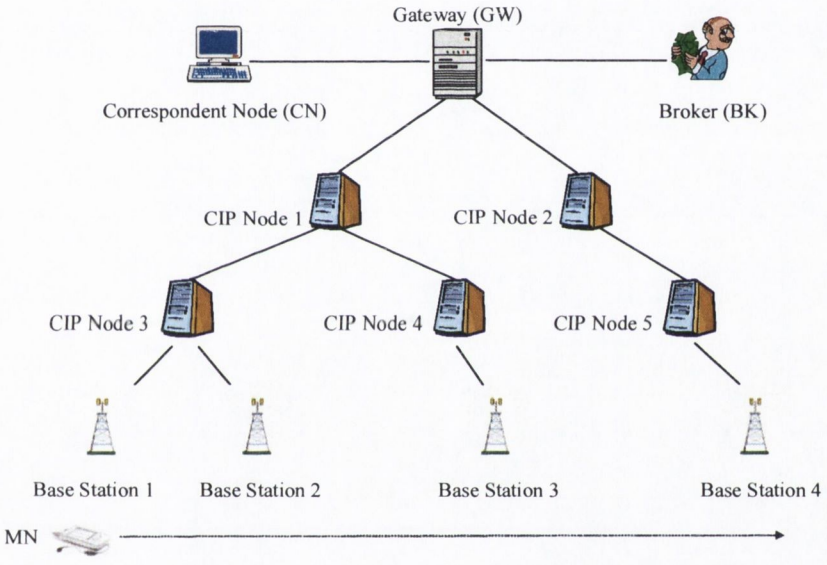


Figure 4-10 MobPay Network

4.5.4 MobPay – Cellular IP Extensions

In addition to the basic CIP entities defined in the CMIS implementation, a Broker (BK) object has been implemented in the MobPay version as shown in Figure 4-10. The broker allows for the purchase of payment tokens by a MN for use in a particular access network. As a result of this modification, a number of new steps were added to the simulation prior to the exchange of datagrams between the CN and MN.

- Prior to the exchange of any data packets in the access network, the MN generates a 100x100 UOBT and sends a purchase request (purchaseReq) to the broker. The MN signs the request with its private key, and encrypts the contents using the public key of the broker which it obtains from the CA signed X.509 certificate.
- The broker creates a commitment by signing the anchors of the UOBT sent as part of the purchase request with its private key and returns them to the MN via the GW. The GW makes the anchors available globally to the CIP nodes within the access network.

- Location update packets from the MN now contain a hash value. The CIP nodes along the path attempt to authenticate these packets by hashing back to the last recorded hash value in their authentication cache prior to forwarding the packet and updating their route cache.

More details of the various code additions and enhancements can be found in Appendix B of this thesis.

4.6 Experiments and Measurements

In order to investigate the performance of the MobPay protocol a number of measurements at different nodes in the network were taken. The aim was to determine the overhead at the CIP nodes in the core network when authenticating the hash value(s) sent along with a route update packet. Comparisons could then be made with the timings obtained from the original CMIS simulation and graphed.

However *ns-2* is a single threaded, discrete event simulator that employs its own virtual timing systems instead of using real time. This means that when an event occurs a counter is incremented by the time scheduler. The amount this counter is incremented by is determined by an *estimation* of the real time duration of the event. To get the time an event occurred in the simulation one cannot just print out the system time. Instead the NS time scheduler must be called during the simulation. This can be done from the OTcl scripts. However a single line in the OTcl script can trigger multiple events. So for some cases it is not possible to know the exact time a particular event occurred. It is also important to note that *ns-2* is not a system simulator and thus does not take into account a node's processing time. It was primarily designed to simulate network link delays, transmission times etc. In order to take into account the cryptographic processing overhead associated with the MobPay protocol, the processing overhead at each node had to be calculated. This time was subsequently added as a delay to the NS scheduler. This had the net effect of propagating the processing delay through the nodes in the network and thus providing a better measure of the overall network latency.

To obtain the cryptographic processing overhead the UNIX *gettimeofday()* library function call was used, which allows timing measurements to be obtained at the sub-millisecond level [Bra01]. In particular the processing overhead at the CIP nodes in the core network was measured. Appendix B.3 provides the code segment which shows how this was achieved. Finally, it is assumed that the generation of the UOBT on the MN and the purchase of micropayment tokens by obtaining a commitment from the BK can be performed offline, and thus do not add any processing overhead during this data transfer phase in the access network.

4.6.1 Experimental Setup

A 400MHz Pentium II with 256MB of memory running the Linux 2.2.20 kernel was used as the testbed machine. The network simulation environment ns2.1b6 was first installed. An older release of the software was used as this is the specified version to which the CMIS micromobility extensions need to be applied. This version served as the base installation for obtaining measurements for the Cellular IP protocol. The whole process was repeated and a parallel installation to which the CMIS patches were also applied was created. The protocol specific code changes were then made to this installation to enable authentication and payment in a micromobility environment. This modified distribution is referred to as MobPay in this thesis.

Having made the changes and verified that all aspects of the protocol had been properly tested, it was time to obtain timing measurements. Figure 4-11 shows a screenshot of a MobPay simulation running in a Nam console. Each run of the simulation produces a large output file to which are logged various important aspects of the simulation, such as when particular packets were sent or received by nodes in the network, the packet type, packet size etc. More details about the NS trace file formats can be found in [FV03, Gre02].

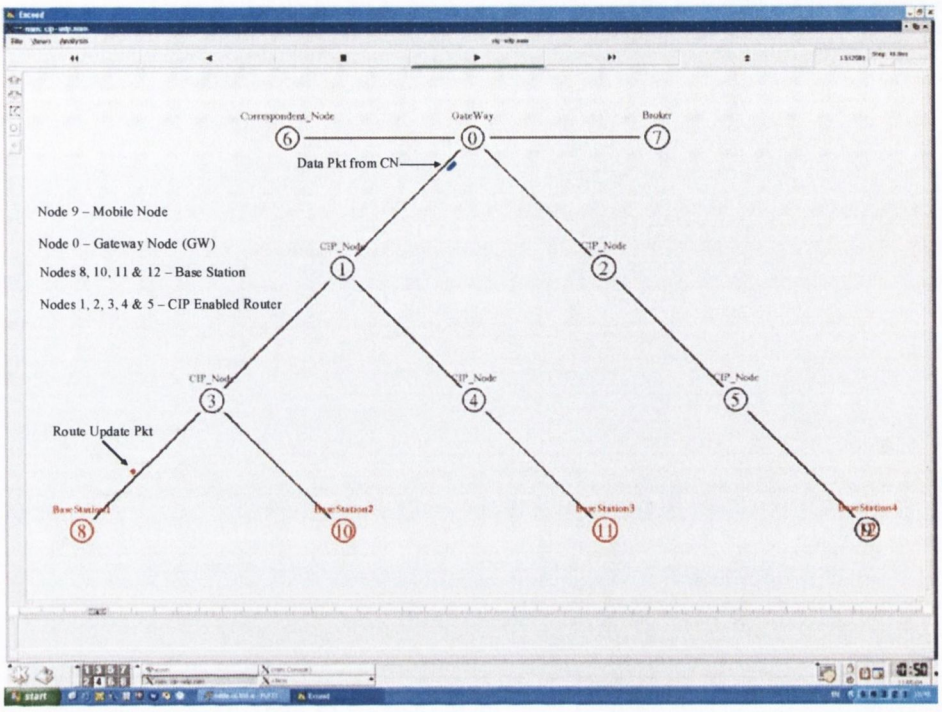


Figure 4-11 Nam Screenshot of an *ns-2* Simulated MobPay Network

Two sample entries from a trace file using the new trace format are presented below:

BS1 (node 8) receives a data packet (CBR)

r -t 2.011831259 -Hs 8 -Hd 4194304 -Ni 8 -Nx 1.00 -Ny 1.00 -Nz 0.00 -Ne -1.000000 -Nl MAC -Nw --- -Ma a3 -Md 0 -Ms 1 -Mt 800 -Is 4194306.2 -Id 0.0 -It cbr -Il 70 -If 3 -Ii 117 -Iv 32 -Pn cbr -Pi 5 -Pf 1 -Po 0

Data packet enqueued at the GW (node 0) for transmission to the CN (node 6)

+ 2.018125 0 6 cbr 70 ------- 3 1.0.2.2 0.0.0.0 5 117

The first entry shows a CBR packet sent by the MN towards the GW node. The scheduler time when the packet reaches one of the base stations (CIP nodes 8, 10, 11 or 12) from the mobile node is recorded and used as the *start time*. The scheduler time is again recorded when the update packet is queued at the GW for transmission to the CN and this value is used as the *stop time*. The difference between the two values is the network delay, plus the processing overhead at each CIP node in the path. So for example, if the MN is in the radio coverage of BS3, the network delay and the cryptographic processing overhead at the nodes in the path from BS3 to the GW (i.e. nodes 11, 4, 1 and 0) is included.

4.6.2 Evaluation of Results

Two traffic types are defined in the CMIS Cellular IP implementation, namely UDP and TCP. A UDP connection would most likely be used in a time sensitive application such as VoIP, while a TCP connection would be employed for a reliable connection transfer such as a HTTP exchange between a Web client and server. In each case the NS simulation was run only once for the CMIS CIP distribution, as the timing values remained constant for a given set of parameters. However the NS simulation was run 10 times for both the

UDP and TCP connections for the MobPay distribution and an average of the measurements was computed. The results obtained are now described in detail.

Figure 4-12 graphs the time it takes for a route update packet to reach the GW node from the first CIP node it encounters in the core network, as the MN moves from BS1 to BS4 and back during a UDP session. The time taken by the unmodified CIP protocol versus the MobPay version was plotted. It can be observed that the CIP values exhibit a flat line with a time of 6.2ms for each route update. The MobPay simulation on the other hand requires on average a time of 6.3ms for an update packet to reach the GW node, and an average protocol overhead of 121μs. Appendix B.4 provides the Perl script which was used to obtain the timing measurements from the output of a UDP simulation.

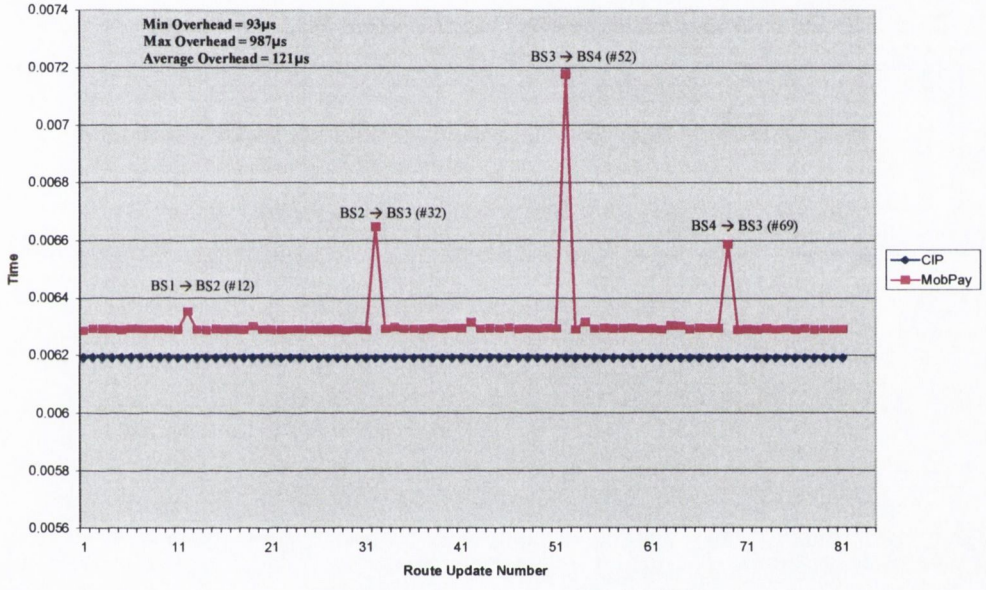


Figure 4-12 Protocol Overhead UDP Traffic

In addition, it can be seen that when a handover takes place, there is usually a larger overhead at points 32, 52 and 69, with the exception of the first handover at point 12. The reason for this is that during the handover from BS1 to BS2, there is only a single node change in the path to the gateway node (BS2), and all the nodes in the path have the last used hash value. The overhead is greatest at point 52 during the handover from BS3 to BS4 as there are two new CIP nodes (5 and 2) in the path to the GW. Each node needs to hash back to the anchor of the UOBT sub-chain in order to verify the hash token.

During a TCP session the MN responds to each packet that is sent from the CN with an acknowledgement (ACK). Each ACK message updates the route caches along the path to the GW and thus must be accompanied by a hash token. In addition, during handover between base stations, the MN also sends a route update message as before. This results in a total of 3960 packets being sent by the MN. Figure 4-13 graphs the time taken by the CMIS CIP version versus the time taken by the MobPay version during a TCP session. It can be seen that the average protocol overhead in the MobPay simulation is 87μs, with larger overheads observed at the handover points in the simulation. These values are broadly in line with what was obtained

for the UDP simulation. It should be noted that for UDP transmission, the MN only sends route update packets prior to the cache entries expiring in the CIP nodes and that in the experiments that were carried out only a single hash token is sent along with each update. This may not be sufficient in a real network scenario where the MN is required to pay for network access and the delivery of datagrams. For such UDP connections, the MN may need to send hash values higher up the UOBT sub-chain to satisfy the payment requirement. However this should not impact greatly the overall timing measurements, as the time required to verify a single hash value is in the order of microseconds.

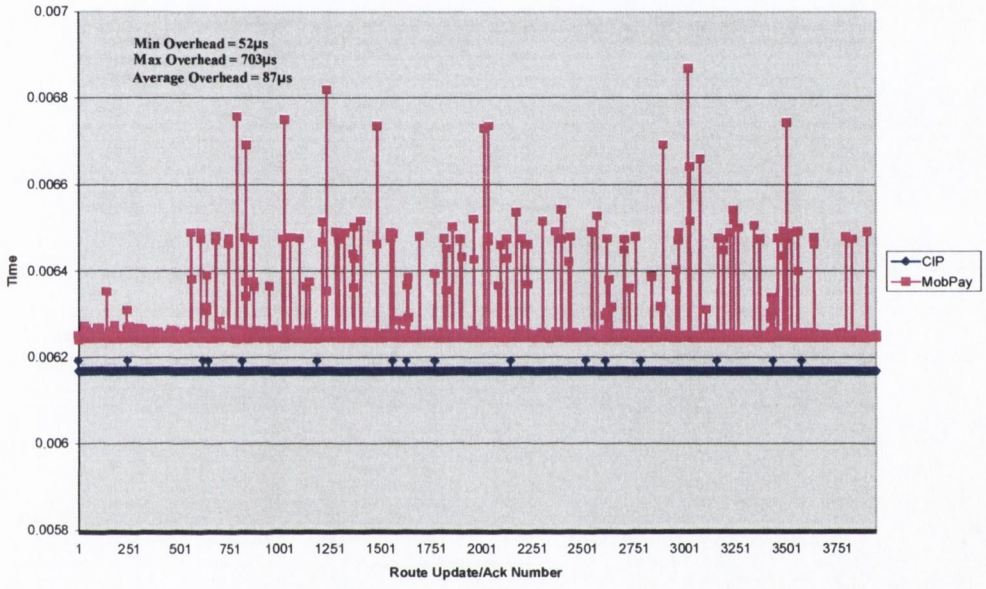


Figure 4-13 Protocol Overhead TCP Traffic

Another solution of this general type has been described by Knospe and Schwiderski-Grosche [KS02a, KS02b, SK02, KS02] as part of the SHAMAN (Security for Heterogeneous Access in Mobile Applications and Networks) project [IST00], in which the authors propose a smart card based payment system for heterogeneous network access in next-generation mobile networks, without the need for a preestablished business relationship with the NO. This solution differs from the payment scenario outlined as part of the MobPay protocol in that the transaction amount can only be established at the end on the communications session. Also in the SHAMAN protocols there is need for each mobile node to be equipped with a smart card module to store and encrypt the payment tokens, and make use of the IETF AAA procedures for authentication and accounting. This again is in contrast to the MobPay approach which eliminates the need encrypting the payment tokens across the network interface and for employing the heavyweight AAA protocols for the authentication of the entities in the system.

4.7 Summary

With the widespread use of PDAs and the availability of low-cost wireless networking hardware, there has been considerable growth in the number of wireless access networks. Currently such networks are operated by individual organizations and are usually closed to users who belong to other network operators or organizations. One of the reasons for this is that such closed networks do not have any AAA provisioning

policies in place, and thus cannot deal with nodes with which they do not have a pre-established security relationship. With a large number of network operators and service providers, it is necessary to guarantee payment and remove any complex trust relationships involved in billing. A number of problems with the current AAA mechanisms for Mobile IP based access networks were identified. The desirable properties of an authentication and accounting system were then drawn up, and a solution which securely achieved those goals was proposed.

To allow for an efficient solution with minimal computational costs during packet authentication, a scheme based on hash chain trees was used. The use of hash chain trees is an advantage over ordinary hash chains, as it allows for efficient generation and storage of hash values within a mobile device which may have limited storage capacity. The number of cryptographic keys required has been kept to a minimum while allowing for fast verification of the authentication data carried within the datagrams. The proposed micropayment solution eliminates the need for complex accounting and billing strategies to be employed in the network to remunerate NOs and VSAPs for service provision.

Unlike traditional billing, the proposed scheme eliminates the fraud associated with postfact billing as it allows the payment tokens to be only spent at the specified NO. The service is provided by the operator as long as the MN attaches the correct payment tokens along with data packets that it transmits though the access network. The requirement to contact a distant home network for user location and authentication information has also been eliminated. A preferred VASP is used instead to store the current location details, which is paid in real time for the privilege of maintaining the same. Other users who wish to communicate with a roaming user must use the NAI of the MN to contact his location management server, and pay to obtain the location details. This feature will help in reducing the number of unsolicited or nuisance calls.

As mobile communications becomes increasingly sophisticated and ubiquitous, it is critical the problems of CDR billing be addressed. With this in mind the proposed protocol seems to be a flexible, secure, and practical answer. In Chapter 5, the basic MobPay protocol has been adapted to an ad hoc network scenario. In the case where the destination is more than one hop away from the source, multiple nodes in the relay path will have to be compensated. Hence a multi-party real-time micropayment protocol will be required.

5 Multi-Party Micropayments for Mobile Ad Hoc Networks

"Take care of the pennies and the pounds will take care of themselves."
Scottish Proverb

5.1 Introduction

Towards the end of Chapter 2, an overview of an emerging networking paradigm which will have a profound impact on the make up of next-generation mobile networks was presented. Self-organizing or ad hoc networks as they are more popularly known have been a topic of research for the military for many years. It is only in recent times that research has focused on developing ad hoc network routing protocols to work in general purpose networking environments. An ad hoc network is an infrastructureless mobile network which consists of a group of nodes that communicate with each other using multi-hop wireless links. Each node acts as a router to forward packets for other nodes within the network.

The majority of ad hoc networks routing protocols have been designed with closed user groups in mind. In closed environments such as military or emergency services networks, all the nodes in the network belong to a single authority, and are motivated to work with each other to cooperate in the relaying of packets between the source and destination. Recently ad hoc networks have found their way into everyday networking environments, where mobile devices may be under the administrative control of individual users. These users may not necessarily be motivated to provide services for free to others in the network. A typical situation could arise whereby a node which is outside the communications range of the nearest base station, may wish to relay packets through a number of intermediate nodes in the ad hoc network, to access services in the fixed network such as the PSTN or Internet. One of the major concerns of such users will be to preserve the limited battery life of their mobile devices and the relaying of packets directly impacts on this.

It is envisaged that ad hoc networks will complement cellular networks and WLANs in the future for providing *last-mile* access to the wired infrastructure. Hence mechanisms must be in place to compensate the nodes involved in the relaying process and for any value-added services that they provide to other nodes. However in ad hoc networks, the luxury of long-lived trust relationships between nodes in the network does not exist, as is the case in the wired Internet. Also, it is not always possible to contact a TTP to verify an identity or payment instrument that may be presented to a node. In the latter case, the cost of contacting a TTP to verify payment tokens may outweigh the actual benefits gained as highlighted in Chapter 3.

In this chapter, a lightweight multi-party micropayment scheme based on UOBT hash chains is presented, which allows a node to pay others who relay packets on its behalf in real time. Due to the dynamic nature of an ad hoc network, the topology of the network can change unpredictably. The design of the payment scheme is flexible enough to be able to cope with such route changes, without the need to contact a trusted third party

such as a bank or broker to pay the nodes in the new path. The related work in the field, followed by the system model is presented in Sections 5.2 and 5.3. In Section 5.4, a number of protocol goals for an ideal payment system for ad hoc networks are outlined. In Section 5.5, a multi-party micropayment scheme is proposed and its various merits and drawbacks are discussed. Sections 5.6 and 5.7 provide the implementation details and experimental results. Finally in Section 5.8, a number of concluding remarks are made.

5.2 Related Work

Some previous work in this area has been done within the framework of the Terminodes project [BBCG+01]. The authors indicate that there could be *selfish* nodes in the network that will use services provided by other nodes, but modify their own behavior such that they do not cooperate in providing free services to others in the network. They have proposed a virtual currency called *nuglets* that attempts to stimulate cooperation within the network by rewarding nodes for service provision [BH01]. Nodes in a Terminodes network are required to pay for services from other nodes using nuglets. In addition, the authors feel that payment using nuglets on a per packet basis will deter users from overloading the network.

Since the only way to transmit data on the network is by paying other nodes using nuglets, it is in the interest of users to keep their nodes in an active state and accumulate nuglets by relaying packets for others. When transmitting a packet using the Packet Purse Model, a node attaches sufficient nuglets such that each intermediate relay node will acquire some of the nuglets to cover its forwarding costs. To prevent nodes from acquiring more than their fair share, the designers make use of tamper-resistant modules in each node. The secure hardware module maintains a nuglet counter and modifies its contents according to the number of nuglets received or spent by a node. To prevent forgery of nuglets contained within the packet purse during transmission, neighboring nodes use public-key cryptography to negotiate symmetric keys in order to encrypt the packet contents. The authors also present the Packet Trade Model, where each intermediate node buys a packet from the previous node for some nuglets and sells it to the next node for more nuglets.

There are a number of drawbacks to the Terminodes approach. There is total dependence on the secure hardware to protect nuglets in the system. If the module's security is compromised, users can mint nuglets at their discretion and disrupt the cooperative nature of the network. In addition, there are no methods defined to determine the total cost of transporting a packet from a source to destination. If the source fails to attach sufficient nuglets with a packet, then an intermediate node may drop the packet prior to it reaching its destination. There are no mechanisms in place to notify the sender of the same.

The Secure Charging Protocol (SCP) proposed by Lamparter et al. [LPW03] is similar to the Terminodes approach in that it provides incentives for forwarding packets in the network. Each node in the SCP protocol is allowed to send its own packets if in the past it has forwarded sufficient foreign packets. The authors depart from the idea of purely self-configuring ad hoc networks in the total absence of any endorsing agency. Instead they believe that some governing agency such as an Internet Service Provider (ISP) will play an administrative role in the network, such as authenticating the nodes in a given communication, and ensuring that nodes are charged or rewarded for packet transmission and forwarding. The protocol requires the use of the heavyweight AAA protocols, public-key certificates and digital signatures to verify identities at each intermediate node. It also makes use of a keyed hash chain, which allows the serving Access Point (AP) of the ISP to verify each intermediate node's participation in the packet forwarding process. SCP requires complex interactions between the MN, AP, CN and other intermediate nodes in the network.

Zhong et al. [ZCY03] propose Sprite, a credit-based incentive scheme for ad hoc networks which does not need any tramper-resistant hardware to ensure the security of the system. When a node receives a message it

keeps a *receipt* of the same. At a later date when the node has a fast connection to the Credit Clearance Service (CCS), it reports to the CCS the messages that it has received or forwarded by uploading the receipts. The CCS uses a complex mechanism to compute the charge and credit to each node involved in the transmission of a message, which is dependent upon the total reported receipts of a message. The amount of payment is also dependent on the particular nodes that submitted the receipts. There are a number of drawbacks to the scheme. The first is to do with the storage of receipt material at each node for all the packets that it has relayed since the last time it was in contact with the CCS. A related issue is the requirement to contact the CCS at periodic intervals to be able to obtain credit for forwarding packets, without which the node will not be able to transmit any of its own packets in the network. The scheme also requires that each node be issued with an identity certificate and makes use of digital signatures to secure the message contents.

In [SBHJ], Salem et al. propose a charging and rewarding scheme for packet forwarding. The proposed system differs from the pure ad hoc network approach, in the sense that it mandates that all communications must pass through fixed base stations in the path between the source and destination nodes. The base stations are connected to a high speed backbone network. Nodes that cannot reach a base station directly can relay their packets via a number of intermediate nodes in the ad hoc network. The network operator charges the initiator prior to delivery of the datagram to its intended destination. Intermediate nodes in the path between the source (node A) and source base station (BS_A) are remunerated when the packet is received by the base station, whereas the downstream forwarding nodes between the destination (node D) and the destination base station (BS_D) are remunerated only if the packet is delivered to the destination node.

The protocol makes use of Message Authentication Codes (MACs). The initiator of session creates a MAC on a number of parameters using its key K_A. Each intermediate node i that decides to participate in the forwarding of the specified traffic then computes a MAC on the whole request using its key K_i, replaces the MAC in the request with the newly computed MAC and forwards the request. Thus when the packet arrives at the source base station BS_A, it contains a single MAC that was computed by node A and all the nodes on the route in an iterative manner. This is more efficient than attaching a new MAC at each hop in the network. Once the session has been setup, the source and destination can start exchanging datagrams. The main drawback of this scheme is the imposition of fixed network elements in the path between the source and destination nodes. Also each node must register with the NO to obtain a symmetric key which is shared between all the base stations in the system. This key needs to be linked to the node's identity in some way.

5.3 System Model

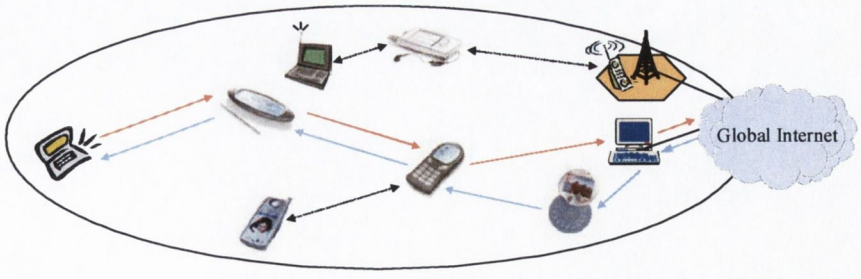


Figure 5-1 Multi-Party Micropayments for Ad Hoc Networks

It is envisaged that ad hoc networks will complement mobile and fixed communications networks in next-generation networking scenarios. Most of the time nodes within an ad hoc network may communicate with each other. However as in current mobile networks, users in an ad hoc network may also wish to access

services in the fixed network. For example, a user might want to make voice calls to other mobile and PSTN users, or access data services on the Internet. Nodes that are in close proximity to a fixed network can extend the reach of the fixed network to other nodes within a mobile ad hoc network. Consider the scenario illustrated in Figure 5-1. On arrival in a new city, a user finds a route through an ad hoc network to a node that has the capability to relay packets into the fixed network. Each of the nodes in the path from the user to the fixed network is actively involved in relaying the packets towards the destination, and need to be compensated for the same. A protocol employing micropayment technology which allows each of the nodes involved in the relaying of the packets to be paid in real time as they provide the service is now proposed.

In mobile and fixed networks the entities within the core network remain static. This allows a user to negotiate a payment contract to pay nodes along the route to the destination [PO99, Mep00]. The payment contract can be bound to the identities of each of the nodes in the path and can be certified by a trusted third party. However such long-lived contracts are not feasible in ad hoc networks. Due to the dynamic nature of an ad hoc network, the route from the source to the destination may change over a period of time, or even during a call. A more lightweight and flexible scheme that allows all the nodes in the path to a given destination to be paid without the requirement to contact a TTP or a bank, to issue a new payment contract or tokens is required [TO03b]. The proposed protocol differs from the approaches outlined in Section 5.2 in that it does not impose any requirement for a central authority such as an ISP to administrate the network, nor does it impose the restriction of one or more fixed nodes in the path between the source and target nodes. As opposed to providing incentives for packet forwarding, the scheme remunerates the nodes in the relay path with real money.

5.4 Protocol Goals

The protocol proposed in this chapter has been designed to address the issue of payment for packet relaying by nodes in an ad hoc network. From an analysis of existing work in this area it was felt that any payment scheme employed in an ad hoc network must have at least the following goals:

Offline Verification of Payment Tokens – Intermediate relay nodes should not be required to maintain an online connection to a bank or a trusted third party to verify payment.

Flexibility in Choosing Routes – A node should be able to independently choose the optimal route to its destination and pay all nodes along the path to forward packets on its behalf. If there is a change in the route towards the destination, then the source should not have to contact a third party to construct a new payment contract to pay the nodes in the new path.

Lightweight Cryptographic Procedures – Intermediate nodes must be able to quickly verify the payment information carried within a packet. Use of heavyweight cryptographic algorithms may result in delays in packet forwarding by intermediate nodes.

Minimize Fraud in the System – The effort required to steal value from the system should be far greater than the rewards gained. Postfact detection should identify the culprits who can subsequently be disqualified from the system.

5.5 Ad Hoc Payment Protocol Design

This section begins by detailing the roles of the various entities in the system, the requirements for each entity in order to engage in the protocol, and any assumptions that have been made in order for the correct

operation of the protocol. The details of the payment protocol are then outlined, from purchasing tokens from a broker, to distributing signed endorsements, and finally redeeming payment hashes for service provision.

5.5.1 Roles Requirements and Assumptions

The main entities involved in the various protocol exchanges are the mobile nodes in the ad hoc network and the financial brokers in the system. The roles and requirements for each of the above entities are outlined below:

Mobile Node – The protocol allows a mobile node in an ad hoc network to pay each node in the path towards the destination in real time for packet forwarding. Each node in the ad hoc network is required to generate or securely obtain a public-key pair, and to subsequently obtain a public-key certificate for the same from a TTP in the network along with the public-key certificate of the TTP. The public-key pair is used for performing encryption, decryption and digital signature operations using the RSA algorithm. The cryptographic keys are stored in a secure tamper resistant device such as a smart card, and all cryptographic functions are executed securely within this device. This ensures that the cryptographic keys and other payment related parameters never leave the secure smart card environment. The smart card must also be able to perform hash operations using the MD5 and SHA-1 algorithms.

Broker – Financial brokers in the system are trusted entities which are required to generate broker signed commitments, which certify mobile node generated hash chains. The signed broker commitment allows each of the recipient nodes in the path between the source and destination in the ad hoc network to be confident that they are receiving valid payment tokens for packet forwarding. A broker must possess a public-key pair and obtain a public-key certificate for the same from a TTP. It must be able to perform public-key encryption, decryption and digital signature operations. It must also be able to perform hash operations. A broker can also take on the role of a TTP to certify the public key of mobile nodes in the network.

The protocol assumes that there is a globally trusted Public Key Infrastructure (PKI) and associated hierarchy of Certification Authorities (CAs) in place. The top level CAs in the system will certify entities such as the brokers in the system. The brokers in turn can certify individual users or nodes. It is assumed that these CAs are highly trusted entities and implement strict security policies in relation to storing root keys, payment information and certifying individuals or entities in the system. It is also assumed that the smart card device is tamper resistant and that the effort required to break the security of the device will far outweigh the subsequent financial gains. The protocol also assumes that a source node can verify that it has obtained a valid route to the destination from the underlying routing protocol, and that there are no loops in the route which may have been formed by nodes colluding to defraud others in the network.

5.5.2 Broker Commitment

A user must first establish an accounting relationship with a broker, whose main purpose is to aggregate micropayments between entities. The broker supplies each account holder with a smart card, which is loaded with a public-key pair that is bound to the user's identity by means of a digital certificate. The card also contains the broker's public-key certificate. The smart card is a secure and trusted tamper-resistant device that acts on behalf of the broker. Prior to transmitting any packets on the network a user must purchase payment chains through his mobile device from the broker. A macropayment scheme such as a credit card or electronic cash can be used to make the purchase.

Figure 5-2 shows the payment chain purchase protocol. The user instructs the smart card module to generate for example a 40x40 UOBT (1600 hash tokens). To obtain a broker commitment, the user forwards a signed message that consists of the set of anchors along with the length of the sub-chains, the value of a single hash

and the macropayment details to the broker, all encrypted with the broker's public key. The tree root of the UOBT and the associated secret root values never leave the secure hardware module on the user's device.

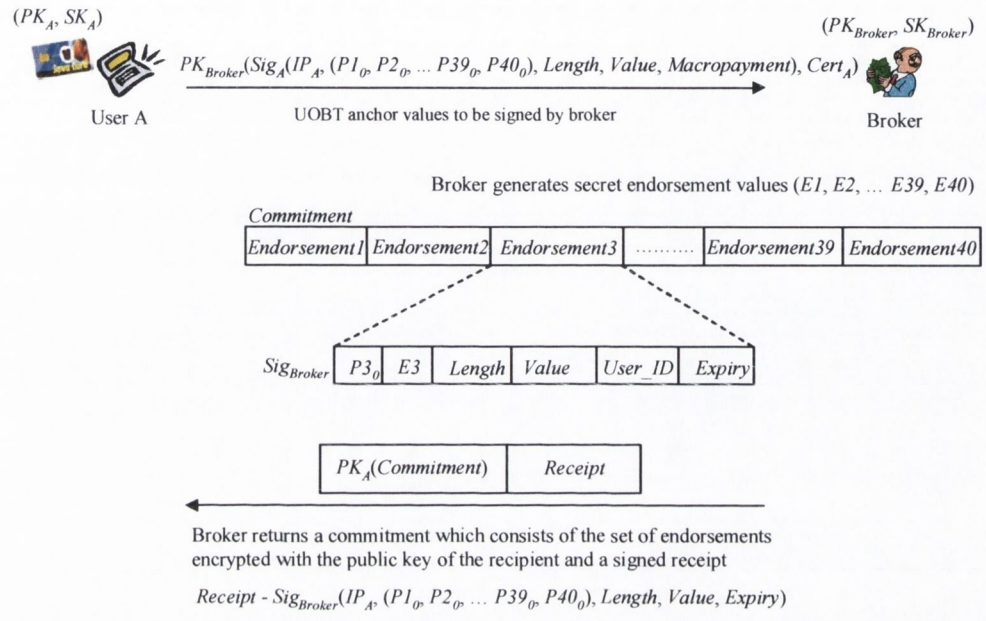


Figure 5-2 Purchase of Payment Chains and Broker Commitment

The broker generates a set of N secret endorsement values, one for each anchor value that was sent by the user. A broker endorsement consists of an anchor value PX_0, the corresponding endorsement value (a random number), the length of the hash chain, the value of a hash in the chain, the identity of the user that purchased the chain and the expiry date of the chain. All of the above fields are signed with the private key of the broker. A broker signed endorsement is of the general form $Sig_{Broker}(PX_Y, E_Y\ Length, Value, User\_ID, Expiry)$. The endorsement allows a recipient to redeem the highest hash value in the corresponding sub-chain of the UOBT at the broker. The *Commitment* (set of broker endorsements) is encrypted with the public key of the user and can only be decrypted within the user's smart card module. The commitment is of the general form $PK_{USR_{ID}}(Endorsement1, Endorsement2 \ldots EndorsementN)$. The broker also forwards a signed receipt to the user, which allows him to verify that the BK has committed to the sub-chains of the UOBT. The broker signed receipt has the general form $Sig_{Broker}(IP_{USR}\ (PX_0, PY_0, \ldots. PZ_0), Length, Value, Expiry)$.

The smart card which acts on behalf of the broker does not allow the user access to the endorsements. This is akin to the *observer* protocol that has been suggested by Chaum in [Cha92]. An observer is a tamper-resistant computer chip issued by some entity that organizations can trust. It acts like a notary and certifies the behavior of a representative in which it is embedded. The endorsements are only released in encrypted form from the user's smart card device to a recipient's smart card device prior to payment for packet forwarding.

Smart card devices are tamper-resistant and *not tamper-proof*, which means that they are physically protected against unauthorized attempts to read or modify their contents. However, no module is designed to withstand adversaries who are prepared to invest millions to break the security of the module [PPSW97]. So even though the scheme relies on these devices to ensure that a user cannot use a payment chain multiple times, it is not able to guarantee this. A determined user could break the tamper resistance of the smart card module

and gain access to the broker-signed endorsements. The protocol limits the exposure of the system from such attacks by associating an expiry date with each chain, after which nodes in the system will not accept the chain as a valid payment instrument.

The size of each broker endorsement is 40 bytes, assuming that the lengths of the anchor and the corresponding endorsement value are 16 bytes each, and that each of the other fields is assigned 2 bytes. The broker's signature with a 1024-bit key amounts to 128 bytes. For a 40x40 UOBT, the total size of the commitment amounts to (40*168) ~7 Kbytes of storage. For a 100x100 UOBT the commitment would be ~17 Kbytes. The above values are however implementation dependant but within the storage capabilities of current high-end and next-generation smart cards.

5.5.3 Charge Assembly and Endorsement Distribution

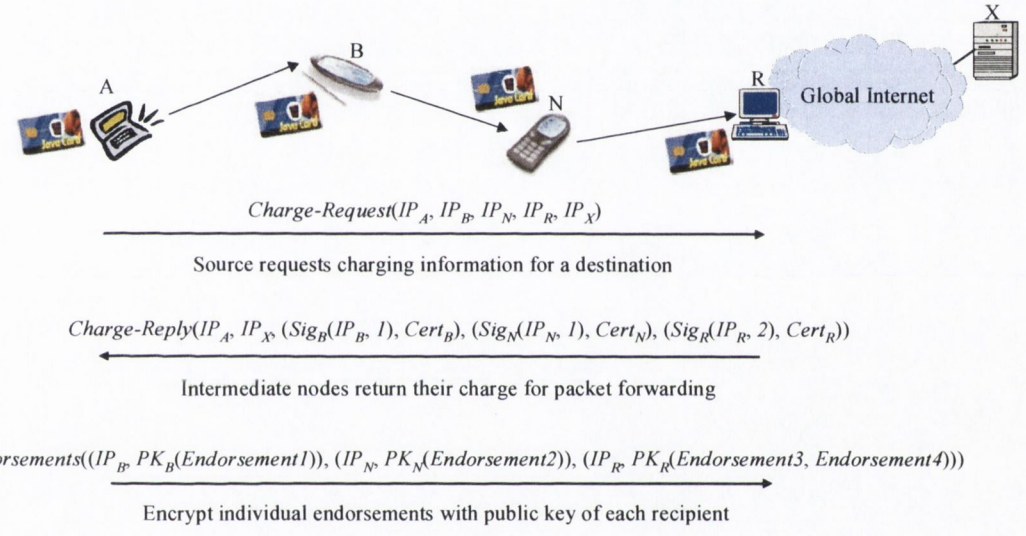


Figure 5-3 Endorsement Distribution

The process of setting up a call to a remote destination requires that the initiating node have knowledge of the total costs involved in forwarding a packet through the ad hoc network. Each node in the path to the destination must indicate its charge for packet forwarding. It is assumed that the source node has knowledge of the routes through the network by making use of an existing reactive or proactive routing protocol. It may then subsequently query nodes along one or more of these routes to obtain the total charge for packet forwarding through the routers along that path. Figure 5-3 shows the overall process of assembling the charging information. In this case, it can be assumed that node *A* has selected a route to the destination node *X* along the path that is served by nodes *B*, *N* and *R*.

Node *A* requests the underlying routing protocol to find a route to the destination node *X* in the fixed network by sending it a *Charge-Request* message. The nodes in the ad hoc network that are in the relay path to the destination attach a *service offering* which can have a number of different security levels. At a very minimum, a service offering consists of a plaintext message with the number of hash tokens or amount that a node requires to forward packets to the next hop. To prevent intermediate nodes from modifying the charge details in their favor, each node can digitally sign the charge-related data with their private key as shown in Figure 5-3. An intermediate node will append one or more certificates so that the source is able to verify the digital signature on the charge details. Finally to prevent replay attacks, a node could add a nonce to the

message before signing it. This information is returned in a *Charge-Reply* message back to node *A*. The general format of a *Charge-Reply* message is of the form *Charge-Reply*(IP_{Src}, IP_{Dest}, (Sig_{Hop_1}(IP_{Hop_1}, *X*), $Cert_{Hop_1}$), (Sig_{Hop_2}(IP_{Hop_2}, *Y*), $Cert_{Hop_2}$) ... (Sig_{Hop_N}(IP_{Hop_N}, *Z*), $Cert_{Hop_N}$)), where *Src* is the node that initiated the communication and *Dest* is the target node. Hop_1, Hop_2 ... Hop_N are the intermediate nodes in the ad hoc network through which the datagram is forwarded. *X*, *Y* and *Z* are the individual node charges for packet forwarding.

In the above case nodes *B* and *N* specify a charge of one hash token per packet, while the gateway, node *R* specifies a charge two hash tokens per packet. Once the source has found the cheapest or most optimum route to the destination, it distributes the broker-signed endorsements to each of the relay nodes in the path. The endorsements are transferred from the secure hardware module in the source node to each of the recipients. Since the endorsements are secret quantities, they are encrypted with public key of each recipient which is obtained from the public-key certificate that was forwarded with the charge reply message.

In Figure 5-3, the *Endorsement1* is encrypted with the public key of node *B*. This allows the source node to spend the first sub-chain ($P1_0$, ... $P1_{40}$) of the UOBT with node *B*. Similarly *Endorsement2* is encrypted with the public key of node *N*. However to node *R*, two endorsements are sent, namely *Endorsement3* and *Enodorsemen4*. This is necessary as Node *R* requires twice as many hash tokens to forward packets into the fixed network. The smart card module in each recipient node will verify the contents of the endorsements and the digital signature of the broker on the same. The general format of this message is *Endorsements*((IP_{Hop_1}, PK_{Hop_1}(*Endorsementnt1*)), (IP_{Hop_2}, PK_{Hop_2}(*Endorsementnt2*)) ... (IP_{Hop_N}, PK_{Hop_N}(*EndorsementntX*))).

Each of the nodes along the path to the destination in the ad hoc network is now ready to accept payment tokens for packet forwarding. Since the initial signaling messages are crucial to the workings of the protocol, the use of a reliable transport protocol such as TCP can be employed to transport the same. This ensures that signaling messages can still be delivered even if packets are dropped due to congestion in the network. Alternatively if the user is restricted to using an unreliable transport protocol such as UDP, then he may transmit the endorsements along with the first three data packets. This will ensure that all intermediate routers will have the required endorsement values to accept payment for packet forwarding. It is assumed that signaling or control messages will be routed through the system free of charge prior to any data transfer. Finally, it should be noted that for very small amounts of data transfer, the signaling overhead associated with distributing the payment parameters could exceed the user data.

5.5.4 Making Payments

For each packet or sequence of packets node *A* wishes to transmit into the network, it attaches a single hash token from the sub-chains $P1_0$, $P2_0$ and two hash tokens from the sub-chain $P3_0$. In order to pay a node multiple hash tokens there is no need to attach multiple hash values. Instead the node can just attach the correct hash value further up the chain. For example, the first packet contains the hash values $P1_1$, $P2_1$ and $P3_2$. Node *B* applies the corresponding hash function to $P1_1$ to obtain the anchor of the chain $P1_0$. It is able to verify that this is the value that was contained within the broker-signed endorsement that was given to it by node *A*. Similarly node *N* is also able to verify the authenticity of the hash token $P2_1$. Node *R* can verify the hash tokens by applying the hash function twice to the value $P3_2$ to obtain the anchor of the chain $P3_0$.

It should also be noted that a node is not required to protect the hash tokens using additional cryptographic procedures. Even though intermediate nodes have access to the hash values they will not be able to redeem them, as they require the corresponding broker-signed endorsement. Under normal operation of the smart card device, the endorsements will remain within the secure hardware module of the smart card and not be

released in the clear. Figure 5-4 depicts the situation where the call has terminated after node *A* has spent 30 hash tokens from the chains *P1* and *P2* at nodes *B* and *N* respectively. At node *R* the chain *P3* has been exhausted and 20 hash values from the chain *P4* have been used. The additional storage requirements at an intermediate node are 184 bytes for each chain that it is willing to accept. This figure comprises of 168 bytes for the broker-signed endorsement and 16 bytes for the highest hash value received by the node. These figures are however subject to implementation details.

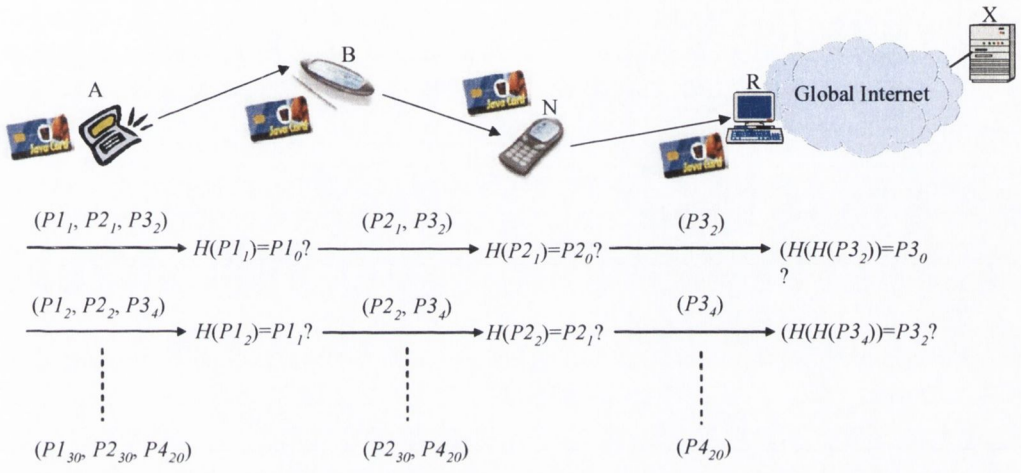


Figure 5-4 Releasing Hash Tokens for Payment

5.5.5 Change in Route – New Path

Changes in the topology of the ad hoc network will result in a change in the route towards the fixed network or other nodes within the ad hoc network. However the user is still required to pay each node along the new path to the destination. The design of the payment system is flexible enough that it is able to cope with this situation without the need to contact a trusted third party or broker. Since the broker has committed to the UOBT that consists of multiple chains, node *A* can immediately start paying each of the nodes in the new path. Figure 5-5 depicts a situation where the new path to the fixed network from node *A* is via nodes *L*, *N* and *R*. As before each node indicates its charge for packet forwarding in the Charge-Reply message. It should be noted that the only new node in the path is node *L*, and that node *A* still has unspent chains with nodes *N* and *R*. Instead of distributing a new set of endorsements to all nodes at the start of a new call, node *A* only distributes endorsements to the new nodes in the path.

In the discussion above node *A* distributes a single endorsement to node *L* and continues using the old chains at the other nodes in the path. At nodes *N* and *R*, node *A* can make use of chains *P2* and *P4* for which the node had previously distributed endorsements. During the last call node *A* used 30 hash values in chain *P2*, which means that there are still 10 unspent hashes left in that chain. Similarly it had 20 hash values left over in the chain *P4*. So in this case node *A* forwards just a single endorsement (*Endorsement5*) to node *L* encrypted with its public key. Along with the first packet node *A* attaches the hash tokens $P5_2$, $P2_{31}$ and $P4_{22}$. Node *L* performs two hash functions on the value $P5_2$ to obtain the anchor of the chain *P5*, which it can verify from the signed broker endorsement. Node *N* performs a single hash operation to obtain the value $P2_{30}$, which it has stored from the last call. Similarly node *R* can perform two hash operations on the value $P4_{22}$ to obtain the highest hash value $P4_{20}$ that it has stored from the last call. If node *A* wishes to continue the

call after it has transmitted 10 packets, it must distribute a new set of endorsements to all the nodes in the relay path.

The proposed solution tries to maximize the number of hash values used in each sub-chain of the UOBT. However node *A* only made partial use of the chain *P5* at node *L* before having to distribute new endorsements to all the nodes in the path to continue the call. An alternative method could be used whereby the node could make use of a chain from a shorter UOBT. A user could purchase multiple UOBTs of different lengths in advance from a broker. Node *A* could then make use a sub-chain from a 20x20 UOBT or two sub-chains from a 10x10 UOBT at node *L*. This would result in no wastage of hash tokens in those chains. This requires additional work on behalf of the mobile node to correctly identify new nodes in the relay path and keep track of unspent hash chains.

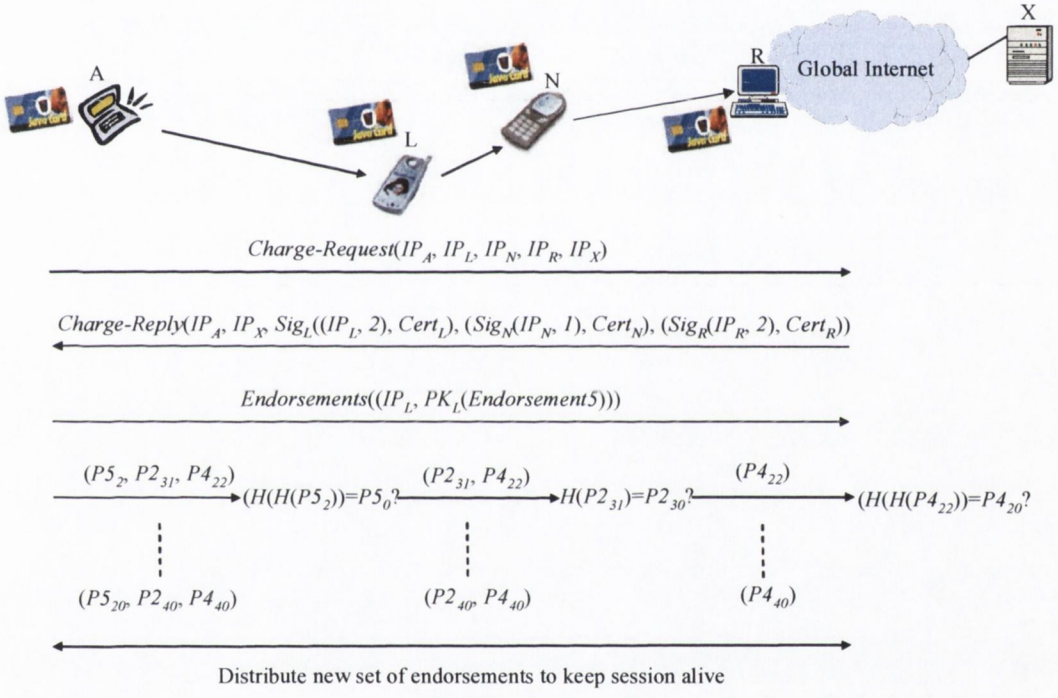


Figure 5-5 Payment of Nodes on New Path

In most cases there will be a bi-directional exchange of packets. For example, when a node requests a Web page from a remote server in the fixed network it will receive data packets in response to that request. The scheme needs to ensure that the nodes in the relay path will forward any response packets back to the source node. Assuming that there are symmetric radio links between neighboring nodes, there are a number of ways in which the protocol can deal with this situation.

The first option is where a payment token allows a node to forward and receive packets through a relay node for a specified amount of time. This method could work well for voice telephony type applications. Alternatively if there is a one-to-one mapping of request and response packets, then it could be assumed that each node on the return path would forward one packet in the reverse direction. However in most situations, a single request may result in a number of packets being sent back as part of the response. To address this problem a node could identify the nature of the traffic during the call setup phase. Each of the nodes the relay path would then send back their charge for packet forwarding based on the characteristics of the call.

5.5.6 Redeeming Tokens

Periodically a node will contact the broker and deposit payment tokens that it has collected for packet forwarding services that it provided to other nodes in the network. In Figure 5-6 node L sends the highest hash value in the chain $P5$ that was spent at it by node A. The user's smart card module sends the hash value $P5_{20}$ along with the corresponding broker-signed endorsement, all encrypted with the public key of the broker. In addition, the smart card digitally signs the message with the private key of node L. The broker verifies the contents of the message, credits the account of node L, and issues a receipt for the same. The general format of the first message is $Redeem\text{-}Token(Sig_{USR_{ID}}(IP_{USR_{ID}}, PX_Y, PK_{Broker}(EndorsementX)))$ and the format of the second message is $Receipt(Sig_{Broker}(IP_{USR_{ID}}, PX_Y, Amount))$. At a later date the broker will also reimburse node A for the remaining 20 unspent hash values of the chain $P5$. The broker can automatically do this once the appropriate chains have expired.

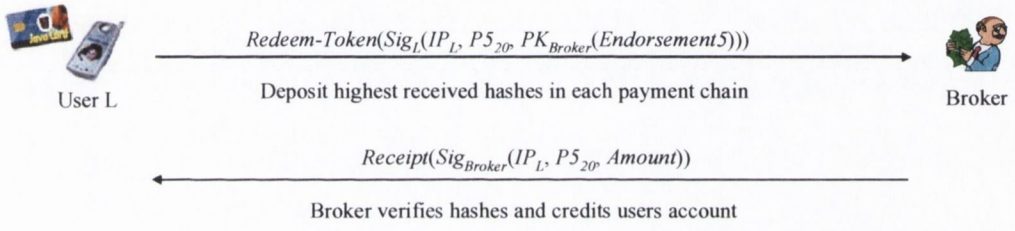


Figure 5-6 Redeeming Payment Hashes

5.5.7 Broker Clearing

The discussions above focused on the use of a single broker in the system. A user usually contacts a local broker and opens an account with him. He receives a smart card module that is pre-loaded with the required root certificates. However the design of the payment protocol is such that there can be multiple brokers in the system and individual users can be associated with any one of them. This allows for a more distributed system and avoids a central point of dependency. Now when a user arrives in a new network, he contacts a local broker and purchases payment chains from him. In addition, he gets a copy of the local root certificates that he may require for verifying digital signatures generated by nodes within the new network. Contacting the broker using a secure terminal e.g. an Automated Teller Machine (ATM) can be used to perform the initial priming operation. The brokers in the system can have accounting relationships and transfer funds between each other to settle user accounts.

5.5.8 Discussion

The proposed solution provides an efficient means of allowing real-time payment to multiple nodes in the relay path between the source and destination. A micropayment scheme using hash functions and offline verification allows the solution to be efficient and scalable. For efficient storage of multiple hash chains on mobile devices the use of unbalanced one-way binary trees was made. To aid performance, public-key algorithms are only used at call setup time and during subsequent endorsement distribution. Note that the existence of a well behaved underlying routing protocol is assumed. However it is possible for a malicious node or a group of cooperating malicious nodes to corrupt the routing information in the network to defraud the system.

On consideration form the network performance point of view is how much of an overhead or latency will be introduced into the network as a result of adding in the payment protocol. In [Mep00], the author has carried

out a comprehensive study of micropayment performance. He has made efficiency comparisons of various hash algorithms versus symmetric and public-key algorithms. One observation is with regards to the number of operations performed per second on a general workstation with a 400MHz Pentium II microprocessor and running the Windows operating system. The author noted that it was possible to perform five times as many MD5 hash computations as DES encryption operations, and over five thousand times as many RSA signature generations. In general the author observed that hashing is an order magnitude faster than symmetric encryption, three orders of magnitude faster than signature verification and four orders of magnitude faster than signature generation. In Chapter 7, a detailed performance analysis on a number of well known cryptographic algorithms on a WinCE Pocket PC device has been carried out. The results obtained were broadly in line with the figures above.

The proposed multi-party micropayment scheme has been designed to be independent of the underlying routing protocol. This is essential as many types of routing algorithms have been defined, and depending upon the type of network or environment one or more of them may be used. The use of long-lived payment contracts which bind all the nodes in the payment path [Mep00] has been eliminated, as this model does not fit well in an ad hoc network scenario. Instead, the charging information from each node in the relay path is securely obtained and a total charge for forwarding a packet through the network is assembled. Without access to an online TTP, the only other solution is to make use of secure tamper-resistant hardware modules. However the proposed solution does not place total trust on the hardware modules contained within each node. If the tamper resistance of one of these devices is compromised the system will suffer from a limited amount of fraud. The upper bound for this is the expiry date associated with the broker-signed endorsement. It is assumed that as these devices become more prevalent and powerful, that compromise of their security features will continue to be an extremely difficult task.

Hash chains are of a finite length and there is a possibility that a node may run out of hash values during a session. In the case of a UOBT if there are unused sub-chains, then the user can switch to the next chain by sending a new set of endorsements. If there were no further sub-chains available, then this would result in the dropping of a connection. This would be particularly true for real-time communications such as voice telephony or video conferencing where the source may transmit a large number of datagrams during a session. However this applies to all payment protocols that have a fixed amount in the user's purse. Alternatively, at call setup time, a value which corresponds to the number of packets that an intermediate node will allow, before expecting the next hash value in the chain to be released by the source can be negotiated. For example, each hash value could allow a source to transmit a hundred packets. This could be implemented simply as a counter at each node.

Each time there is a change in the topology in the ad hoc network, broker endorsements have to be distributed to each of the new nodes in the path. The amount of wastage in the system is dependent on two factors, namely node mobility and chain lengths. If there is high mobility in the network then a node should make use of small chains e.g. chains from a 10x10 UOBT. On the other hand if the routes in the network were relatively stable then it would make sense to use chains of longer lengths. One possibility is for a node to purchase UOBTs of different lengths from a broker. Since each hash value may be worth a tenth of a cent or less, the total value of the tree will not amount to a large sum. Depending on the network circumstances a node could then make use chains from the appropriate tree.

It should be noted that the introduction of monetary rewards in the system may not act only as an incentive for collaboration, but also as an incentive for cheating. Thus any payment system needs to be examined for possible attacks. Since this work concerns the area of ad hoc networks where all communications are wireless in nature and there is no trusted third party or central administrative authority. Extra care has to be taken of

the possibility of the presence of malicious nodes which may try to defraud the system. An informal security analysis of the protocol has highlighted a number of vulnerabilities. Malicious nodes can selectively forward longer routes or routes which contain other collaborating nodes to charge more from the customer. A more sophisticated attack could be mounted whereby colluding nodes could set up a forwarding loop to defraud others and accumulate tokens. Also, the protocol does not allow for reimbursement of charges upon non-delivery to the destination. A malicious node could occasionally drop packets citing network congestion and still charge for partial forwarding. Some of these concerns are addressed in Chapter 6, where a secure routing and packet forwarding scheme for mobile ad hoc networks is presented.

With regards to an ad hoc network scenario, the assumption that there is a global PKI which is always available is not realistic. Proposals have been but forward for a distributed public key management scheme based on *threshold cryptography* [ZH99]. However this requires a threshold number of nodes to be able to generate a certificate. In practice, most schemes will rely on a node to be primed with a number of locally relevant cryptographic keys and associated certificates, prior to it's joining an ad hoc network. In some cases it may not be possible for the node to join a network as it may not possess the required keys. In such instances, the onus will be upon the user to obtain the relevant information by bringing the node in contact with a relevant TTP. Also the limited storage capabilities and processing power of such mobile nodes will play an important role on the development of ad hoc networks as a mainstream networking paradigm.

5.6 Implementation Details

In order to evaluate the proposed mobile ad hoc payment protocol version 2.2.6 of the *ns-2* simulator was used. The NS simulator has been extensively used in evaluating the performance of ad hoc network routing protocols, and has support for routing protocols such as AODV, TORA and DSR. The NS simulator models radio propagation using the *two-ray ground reflection* model which accounts for physical phenomena such as signal strength, propagation, and interference. The medium access control protocol used is the IEEE 802.11 Distributed Coordination Function (DCF).

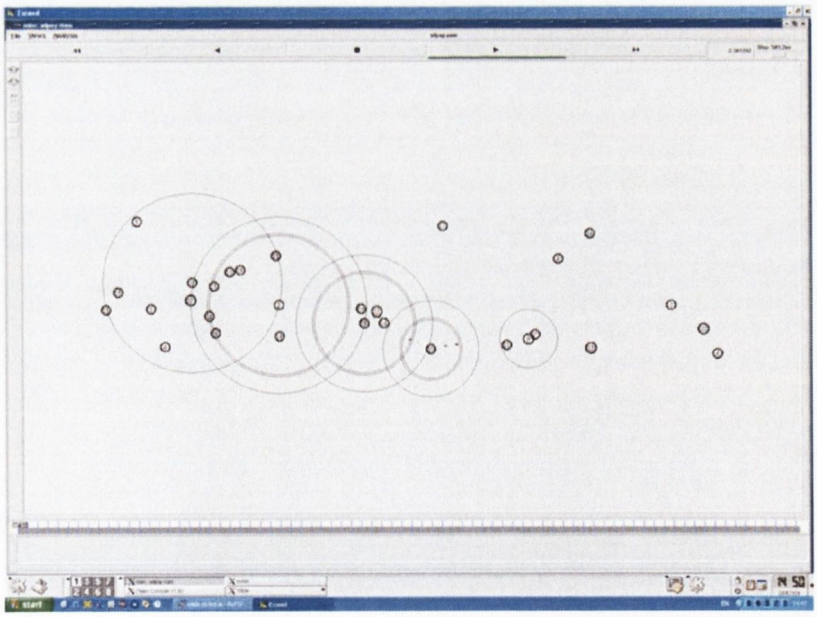


Figure 5-7 Nam Screenshot of an *ns-2* Simulated AdPay Network

Two parallel installations of the NS code base were created for evaluation purposes. The first installation was the unmodified DSR code which is simply referred to as *DSR*. In the second installation, the base DSR protocol was modified to include the payment extensions, and made use of the OpenSSL libraries (version 0.9.7a) to implement the various cryptographic procedures. The modified mobile ad hoc payment protocol is referred to as *AdPay*. The implementation does not employ the use of a broker to sign the anchors of the UOBT. Instead the anchors of the tree at the source node are signed with the private key of the broker, which is made available to the same. It is assumed that obtaining the broker commitment is an offline procedure and does not factor in the delay associated with the same into the AdPay protocol.

The implementation employs a single Constant Bit Rate (CBR) source (node 0) and destination (node 9) pair. This is in contrast to a real network scenario where there may be multiple sources and destinations. This simplification was opted for as there is a requirement to distribute and verify payment tokens in the AdPay simulation. Multiple sources would require flow state identifiers to be carried along with each packet and to also be maintained in the relevant nodes in the network. This would have made the prototype development task much more complex and time consuming.

Also, no special control messages were employed for distributing the broker-signed endorsements. Instead, once a route has been found using the usual route discovery mechanisms, the endorsements are sent in the data packet transmitted by the source node. Each node along the way verifies its assigned endorsement(s) and adds some processing delay to the network. Unlike the base DSR protocol, the AdPay scheme refrains from sending further data packets before the destination node has received the first data packet, and all intermediate nodes have obtained the anchor(s) of the relevant UOBT chains from the broker-signed endorsements. Figure 5-7 shows a screenshot of an ad hoc network simulation in a Nam console. It should be noted that the AdPay prototype developed in this chapter is a purely proof of concept implementation. A detailed security evaluation will be required prior to deploying the protocol in a live network scenario.

5.7 Experiments and Measurements

As highlighted in Chapter 4, NS does not take into account processing delays that may be experienced at a node. However the verification of charge details, distribution and verification of endorsement tokens, and the verification of payment tokens at the various nodes in the path from the source to the destination, adds to the processing overhead to the original protocol which must be taken into account. Again the Unix *gettimeofday()* library function was used to obtain the timing measurements and the cryptographic processing delay was added to the NS scheduler. Adding in the processing delay at each node had an effect on the average end-to-end delay or latency and the number of successfully delivered packets.

5.7.1 Experimental Setup

A 400MHz Pentium II with 256MB of memory running the Linux 2.2.20 kernel was used as the testbed machine. Each node in the simulation moves according to the *random waypoint* model. In this model a node starts at a random position and waits for a duration called the *pause time*. It then chooses a new random location and moves there with a velocity uniformly chosen between *0* and V_{max}. When it arrives at its destination, it waits for the pause time and repeats the process.

Like much of the previous work [HPJ02, PH02b, GIS01, BA01], a number of standard network parameters specific to NS were used. A standard rectangular space of size 1500m x 300m was used to increase the average number of hops in routes used relative to a square space of an equal area. This allowed for a more challenging environment for the routing protocol. The number of nodes in the simulation was varied between 10 and 50. With 10 nodes representing a small network, 30 representing a medium sized network, and 50 nodes representing a large network. Finally, the pause times were varied between 0 and 900 seconds. Zero

seconds represents continuous movement, while 900 represents no movement, as the duration of each simulation is also 900 seconds. Table 5-1 shows the parameters used in the simulation.

Number of Nodes	10, 30 or 50
Max Velocity (v_{max})	20 meters/s
Coverage Area	1500m x 300m
Nominal Radio Range	20 meters
Source Data Rate	4 packets/second
Application Data Payload Size	512 bytes/packet
Raw Physical Link Bandwidth	2Mbps

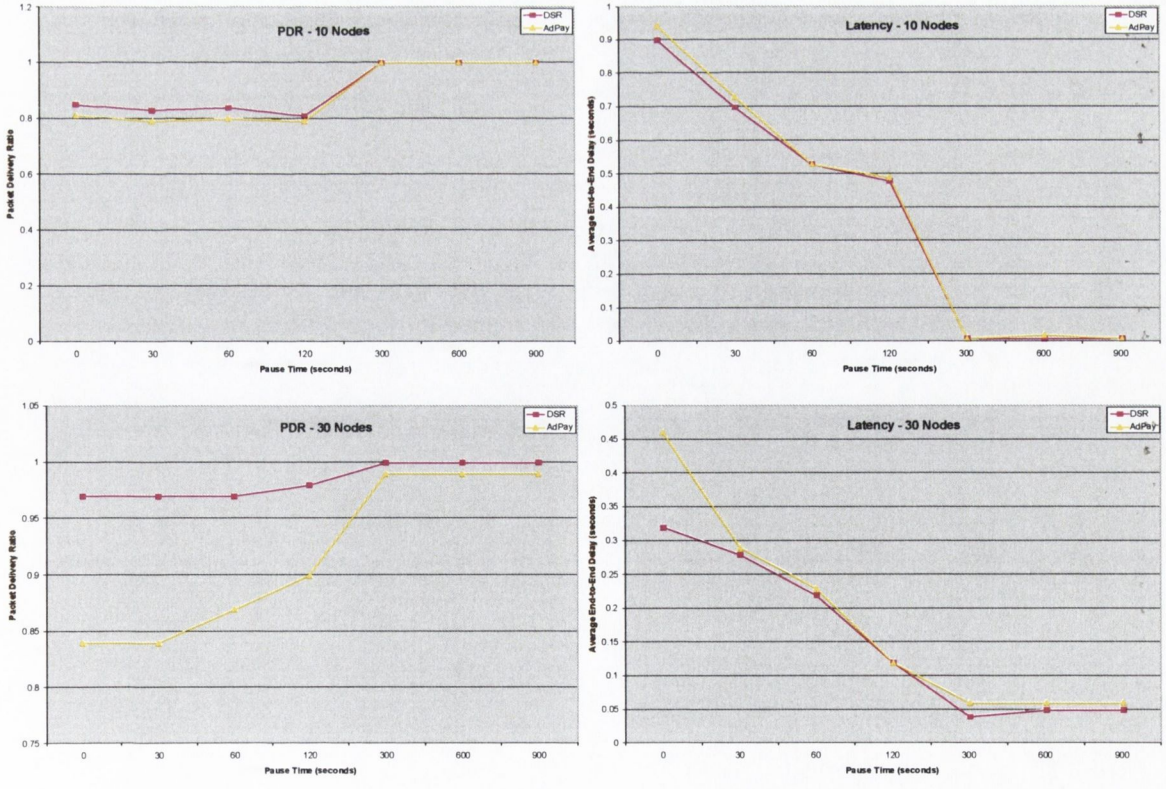
Table 5-1 Parameters for AdPay Simulation

5.7.2 Performance Metrics

Both the DSR and AdPay protocols were run on identical movement and communication scenarios. Each combination of node and pause times was run for 50 separate movement scenarios and an averaging of the results was performed. The performance of the protocols was measured along two metrics:

- **Packet Delivery Ratio (PDR)**: The fraction of application data packets sent that are actually received at the destination node.
- **Average End-to-End Delay (Latency)**: The average time elapsed from when a data packet is first sent to when it is first received at the destination.

5.7.3 Evaluation of Results



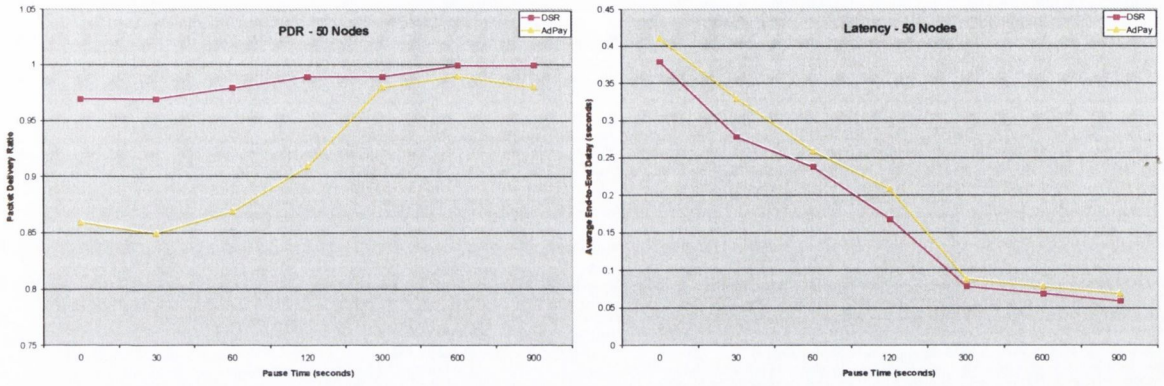


Figure 5-8 Performance Results Comparing AdPay with the Standard DSR Protocol

Figure 5-8 shows the packet delivery ratio and latency values for a network of 10, 30 and 50 nodes respectively. It can be seen that for a small network of 10 nodes the PDR for AdPay mirrors the DSR version. At very low mobility rates (simulation time of 300-900 seconds) a PDR of 100% was achieved. For medium and large sized networks, the PDR for the AdPay protocol is lower than that of DSR when the mobility in the network is high. One reason for this may be the fact that the AdPay protocol has removed the DSR optimization which allows intermediate nodes that have a valid route to the target node in their route cache to return a RREP, as it is required to obtain a charge for packet forwarding by each node in the relay path. Also each node needs to make use of public-key cryptography techniques in assembling the charge reply parameters. The PDR tends to return to a 100% delivery ratio when the mobility rates drop. The latency values are only marginally higher in all three network cases and tend towards zero when the mobility of the nodes becomes low. The data for the various graphs can be found in Appendix C of this thesis.

From the above results it can concluded that the AdPay protocol performs well in relation to the base DSR protocol in terms of packet delivery and end-to-end delay. With the added advantage that intermediate nodes are able to collect and verify in real time, payment tokens attached by the node that wishes its packets to be forwarded in the network. The main reason for this is the use of lightweight cryptographic procedures in the AdPay protocol.

5.8 Summary

With emergence of more mainstream type applications for ad hoc networks, there is a need to move away from the notion of closed user groups, and the assumption that all nodes in an ad hoc network will cooperate with each other for benefit of the network as a whole. The current proposals for stimulating cooperation in ad hoc networks were examined and a number of drawbacks were identified. It was concluded that a real world payment system could provide a good solution to resolving the problem of reliable forwarding of packets in ad hoc networks. The desirable properties of a payment system were then drawn up, and a solution which would achieve those goals was proposed.

In this chapter a method for compensating nodes for packet forwarding in an ad hoc network has been presented. The proposed architecture allows routers to charge per-packet and adapts to routing changes. A node receives one or more routes to a destination along with a secure and verifiable charge for packet forwarding by each intermediate router. Based on this information the node can choose a suitable route and distribute endorsement values securely to each node in the path. The nodes are able to verify the same and can immediately start accepting payment for packet forwarding. Changes in the route between the source and

destination do not require the source to contact a third party, such as a bank or broker to obtain payment tokens for the nodes in the new path. It is able to immediately use additional chains to pay the new nodes in the routing path. The experimental results have shown that the AdPay protocol has minimal overheads and compares favorably in relation to the base DSR protocol.

The proposed protocol is different from the existing approaches for stimulating cooperation in mobile ad hoc networks in a number of ways. Instead of using a counter or some other non-monetary form of incentive, a real payment system to reward nodes for packet forwarding in the network is employed. All the nodes in the network are paid in real time without the need for long-lived payment contracts or the need for postfact settlement of accounts. The AdPay protocol can function in a totally pure ad hoc network scenario as it does not impose any restrictions in terms of having a connection to an online TTP, or the requirement for fixed nodes to be part of the relay path between the source and destination nodes.

Finally, in Section 5.5.8, some security concerns were highlighted with regards to misbehaving nodes which may inject erroneous routing information in order to misdirect packets in the network, or selectively drop packets that are not destined for them. In Chapter 6, many of the well known attacks that can be carried out by malicious nodes in an ad hoc network are presented. A number of these attacks are addressed by developing a secure routing and packet forwarding scheme.

6 Secure Routing and Packet Forwarding in Ad Hoc Networks

"Take nothing on its looks; take everything on evidence. There's no better rule."
From "Great Expectations", by Charles Dickens (1812-1870)

6.1 Introduction

Current ad hoc routing protocols implicitly trust all the participants to cooperate in the relaying of datagrams in the network, and are not able to cope with network disruptions due to the malicious behavior of one or more of the participants. An attacker for example can masquerade as a legitimate node and advertise false or non-existent routes in the network. More sophisticated attacks can be mounted whereby a node selectively drops datagrams and points to collisions on the wireless transmission medium as the probable cause. Hence there need to be mechanisms in place to monitor such malicious activity, in order to be able to identify nodes that misbehave, and subsequently blacklist them from the network. However the lack of infrastructure and access to online TTPs makes this task difficult in ad hoc networks. Also, the limited capability of mobile devices means that use of heavyweight cryptographic algorithms, such as the IETF AAA and IPsec protocols highlighted in Chapter 2 required for authenticating node identities and securing routes becomes impractical.

In wired networks such as the Internet, the routers within the core network are administered by well known network operators and are therefore assumed to be trustworthy. However in ad hoc networks where each node acts as a router, a user needs to be able to authenticate the identity of each of the nodes in the path during the route discovery process, to prevent attackers from advertising non-existent routes. By injecting erroneous routing information or simply replaying old routing information, an attacker can successfully partition the network into isolated subnets, in order to prevent one set of nodes from reaching another [DLA02]. Such attacks are also referred to as *routing-disruption* attacks. An attacker can flood the network with excessive route requests that can overload the transmission medium and hamper the normal flow of traffic within the network. This is also known as the *resource-consumption* attack. Nodes that receive such route requests and blindly forward them without verifying that they originated from a valid source can unwittingly aid an attacker in the network. In addition to authenticated route discovery, guaranteed datagram forwarding is also required in ad hoc networks, as selfish nodes may decide not to participate in the packet relaying process so as to save processing time or battery life.

An attacker may simply create a routing *black hole* which attracts and drops data packets. A more sophisticated attack can be mounted whereby a node selectively drops packets or denies having received a packet due to transmission collisions in the network. This last attack is also referred to as creating a *gray hole* [HP04]. Finally, a more subtle type of routing-disruption attack is to create a *wormhole* in the network using a pair of attacker nodes A and B, linked via a private network connection. In the wormhole attack, the

attacker records packets at one location, tunnels them to another location, and retransmits them from there onwards into the network. The presence of a wormhole for example could lead two nodes to believe that they are neighbors, even though they may be multiple hops away from each other. Once these nodes start using the wormhole, the attacker could selectively drop packets which could severely disrupt communications.

In this chapter, a proposal for a secure routing protocol which allows for authenticated route discovery and datagram delivery is presented. The scheme is designed primarily to address the gray hole attack in ad hoc networks. The scheme makes use of the *optimal hash traversal* technique [Jak02] outlined in Chapter 3. Each node monitors the network constantly for dropped packets and maintains a rating for each active node that it has knowledge of in the network. The protocol is able to use this information to rate end-to-end paths in the network. Specifically, the protocol allows a node to cryptographically prove to others in the network that a particular node is misbehaving and subsequently disqualifies it from the network.

The rest of this chapter is organized as follows. The related work in the area of secure routing for ad hoc networks is presented in Section 6.2. The assumptions that are made about the nature of the wireless network and the design goals of the protocol proposed in this chapter are listed in Section 6.3. In Section 6.4, the system model upon which the protocol is based is outlined. A detailed account of the system is given in Section 6.5. Sections 6.6 and 6.7 provide the implementation details and experimental results.

6.2 Related Work

A number of protocols have been proposed for secure route discovery in ad hoc networks. The Secure Routing Protocol (SRP) is a set of security extensions that can be applied to any ad hoc routing protocol that uses broadcasting as its route querying method [PH02a]. The authors mention DSR as a particularly appropriate protocol for incorporating their proposed security extensions. The SRP scheme guarantees that a node initiating a route discovery process will be able to identify and discard replies providing false topological information. It requires that the initiator (node A) and destination (node D) nodes share a pre-established Security Association (SA), such as a shared key K_{AD}.

The source node initiates a route discovery process by constructing a Route Request (RREQ) packet identified by a pair of unique identifiers. The identifiers plus the non-mutable fields of the IP header are used as input for calculation of the Message Authentication Code (MAC) along with the key K_{AD}. In addition, the identities of the traversed intermediate nodes are accumulated in the route request packet. The RREQ reaches the destination which constructs a Route Reply (RREP). It calculates a MAC covering the RREP contents and returns the packet to the initiator over the reverse of the route accumulated in the respective request packet. SRP will not accept replies from intermediate nodes, as they do not have knowledge of the key K_{AD}, and thus will not be able to create a valid MAC. It does not attempt to prevent unauthorized modifications of fields that are ordinarily modified in the course of forwarding these packets. For example, a node can freely remove or corrupt the node list of a RREQ packet that it forwards. SRP does not attempt to address the route maintenance question and route error messages generated by intermediate nodes are retained in SRP.

The Authenticated Routing for Ad Hoc Networks (ARAN) is based on the AODV protocol [SDLS+02]. In ARAN, each node has a digital certificate signed by a trusted authority which binds the IP address of the node with its public key. During a route discovery process, the initiator (node A) sends a signed RREQ packet that includes amongst other things the destination address, a nonce, a timestamp and the certificate of the initiator. The initiator's immediate neighbor (node B) verifies the signature and signs the contents of the message. It appends its own certificate and rebroadcasts the message. Node C which receives the packet next verifies node B's signature first and then verifies the signature of the initiator. Node C removes node B's signature and certificate. It then signs the message with its private key, and appends its own certificate and

rebroadcasts the message. When the RREQ reaches the target node (node *D*), it signs a RREP and unicasts it to the node that sent the RREQ. The RREP is forwarded in much the same way as the RREQ, with each node that verifies the RREP adding an entry for the target node into its routing table.

ARAN is vulnerable to Denial of Service (DoS) attacks based on flooding the network with false control packets for which signature verification are required. Nodes that are not able to verify the signatures with a reasonable timeframe may be forced to drop packets [HP04]. Certificate revocation can also be problematic, especially if the blacklisted node is the sole connection between two parts of the ad hoc network. In such a situation the blacklisted node may not forward the notice of revocation for its certificate, resulting in a partition of the network, until the blacklisted node is no longer the sole connection between two parts of the network. ARAN is not suitable for source routed protocols such as DSR.

Secure AODV (SAODV) is another proposal designed to secure AODV [ZA02]. The basic idea behind SAODV is to use a digital signature to authenticate most of the fields of the RREQ and RREP, and to use hash chains to authenticate the hop count. Every time a node originates a RREQ or a RREP packet, it generates a new hash chain, the length of which is equivalent to the maximum hop count. The initiating node adds the anchor of the chain to the packet header and signs the message with its private key. It also appends the field *Hash* which is the root of the chain to the message extension header.

In SAODV every time a node receives a RREQ or RREP message, it applies the same hash function *Max_Hop_Count – Hop_Count* times to the Hash field, and verifies that the resulting value is equal to the anchor of the chain included in the digital signature. Prior to re-broadcasting the packet the node hashes the Hash field in the message extension header, in order to account for the new hop. The hop authenticator reduces the ability of a malicious intermediate node from modifying the hop count arbitrarily without detection. Since the protocol uses digital signatures, it requires the existence of a key management scheme or a CA in order to verify the public keys of other nodes that participate in the network. The main differences between ARAN and SAODV are that ARAN uses an extra signature to authenticate the previous hop, and SAODV uses a hash chain to authenticate the metric [HP04].

The Secure Efficient Ad hoc Distance vector (SEAD) is a secure ad hoc network routing protocol based on the design of the Destination-Sequenced Distance-Vector (DSDV) algorithm [HJP02]. In order to find the shortest path between two nodes, distance vector routing protocols utilize a distributed version of the Bellman-Ford algorithm. Each entry in a node's routing table contains the address of some destination, the node's shortest distance called the *metric* (usually in number of hops), and the address of the node's neighbor router that is the first hop on the shortest route to the destination. The SEAD protocol employs the use of hash chains to authenticate hop counts and sequence numbers. The protocol assumes that there is an upper bound m on the diameter of the network. Prior to sending a routing update, a node generates a hash chain of length n, such that n is divisible by m. It then distributes the anchor of the hash chain to all other nodes in the network. For authenticating a routing update with sequence number i and metric j, it releases the hash token $h_{n-i*m+j}$.

For example, if a node lists an entry for itself in that update, it sets the address in that entry to its own address, the sequence number to its own next sequence number, the metric to 0, and the hash value to the first element in the group of its own hash chain elements corresponding to that sequence number. If the node lists an entry for some other destination in that update, it sets the address in that entry to that destinations node address, the sequence number and metric to the values for that destination in its routing table, and the hash value to the hash of the hash value received in the routing update entry from which it learnt the route to the destination. This use of a hash value corresponding to the sequence number and metric in a routing update

entry prevents any node from advertising a route to some destination claiming a greater sequence number greater than that destinations own current sequence number, due to the one-way nature of the hash chain. Likewise, no node can advertise a route better than those for which it has received an advertisement, since the metric in an existing route cannot be decreased.

In order to avoid DoS attacks, a receiving node can specify the exact number of hashes it is willing to perform for each authentication. The SEAD protocol proposes two different methods in order to authenticate the source of each routing update. The first method requires clock synchronization between the two nodes that participate in the ad hoc network, and employs a broadcast authentication mechanism such as Telsa. The second method requires the existence of a shared secret between each pair of nodes. This secret can be utilized in order to use a MAC between the nodes that must authenticate a routing update message.

Ariadne is another secure on-demand routing proposal by the authors of the SEAD protocol. It is based on DSR and relies on symmetric cryptography techniques [HPJ02]. It assumes the presence of bidirectional links and can authenticate routing messages using one of three schemes: shared secrets between each pair of nodes, shared secrets between communicating nodes combined with broadcast authentication, or digital signatures. The authors prefer the use of the broadcast authentication scheme Telsa, which requires loose time synchronization between the nodes in the network. For an initiator (node A) to be able to perform an authenticated route discovery to a target (node D), they must share the secret keys K_{AD} and K_{DA}, respectively, for message authentication in each direction. The protocol also assumes that each node also generates a Telsa one-way hash chain, and that all nodes know an authentic key of the hash chain (e.g. the anchor of the chain) of each other node.

In Ariadne, the initiator of the RREQ (node A) creates a MAC h_0 on a number of parameters and uses the key K_{AD}. When node B receives a RREQ for which it is not a target, the node removes h_0 and adds its own address to the RREQ node list. It creates a hash $h_1 = H(B, h_0)$ and adds h_1 to the RREQ. Node B then creates a new MAC on the RREQ using the Telsa key K_{B_1}. Each intermediate nodes repeats the process until the packet reaches the target node. The target node verifies the RREQ by determining that the keys from the time interval specified have not been released, and that nodes in the path are in the hash chain list $H(C, H(B, \text{MAC}_{K_{AD}}(\ldots)))$. Before returning a RREP, the target node waits for the intermediate nodes to release the corresponding Telsa keys. The most important requirement for Ariadne is the existence of loose clock synchronization in the ad hoc network. Also, the delayed release of Telsa keys by the intermediate nodes could slow down the performance of the network and will increase the network traffic.

In [MGLB00], the authors propose a system in which a node can detect misbehavior in the network and subsequently report this to other nodes by making use of the *watchdog* and *pathrater* functions. Each node maintains a rating for nodes in the network that it has knowledge of through the route discovery process and chooses a route which comprises nodes with the highest rating. CONFIDANT [BB02] is another scheme based on DSR which consists of four components: the monitor, the trust monitor, the reputation system, and the path manager. The components are present in every node. For each packet a node forwards, the monitor on that node attempts to ensure that the next-hop node also forwarded the packet correctly. When a monitor detects an anomaly, it triggers action by the reputation system, which maintains a local ratings list. The trust monitor is responsible for handling rating lists from other nodes in the network. If a list is received from a highly trusted node, the trust monitor can directly place the information from the list into its local ratings list. On the other hand, if a list is received from an untrusted source, the recipient can totally ignore it or give it substantially less weight than a list received from a more trusted node. Finally, the path manager chooses paths from the node's route cache based on a blacklist and the local ratings list.

Misbehavior detection schemes in principle can detect wormhole type attacks. If the watchdog or monitor sees a routing message with *Hop_Count* = *X+1* being sent by a neighbor, but did not see a message with *Hop_Count* = *X* being sent by the same node, then the node is either fabricating the routing message or there is a tunnel. The main drawback of these approaches is that there are no mechanisms for verifying the integrity and authenticity of such reports by a third party in the network. An attacker can easily use the feedback mechanism to blacklist legitimate nodes in the network [ZA02].

6.3 System Model

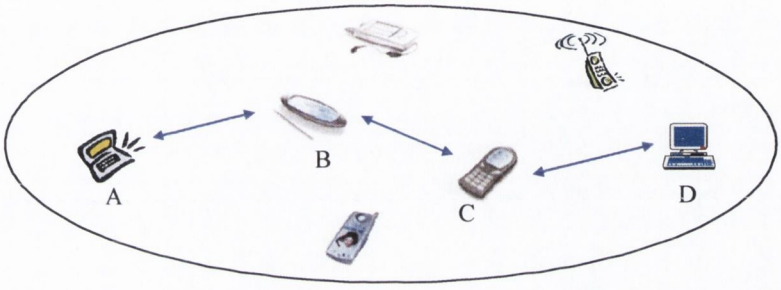


Figure 6- 1 Secure Routing in Ad Hoc Networks

A scheme that allows secure routing and packet forwarding in ad hoc networks is now proposed. The scheme enables a node initiating a route discovery process to be able to identify and discard any replies that provide false topological information. Intermediate nodes in a path examine the cryptographically signed routing data within each signaling packet, and do not propagate packets that cannot be verified. The protocol has been designed to prevent would be attackers from flooding the network with false route request or reply packets in order to circumvent the normal route discovery process. Furthermore, the protocol allows a node to identify and subsequently provide cryptographic proof of misbehaving nodes in the network that may selectively drop datagrams to disrupt traffic flows.

The protocol also has built-in safeguards to prevent attackers from exploiting the protocol itself which could lead to DoS attacks. Wireless communications are prone to external interference such as noise that may result in transmission collisions. Therefore the protocol does not immediately blacklist a node as being malicious when one or more datagrams are dropped. Instead the scheme makes use of a path rating algorithm, and only excludes a node from the network when there is sufficient proof that the node has been deliberately misbehaving. Figure 6-1 shows an example of an ad hoc network where node *A* has found a route to node *D* via nodes *B* and *C*.

A node wishing to find a route to a destination initiates a route discovery process by making use of the underlying ad hoc routing algorithm. Along with each route request the initiator sends a *signed identity token* which allows nodes in the path to the destination to verify the identity of the initiator. A node may decide not to forward packets from one or more nodes in the network that generate excessive RREQs. Each node in the path to the destination co-signs the initiator generated identity token with their private key and appends their public-key certificate. The target node on receiving the RREQ verifies each of the signatures on the identity token and generates a route record of the nodes on the path. It signs the route record with its private key and sends a RREP to the source of the request. Nodes on the reverse path verify the route reply before allocating a new hash chain for that particular flow, and attach the signed anchor of the chain prior to forwarding the packet. A flow is identified by a unique flow identifier ($Flow_{ID}$) and prevents the message being replayed by a malicious node at a later stage.

Each node also runs a Path Rating Agent (PRAT) that maintains a metric associated with nodes in the network that it has knowledge of, and updates this metric for every failed transmission in the network. Once the source node has one or more routes to a destination it chooses the one with the highest routing metric. It starts forwarding packets over that path and monitors the network traffic of its immediate neighbor on the path. By promiscuously listening to its neighbor's transmission it is able to ascertain if the neighbor relayed a datagram and updates its authentication cache (AuthCache). All other nodes in the path operate in a similar manner, and if a node drops a packet either accidentally or maliciously, its neighbors generate a signed alert (*Alert*) informing the initiator of the same. If the threshold associated with a node is breached, then the PRAT will assign a negative rating for that node, and will not use any paths in which that node exists. It will subsequently report this to its CA server and forward all signed messages that it received from other nodes in the network as proof of the same.

6.4 Protocol Goals

The protocol goals for the proposed scheme are now presented. It is assumed that any good secure routing protocol must satisfy the following minimum requirements:

Authenticated Route Discovery – A node wishing to find a route to a destination must be able to authenticate the identity of each node in the chosen path. To prevent an attacker from flooding the network with false route requests or replies each node must authenticate a datagrams contents prior to forwarding it.

Identify Misbehaving Nodes – There is a need to identify nodes that may selectively drop packets in the network to disrupt traffic flows. However, concrete proof of the same is required to be presented to other nodes before they will accept such assertions. Otherwise an attacker may try to exclude nodes from the network by reporting them as having been misbehaving.

Identify Reliable Paths – Each node should maintain a rating for all other nodes that it encounters in the network through the route discovery process. It should make use of the highest rated path to the destination even if a shorter path exists. This should result in a higher percentage of datagrams being delivered to their destination in the face of adversaries in the network.

Lightweight Cryptographic Procedures – Nodes must be able to quickly verify the authentication information carried within datagrams. Excessive use of heavyweight cryptographic algorithms and functions may result in delays in packet forwarding by intermediate nodes.

6.5 Protocol Design

The details of the protocol are now provided in the following sections:

6.5.1 Roles, Requirements and Assumptions

The main entities involved in the various protocol exchanges are the mobile nodes in the ad hoc network. The roles and requirements for the mobile nodes in the system are outlined below:

Mobile Node – The protocol allows a mobile node in an ad hoc network to pay each node in the path towards the destination in real time for packet forwarding. Each node in the ad hoc network is required to generate or securely obtain a public-key pair, and to subsequently obtain a public-key certificate for the same from a TTP in the network along with the public-key certificate of the TTP. The public-key pair is used for performing encryption, decryption and digital signature operations using the RSA algorithm. The cryptographic keys are

stored in a secure tamper resistant device such as a smart card, and all cryptographic functions are executed securely within this device. This ensures that the cryptographic keys and other payment related parameters never leave the secure smart card environment. The smart card must also be able to perform hash operations using the MD5 and SHA-1 algorithms.

A number of important assumptions are made in order for the correct operation of the protocol. It is assumed that all wireless network links are bidirectional and that nodes operate in *promiscuous* mode in order to overhear their neighbor transmissions. For example, if node *B* is communicating with node *C*, node *A* which is in the wireless range of node *B* can overhear its transmission to node *C*. The proposed scheme is best suited to *source-routed* protocols such as DSR [JMH03]. However unlike the DSR specification the protocol does not make use of intermediate cached routes. Thus if node *A* requires a route to node *D*, it will obtain an authoritative signed reply from the same, and will not accept replies from intermediate nodes such as *B* or *C*. The scheme also assumes that multiple nodes in the network do not collude together to cause havoc by advertising false routes or dropping packets. This last assumption however cannot be guaranteed, which is an important consideration in a real world scenario. Another important assumption that has been made is that there is a PKI in place which can be contacted in an offline manner, and that all nodes have a public-key certificate signed by a TTP in the system. It is assumed that these CAs are highly trusted entities and implement strict security policies in relation to storing root keys and certifying individuals or entities in the system. It is also assumed that the smart card device is tamper resistant and that the effort required to break the security of the device will far outweigh the subsequent financial gains.

6.5.2 Authenticated Route Discovery

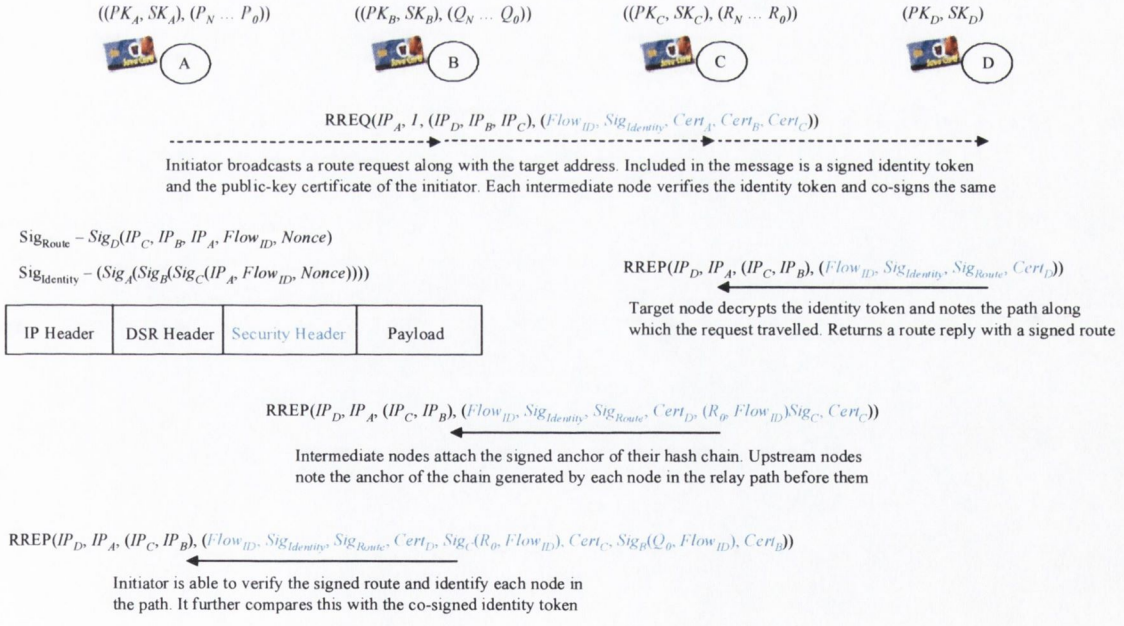


Figure 6-2 Authenticated Route Discovery

The protocol employs an additional header referred to as the *security header* which carries all security related parameters is employed. The security header sits between the DSR header and the datagram payload as shown in Figure 6-2. As per normal DSR routing, the initiator node *A*, broadcasts a RREQ with node *D* as the target node. In the security header the initiator sends a signed identity token ($Sig_{Identity}$), which consists of a

digital signature on the concatenation of its network address and the $Flow_{ID}$. Node A also includes its certificate in the security header. The general format of a route request is $RREQ(IP_{Src}, 255.255.255.255, (IP_{Dest}, IP_{Hop_1}, IP_{Hop_2} \ldots IP_{Hop_N}), FlowID, Sig_{Src}(Sig_{Hop_1}(Sig_{Hop_2} \ldots Sig_{Hop_N}(IP_{Src}, FlowID, Nonce))), Cert_{Src}, Cert_{Hop_1}, Cert_{Hop_2} \ldots Cert_{Hop_N})$, where Src is the node that initiated the communication and $Dest$ is the target node. $Hop_1, Hop_2 \ldots Hop_N$ are the intermediate nodes through which the datagram is forwarded.

Intermediate nodes in the path to the target co-sign the identity token and attach their certificate to the security header. The size of $Sig_{Identity}$ remains constant and is equal to the size of the private key which is typically 1024 bits. Node D on receiving the RREQ verifies all the signatures on the identity token, and ensures that none of the nodes in the path to the initiator are part of its current Certification Revocation List (CRL), or have a negative rating. Based on the sequence of signatures on the identity token, node D creates a signed route (Sig_{Route}). Node D sends a RREP to the initiator in which the security header consists of the $Flow_{ID}$, $Sig_{Identity}$, Sig_{Route} and $Cert_D$, its public-key certificate.

Each intermediate node on the reverse path generates a new hash chain for this flow and attaches the signed anchor of the chain to the security header along with its certificate. In Figure 6-2 Node C attaches the anchor (R_0) of its hash chain ($R_N \ldots R_0$), while Node B attaches the anchor (Q_0) of its hash chain ($Q_N \ldots Q_0$). Upstream nodes along the path to the initiator note the anchors of each of the chains that will be used for this flow. On receiving the RREP the initiator verifies both the $Sig_{Identity}$ and Sig_{Route} fields to ensure that it has received an authenticated and valid route. Node A is now ready to use the path to transfer data packets to the node D. The general format of a route reply is $RREP(IP_{Dest}, IP_{Src}, (IP_{Hop_1}, IP_{Hop_2} \ldots IP_{Hop_N}), (Flow_{ID}, Sig_{Src}(Sig_{Hop_1}(Sig_{Hop_2} \ldots Sig_{Hop_N}(IP_{Src}, FlowID, Nonce))), Sig_{Dest}(IP_{Hop_N} \ldots IP_{Hop_2}, IP_{Hop_1}, Flow_{ID}, Nonce), Cert_{Dest}, Sig_{Hop_N}(R_0, Flow_{ID}), Cert_{Hop_N} \ldots Sig_{Hop_2}(Q_0, Flow_{ID}), Cert_{Hop_2}, Sig_{Hop_1}(P_0, Flow_{ID}), Cert_{Hop_0}))$.

6.5.3 Secure Datagram Delivery

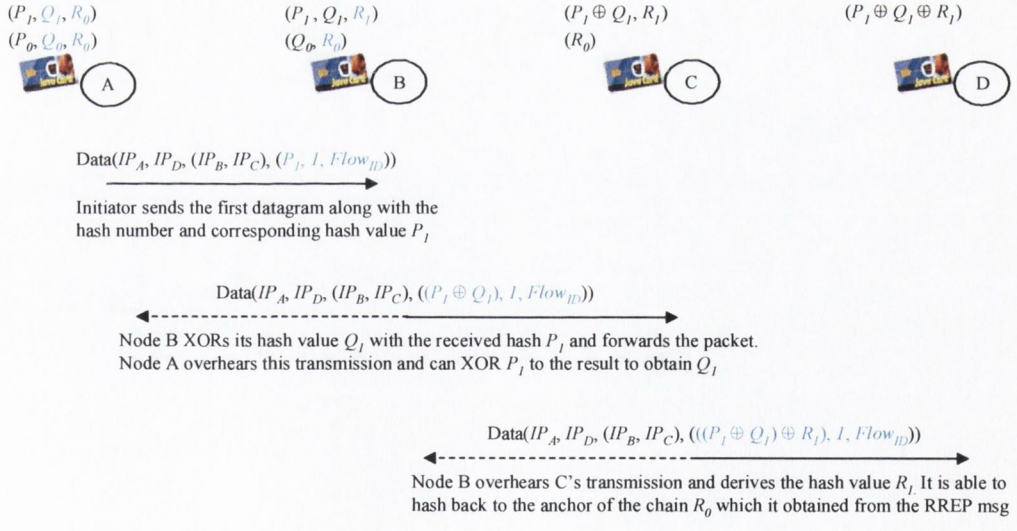


Figure 6-3 Secure Datagram Delivery

Once the RREP has propagated through the network to the initiator, the starting position of each of the nodes in the path is as shown in Figure 6-3. Node C stores the anchor of the hash chain R_0 that it previously generated in its AuthCache. Node B stores the anchor of the chain that it generated Q_0 and R_0 which it

obtained from the RREQ, while node A stores the anchor values P_0, Q_0 and R_0 in its AuthCache. During the data transfer phase, the initiator node A, attaches the next token in its hash chain along with each datagram. Each node also maintains a buffer in which it stores the last transmitted packet. It compares the contents of the buffer with the packet that is forwarded by its neighbor node and ensures that the packet was relayed without any modifications.

For example with the first packet node A attaches the hash value P_1, the corresponding hash number and the $Flow_{ID}$. Node B receives the packet and XORs the first value in its hash chain Q_1 with the received hash value P_1. When node B forwards the packet, node A is able to hear this transmission and XORs P_1 to the value $P_1 \oplus Q_1$ to obtain Q_1, which it stores in its AuthCache. Node C stores the value $P_1 \oplus Q_1$ in its AuthCache and also XORs the first hash value of its chain R_1 to obtain $P_1 \oplus Q_1 \oplus R_1$. As before node B overhears this transmission and applies the value $P_1 \oplus Q_1$ to obtain R_1. Node C relays the packet to node D which stores the value $P_1 \oplus Q_1 \oplus R_1$ in its authentication cache. Under normal circumstances each node XORs the correct hash value from its hash chain to the received packet and updates its AuthCache. The size of the security header remains constant and does not increase with each hop in the network.

6.5.4 Identifying Misbehaving Nodes

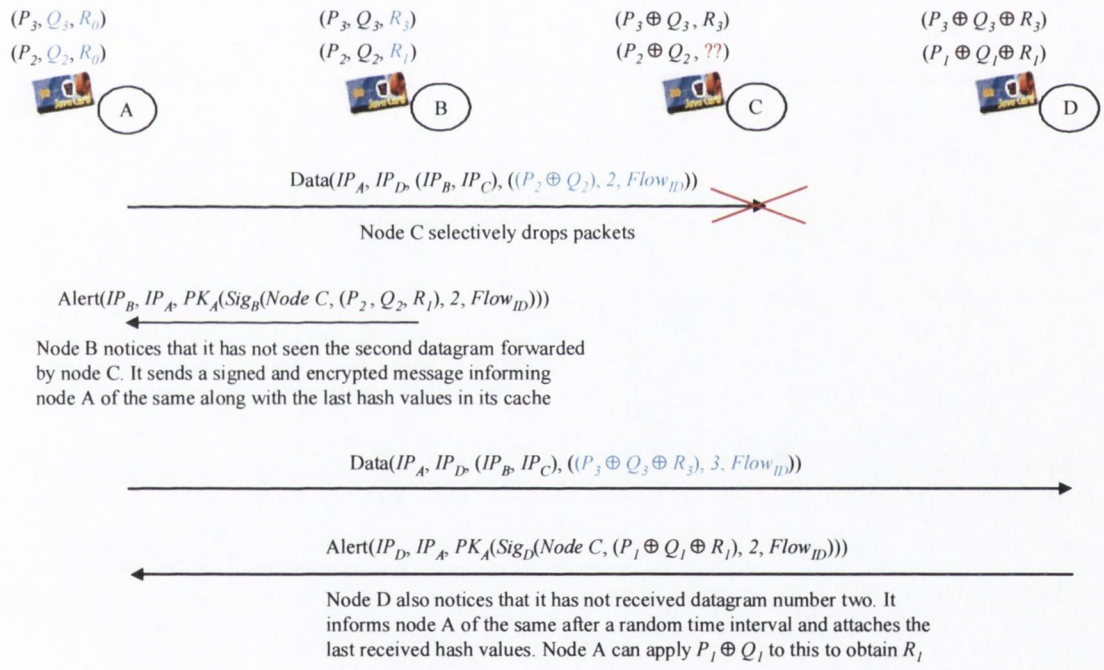


Figure 6-4 Misbehaving Intermediate Node

It is relatively straight forward to identify nodes that launch a DoS attack by refusing to relay *any* packets in the network. However it becomes a much harder task to identify a node if it decides to selectively drop packets. A node may do this to preserve its battery life or because it is wishes to disrupt traffic flows in the network. Due to the inherent lossy nature of a wireless environment, a node can claim that it did not receive a packet due to transmission collisions. The proposed system protects against such misbehavior by listening to the neighbor's transmission and alerting the initiator if a packet is dropped by a node in the path. This does

not immediately result in the node being blacklisted. Instead the received information is fed into a path rating algorithm which helps to determine the rating of a node in the network.

Assuming that nodes *B* and *D* do not collude, Figure 6-4 depicts the situation where node *B* forwards a datagram with the hash value $P_2 \oplus Q_2$ but does not hear its neighbor node *C* relaying the packet onto node *D*. Node *B* sends an *Alert* to node *A* at a random time interval which contains the last hash values in its authentication cache (P_2, Q_2, R_1). The message is signed with the private key of node *B* and further encrypted with the public key of node *A*. The latter prevents node *C* from gaining any knowledge of the contents of the message. Node *B* then forwards the next datagram from the initiator with hash value $P_3 \oplus Q_3$ to node *C*. Node *C* relays the packet to the target node with hash value $P_3 \oplus Q_3 \oplus R_3$. Node *B* is able to obtain the value R_3 from this transmission.

At this point node *D* realizes that it has not received packet number 2 of the flow. Node *D* also sends an *Alert* to the initiator after a random time interval along with the contents of its AuthCache ($P_1 \oplus Q_1 \oplus R_1$). Node *A* is able to XOR the hash value $P_1 \oplus Q_1$ which it received from node *B* to this to obtain R_1. This gives node *A* confidence that node *C* did not relay datagram number 2, either maliciously or due to a collision. Node *A* will only decrease the rating for node *C* if it receives two or more signed *Alerts* from other nodes in the path to the destination. The only way for node *C* to prevent *Alerts* from reaching node *A* would be to drop all packets destined to it. This in effect would be the black hole attack and would immediately disqualify the node. The general format of an alert message is $Alert(IP_{Dest}, IP_{Src}, PK_{Src}(Sig_{Dest}(Node_{ID}, (P_X \oplus Q_X \oplus R_X), X, Flow_{ID})))$, where *X* is the hash number..

6.5.5 PRAT Algorithm

The protocol makes use of the concept of a pathrater agent which is similar to that outlined in [MGLB00]. Each node in the network runs a path rating agent and maintains a rating for nodes that it has gained knowledge of through the route discovery process. It averages the ratings of the nodes in a path and calculates a path metric. If there are multiple paths to a destination the protocol makes use of the path with the highest metric. The agent for example assigns a rating of 0.5 to each new node that it encounters in the network. The agent decrements a node's rating by 0.05 for every *Alert* sent or received up to a minimum value of 0.0. Note that in the case of a source node, it must receive two or more *Alerts* prior to decrementing a node's rating. After which the node is deemed as being malicious and assigned a negative rating of -1.0. A node will not use or advertise a path with one or more nodes with a negative rating or a CRL listing, and will report any misbehaving nodes to its CA when it is in contact with it next. It will only reset the rating of a node to neutral after a long timeout period if it has not been in contact with its CA, or if it receives a new CRL which does not contain an entry for that node.

6.5.6 Discussion

As with any security protocol there is an overhead associated with providing authenticated route discovery and datagram transmission. Assuming that the IP and DSR headers are as in the original protocol, an attempt is made to try and quantify the size of the additional security header in the discussion below. The route request message generated by the initiator grows in size as it travels through the network towards the target node. Assuming that a 1024-bit or 128-byte RSA key is used, and the size of a public-key certificate is ~640 bytes. The size of the accumulated route request $Size_{RREQ} = Sig_{Identity} + N \times (Cert_N)$. For $N = 3$ $Size_{RREQ} = \sim 2$ Kbytes where *N* is the number of hops from the source to the destination. Similarly the size of the route reply $Size_{RREP} = N \times (Cert_N + Sig_{Route}/Signed\ Anchor) + Sig_{Identity}$ for $N = 3$ is ~2.4 Kbytes. The overhead associated with each datagram is ~19 bytes. The above values are however theoretical calculations and would vary slightly depending on the actual implementation.

The proposed solution requires a PKI to be in place but does not necessarily require an online connection to a CA at all times. In the absence of access to a CA, a node may not have the most recent CRL and may accept one or more certificates that may have been revoked. This could result in a node initially using a route which contains nodes that have been blacklisted by the CA, and could result in the dropping of datagrams. However the built-in feedback mechanism allows the nodes in the path to detect and report any misbehavior. This will result in a decrease in the rating for a particular node. If the rating becomes negative then the node will discontinue using any routes in which the misbehaving node is present. The source will eventually report this to the CA and also receive an up to date CRL.

A malicious node could limit its power output such that the signal is strong enough to be overheard by the previous node but too weak to be received by the true recipient [MNP04]. Once the route discovery process is complete, a malicious node could then selectively drop packets by controlling the power output, without the knowledge of its neighboring node. Such an attack could also be possibly mounted using directional antennas. This is a highly sophisticated attack and would require specialized knowledge and hardware to implement. However it is not beyond the realms of possibility for a determined attacker. From an application point of view, there are two types of sessions that can be initiated, namely UDP or TCP. For UDP type connections such as VoIP, the user will expect to hear a response from the correspondent user within a reasonable time period. Similarly for TCP connections, such as FTP or HTTP transfers, the application will require an acknowledgement (ACK) to any outstanding transmitted datagrams. In each case, failure to receive any response or datagrams from the upstream neighbor will result in the manual or automatic dropping of the connection. This is akin to a black hole attack where all the packets are dropped by a node in the network. Thus even though the protocol will fail under such an attack, the application or transport layers will be able to detect the same.

As is the case with some of the other secure routing schemes outlined in Section 6.2, some of the optimization of the DSR protocol have been disabled in order for the protocol to function correctly. One important optimization that has disabled is the ability for an intermediate node to respond to a RREQ message if it has a valid route to the destination cached in its routing table. This can lead to grater delays in obtaining a route to a particular destination.

6.6 Implementation Details

The *ns-2* simulator version 2.26 was used in order to evaluate the performance of the proposed secure ad hoc routing protocol. The base DSR protocol was modified in order to include the secure routing extensions and the OpenSSL libraries (version 0.9.7a) were used to implement the various cryptographic routines. As with the AdPay implementation in Chapter 5, this implementation also employs a single CBR source (node 0) and destination (node 9) pair. This is in contrast to a real network scenario where there may be multiple sources and destinations. This simplification was opted for, as there is a requirement to generate hash chains for each node in the path, and to XOR the correct hash token at each node prior to forwarding a data packet in the network. Multiple sources would have made the prototype development task much more complex, as they would have required flow-state identifiers to be maintained in relevant nodes in the network.

In addition to regular DSR nodes in the network, rogue or malicious DSR nodes were added to the network. These rogue nodes display normal behavior most of the time and forward packets if they are part of the route to a destination. However they also randomly drop packets from time to time to disrupt the normal flow of traffic through the network. Such malicious nodes run a modified version of the base DSR protocol which is referred to as the Rogue DSR (RDSR) protocol in the simulation. Appendix D provides the reader with more details of how to go about changing the base DSR protocol for specified nodes to exhibit malicious behavior.

It should be noted that the SecAd prototype developed in this chapter is a purely proof of concept implementation. A detailed security evaluation will be required prior to deploying the protocol in a live network scenario.

6.7 Experiments and Measurements

Three parallel installations of the ns-2.26 code were used in order to test the protocol. The first was the unmodified DSR code which is referred to as *DSR*. In the second installation, malicious nodes were introduced which ran the RDSR protocol, and this installation is referred to as *DSR+Mal*. This allowed for the simulation of an ad hoc network with malicious nodes which intermittently dropped packets. The final installation contained malicious nodes along with the security enhancements and is referred to as *SecAd* (Secure Ad Hoc). This version enables the detection of the dropping of packets in the network, and to subsequently blacklist nodes which are deliberately misbehaving. As with the previous protocol simulations, the cryptographic processing overhead was added into the scheduler. Adding in the processing delay at each node had an effect on the packet delivery ratio and average end-to-end delay.

6.7.1 Experimental Setup

Once again a 400MHz Pentium II with 256MB of memory running the Linux 2.2.0 kernel was used as the testbed machine, and a number of standard network parameters specific to NS were employed. A standard rectangular space of size 1500m x 300m was used to increase the average number of hops in routes used. The simulation was run with 10 and 30 nodes, with 10 nodes representing a small network and 30 representing a medium sized network. The number of malicious nodes in each case was 50%. Finally, the pause time was varied between 0 and 900 seconds. Zero seconds represents continuous movement, while 900 seconds represents no movement at all. Table 6-1 shows the parameters used in the SecAd simulation.

Number of Nodes	10 or 30
Malicious Nodes (RDSR)	5 or 15
Max Velocity (v_{max})	20 meters/s
Coverage Area	1500m x 300m
Nominal Radio Range	20 meters
Source Data Rate	4 packets/second
Application Data Payload Size	512 bytes/packet
Raw Physical Link Bandwidth	2Mbps

Table 6-1 Parameters for SecAd Simulation

6.7.2 Performance Metrics

The DSR, DSR+Mal and SecAd installations were run on identical movement and communication scenarios. Each combination of node and pause times was run for 10 separate movement scenarios and an averaging of the results was performed. As in Chapter 5, the performance of the protocol was measured along two metrics, namely the Packet Delivery Ratio (PDR) and the Average End-to-End Delay (Latency).

6.7.3 Evaluation of Results

Figure 6-5 shows the packet delivery ratio for the three versions of the protocols. It can be observed that for the small sized network scenario of 10 nodes, the PDR for SecAd is consistently better than the version with malicious nodes (DSR+Mal). As the pause time increases, the PDR for all three versions improves, since the nodes become stationary and a malicious node has a relatively local effect. At 900 seconds (no movement) the PDR for the SecAd protocol is 96% which is as good as the DSR version. The results for the medium

sized network of 30 nodes are not as consistent, with the SecAd protocol faring slightly less better at 0, 30 and 900 seconds.

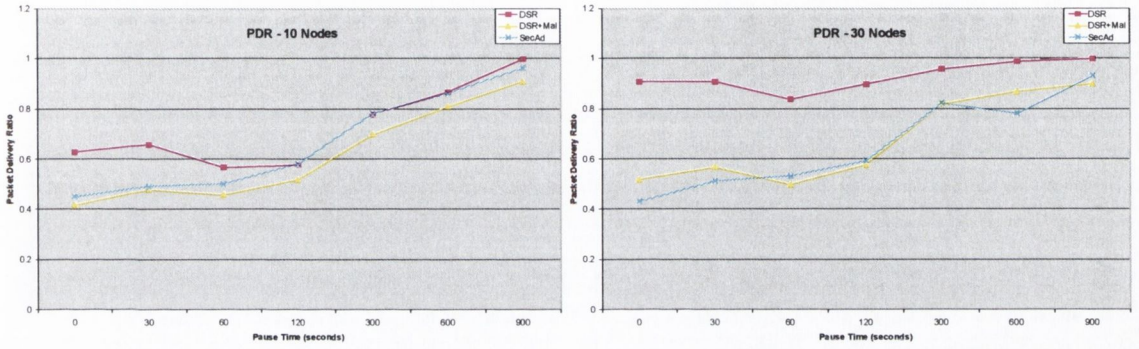


Figure 6-5 Packet Delivery Ratio for DSR, DSR+Mal and SecAd Protocols

Figure 6-6 depicts the average end-to-end delay incurred in the network for 10 and 30 nodes respectively. It can be seen that after incorporating the SecAd scheme, the average end-to-end delay has increased from values in the order of seconds to values in the order of minutes. This is primarily due to the cryptographic overhead incurred in verifying the routes and making sure that packets are forwarded by intermediate nodes in the path to the destination. The first task requires making use of repeated public-key cryptographic procedures in order to generate and verify digital signatures.

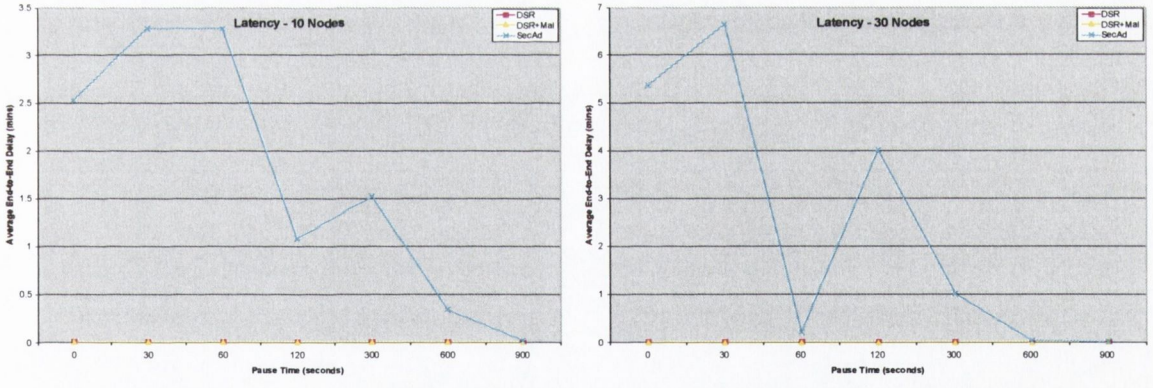


Figure 6-6 Average End-to-End Delay for DSR, DSR+Mal and SecAd Protocols

For the network with 30 nodes it can be seen that the end-to-end delay is quite large, especially during the period when the network mobility is very high. This may be part due to the fact that there are a large number of nodes in the path between the source and destination pairs and that the routes may change frequently. The cryptographic overhead associated with setting up the path may be great, since each node along the path needs to be verified prior to any data packets being exchanged. In some cases, by the time a secure path has been established by SecAd, some nodes may have already changed their position in the network and that route becomes unusable. Another factor that adds to the overall delay is the fact that the SecAd protocol is always required to obtain the RREP message from the target node as route caching mechanism of the DSR protocol has been disabled. As the pause times increases (mobility of the nodes decreases) it can be seen that the delay for the protocol falls rapidly towards zero.

6.8 Summary

A number of open research challenges still remain in the area of securing wireless ad hoc networks. First, the secure routing problem is not well modeled. A more complete model of possible attacks would allow designers to better evaluate the security of their protocols. At the present time many solutions are designed explicitly with certain attack models in mind and may collapse under unanticipated attacks. Second, there are conflicting requirements in terms of designing efficient routing protocols that have both strong security and high network performance, as security never comes for free [YLYL+04]. In many cases the security enhancements remove important performance optimizations.

In this chapter a method to provide for authenticated route discovery in ad hoc networks has been presented. The protocol allows for the initiator of a route request to be assured of the identities of each of the intermediate nodes in the path, as well as the destination node. In addition, the protocol is able to monitor the nodes in the path and determine if they participate in the correct forwarding of datagrams in the network. The protocol employs a path rating agent in each node to keep a rating for all active nodes in the network that a user has knowledge of through the route discovery process. The protocol constantly updates the rating for all such nodes and avoids using nodes that have a low or negative rating. This allows a node to dynamically choose the most reliable paths in the network to a destination. It is envisaged that ad hoc networks will be used in to extend the reach of the fixed infrastructure in next generation networking environments. An important ingredient required to transform ad hoc networks into a mainstream networking technology will be the use of efficient, lightweight secure routing and monitoring schemes, in conjunction with some form of multi-party payment system, to compensate nodes for relaying packets in the network.

The SecAd protocol is a first attempt to try and address the complex issue of routing security in ad hoc networks. The proposed solution differs from the previous approaches to secure routing in ad hoc networks, in that in addition to authenticating each node in the routing path, it also makes use of a path rating agent in conjunction with lightweight cryptographic mechanisms to monitor and rate the nodes in the relay path. A user is able to cryptographically prove to other nodes in the network that a node dropped a packet that it was expected to relay. However, security does not come cheap and the simulation results have shown the considerable overheads incurred in terms of the average end-to-end delay and packet delivery ratio. Some of the optimizations that can be applied to the routing protocol have had to be disabled. The area of secure routing in ad hoc networks remains an open issue, and undoubtedly there will be more efficient and innovative solutions proposed in the future.

7 Cryptographic Performance Analysis

"Where performance is measured, performance improves. Where performance is measured and reported, the rate of improvement accelerates."

Thomas S. Monson

7.1 Introduction

The use of mobile computing devices (e.g. handhelds, palmtops and mobile phones) has increased substantially over the past decade and in particular over the past few years. Personal Digital Assistants (PDAs) which started initially as devices to store personal information have grown more compact with more powerful CPUs. They have evolved to support advanced communications applications that have traditionally been the domain of workstations. At the same time there have been significant changes in the way business is done with the introduction of electronic commerce endeavors through the Internet. Electronic commerce involves the use of strong cryptographic functions and protocols in order to provide adequate security services for payment transactions. These functions can easily be afforded by fixed workstations, but the literature [DB99, GG01] would suggest that on mobile devices such operations are slow and expensive, due to constrained processors, limited memory and battery life. However, the latest generation mobile devices are equipped with much faster CPUs, which facilitate the use of strong cryptographic functions for the construction of security-related protocols.

Throughout this thesis hash functions have been employed to implement the micropayment and security protocols. In order for micropayments to be efficient and scalable, they need to maximize the use of lightweight cryptography. It is generally assumed that public-key cryptography is much more computationally intensive than traditional symmetric cryptography, which in turn is considered less efficient than hash functions. For example, the RSA cipher relies on exponentiation of large numbers which requires many repeated multiplication operations on a standard CPU. The purpose of this chapter is to use experimental measurements to accurately compare the performance of specific cryptographic algorithms on constraint mobile computing devices, such as PDAs. The reader is provided with concrete proof for the above stated assumptions by analyzing the performance characteristics of the various cryptographic algorithms, which in turn gives the reader confidence that the proposed authentication and accounting protocols for next-generation networks will have reasonable performance on mobile devices.

The parameters that are common to all the tests that were performed are presented in the following section. Section 7.3 examines the computational performance of the cryptographic algorithms upon which many security protocols are built. In Section 7.4.1, an evaluation of the performance of the UOBT scheme that was employed in Chapters 4 and 5 is carried out. Similarly in Section 7.4.2, the performance of the Optimal Hash Sequence Traversal scheme that was employed in Chapter 6 is evaluated. Finally, in Section 7.5, conclusions are drawn based upon the performance analysis test carried out in this chapter.

7.2 Methodology

The benchmarking of the OpenSSL cryptographic library was done on an iPAQ H3660 with a 206MHz StrongARM processor and 32MB RAM (16MB ROM), running the Windows CE Pocket PC 2002 operating system [WinCE]. For the implementation of the investigated protocols, a Windows CE port of the OpenSSL cryptographic toolkit version 0.9.7d [OSSL] was employed. While writing the benchmarking programs, no compiler optimizations were used, and the thread priority for the benchmarking programs was normal.

7.3 Performance Comparison of Cryptographic Algorithms

For the performance analysis tests a number of widely used cryptographic algorithms which are considered secure were selected. RSA [RSA78] was chosen as the algorithm for public-key digital signatures, DES [NIST99] and AES [NIST00a] were chosen as the symmetric ciphers, while MD5, SHA-1 and SHA-256 were chosen as the hash functions [NIST05b].

For RSA, a key size of 1024 bits was employed, as anything smaller is considered to be too short for sensitive data. The benchmarking programs were performed on the commonly used public-key operations, specifically the signing and verification of data. For each one of the iterations of the benchmarking program, initially a SHA-1 hash object was created and given 64 bytes of data to hash. This hash was signed with RSA key size of 1024 bits and 100,000 iterations were performed. For RSA verification, the same process was repeated using the same number of iterations. For symmetric encryption the AES cipher was used. Again 100,000 iterations were done for each AES encryption cycle with a 256-bit key. The hashing algorithms that were compared were MD5, SHA-1 and SHA-256 for 100,000 iterations, with 64 bytes of input data.

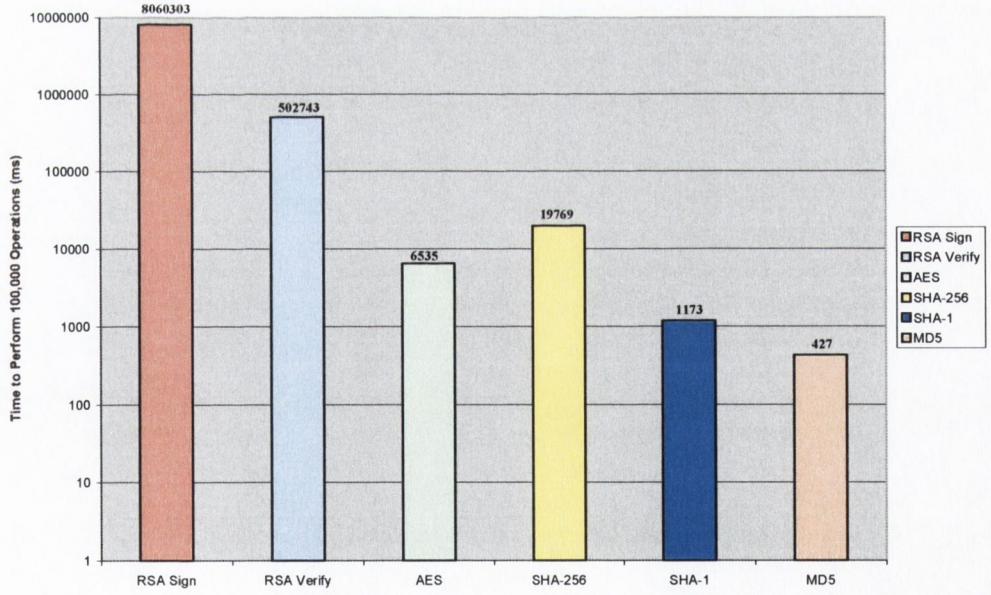


Figure 7- 1 Timing Measurements of Low-level Cryptographic Primitives on an iPAQ H3660

Figure 7-1 shows the time taken in milliseconds to perform 100,000 operations for each of the algorithms tested. A more intuitive chart is presented in Figure 7-2, which shows the number of operations per second each algorithm is capable of performing. It can be seen that one can perform over 234,000 MD5 hashes,

85,200 SHA-1 hashes, 199 RSA signature verifications, and 12 RSA signature generations per second. A logarithmic scale is required to illustrate such huge differences. One can observe the differences in hash function speeds where MD5 is nearly three times as fast as SHA-1. However SHA-1 is considered the more secure than MD5 as the length of the message digest output is 160 bits. Also in 2004 collisions were found in the MD5 compression function. Overall, hashing with MD5 can be up to an order of magnitude faster than symmetric encryption, three orders of magnitude faster than signature verification, and four orders of magnitude faster than signature generation. With SHA-1, each of the above values is reduced by an order of magnitude.

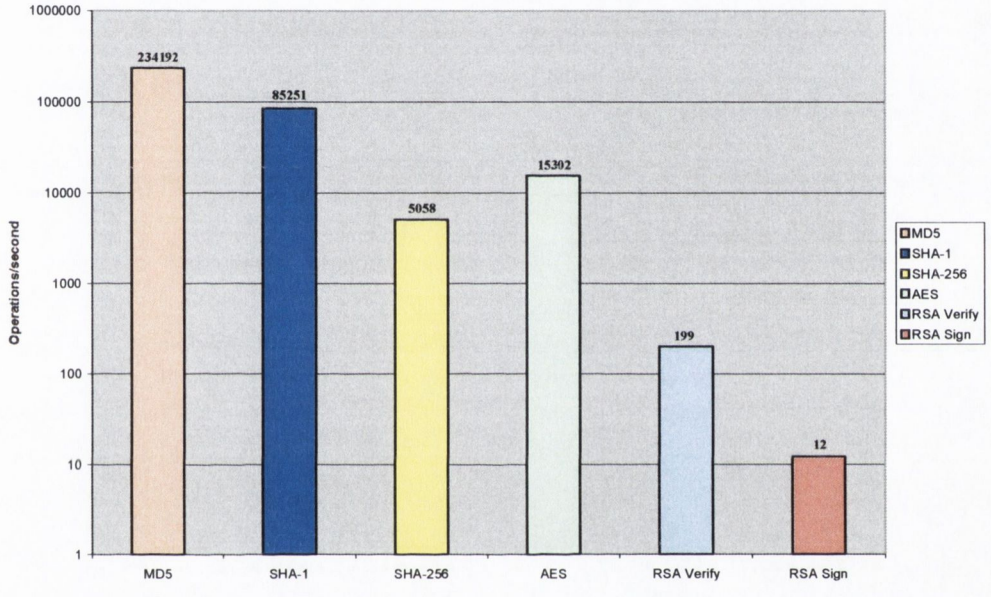


Figure 7- 2 Efficiency Comparisons of Cryptographic Functions

In February 2005 a collision attack was reported on SHA-1 [WYY05, NIST05a]. With this in mind a version of the SHA-256 code was ported to the WinCE platform. SHA-256 is comparably slower than any of the other hashing algorithms that have previously been benchmarked. One reason for this is that the output of the message digest function is now 256 bits (32 bytes). A more suitable comparison would with the AES algorithm with a key size of 256 bits and a block size of 128 bits. Also, the version of the SHA-256 code that was used may not be as highly optimized as are versions of other algorithms that are included as part of the OpenSSL cryptographic library. However, the SHA-1 attack is a specific collision attack and should not affect key derivation and hash chain generation functions.

Work has also been done in benchmarking the AES algorithm and a number of hash functions on desktop Pentium class machines in [Lip05, Knu05, AESL05, Bar05]. Specifically in [Bos05], the author ran a number of tests on a 90 MHz Pentium machine using fast implementations of the MD5 and SHA-1 algorithms. For MD5 the number of bits hashed per second was in the region of 191Mbps., while a figure of 119Mbps (234192 * 512) was obtained for the benchmarks performed on a PDA in this chapter. Similarly for SHA-1, the number of bits hashed was approximately 55Mbps, while the figure obtained on a PDA was 43Mbps (85251*512). In general the results obtained on a constrained mobile device such as a PDA running the WinCE operating system were comparable to that obtained on a fixed desktop machine.

7.4 Efficient Hash Token Generation Schemes

As outlined in Chapter 3, two bulk hash generation schemes have been employed in this thesis, namely the Unbalanced One-way Binary Tree (UOBT) and the Optimal Hash Traversal schemes. The UOBT [YHH99] is an efficient hash chain scheme, where the root of each chain is derived from another hash chain. If a UOBT is used with a backbone chain length equal to the length of each sub-chain, it can be shown that the average computational overhead to compute the next hash is $n^{1/2} - 1$, where n is the number of values in the UOBT, and $n^{1/2}$ is the square root of n. This gives an efficiency improvement from $O(n)$ to $O(n^{1/2})$.

In [Jak02], the author presents an optimal hash sequence traversal scheme which has an upper bound of memory and computational complexities of $O(\log n)$. The user pre-computes all the values in the hash chain starting from the root prior to using them on a powerful machine such as a desktop computer. However, he is only required to store a number of strategic hash values or *pebbles* in the chain for later use on the mobile device. A pebble stores the hash chain value for a position it is associated with, and the location of these pebbles is modified over time.

7.4.1 UOBT Performance Evaluation

For the UOBT scheme a 100x100 tree consisting of 10,000 hash tokens was generated. The total time taken to generate 100,000 binary trees was 21785052ms. This equates to approximately 0.22 seconds for each UOBT generation. According to [YHH99], a 100x100 UOBT requires 99 hashes on average to compute a required hash value in the tree, which equates a value of 0.002 seconds ($0.22*99/10^4$).

7.4.2 Optimal Hash Sequence Traversal Performance Evaluation

For the Jakobsson scheme [Jak02], hash chains of length 1024 (10^{10}) and 2048 (10^{11}) were generated, and the resulting pebbles were stored. Assuming that the initial chain generation is performed in an offline manner on a powerful desktop machine, only the subsequent traversal through the chains was measured. In each case the experiment was repeated 100,000 times and the MD5 algorithm was used. Appendix E provides an implementation of the Jakobsson algorithm which has been adapted for the WinCE Pocket PC 2002 environment. For a chain with 1024 tokens that has storage requirements for 10 pebbles, the total storage requirements are 10 x ($2\log_2 n + 128$) or 185 bytes. The total time taken was 10607718ms, which equates to 0.11seconds for each iteration, or 0.0001 seconds on average to generate a single hash token in the chain. Similarly for a chain with 2048 tokens that has a storage requirement of 11 pebbles (~206 bytes). The total time taken was 14762741ms, which equates to approximately 0.15 seconds per iteration, or 0.0001 seconds on average to generate the next hash token in the chain.

Hence it can be seen that the optimal hash traversal scheme is more efficient in terms of memory and computational requirements than the UOBT scheme. It is particularly well suited for the MobPay protocol, where a hash chain of length 2^{32} will last for more than 68 years if a single hash value is released per second. The memory requirements are reduced to two hash tokens in each of the routers in the access network. Also the mobile node only needs to send a single hash value to the broker to obtain a commitment.

7.5 Summary

The chapter demonstrates the feasibility of using strong cryptographic functions on constraint mobile devices for building security protocols. The comparison between the work carried out in this thesis and previous related work revealed interesting results regarding the advances of constrained devices [DB99]. The order of time required for RSA key generation has been reduced from minutes to seconds, and message digesting with the SHA-1 algorithm has become faster by three orders of magnitude.

Also the bulk hash generation schemes that were employed performed well in the tests. In addition, both schemes make use of efficient storage methods which do not require the entire tree or chain to be stored on the mobile device, making their use much more practicable. The benchmarks show that currently available constraint mobile computational devices which will form the bulk of nodes in next-generation wireless networks can perform strong cryptographic functions.

From the experiments obtained it can be seen that hash chains provide the best overall computational performance. They are best suited to scenarios with computationally constrained devices, which have limited storage and bandwidth and vendors who have to process a large number of payments per second. In addition to performance, their security properties are stronger than those of the next best performer i.e. symmetric key schemes, where non-repudiation cannot be provided due to the shared nature of the keys. It can be concluded that the use of fast, lightweight hash algorithms will have a minimal impact on existing end-user applications, such as voice telephony and Web browsing in next-generation mobile networks.

8 Conclusions

"Now this is not the end. It is not even the beginning of the end.
But it is, perhaps, the end of the beginning."

Winston Churchill (1874 – 1965)

8.1 Summary of Contributions

The main goal of this thesis was to evaluate the use of real-time electronic payment techniques using lightweight cryptographic methods, as a replacement for traditional billing in mobile communications environments. A micropayment protocol which allows network operators to be paid in real time for service provision, and simultaneously provides for authentication of routing updates within their networks, was implemented. The proposed solution addresses the twin problems of authentication and accounting that next-generation mobile network operators and end-users will face, in an environment of ubiquitous mobile communications.

A review of 2G and 3G network technologies highlighted the inadequacies of the currently employed authentication and accounting protocols. With large numbers of network operators, service providers and millions of end-users, these problems are set to become ever greater in next-generation mobile networks. Current billing techniques require total trust to be placed in the NO to generate accurate usage records for network usage. This is an unacceptable requirement in a ubiquitous communications environment where there may be large numbers of independent operators and service providers. These problems provided the motivation for this research. The resulting micropayment and secure routing schemes were designed as efficient, flexible and secure solutions for next-generation mobile networks.

A major contribution of this dissertation is the design of a real-time authentication and accounting protocol for next-generation mobile networks. The scheme allows for prepaid, identified hash chains to be used to pay for accessing network services in real time by roaming mobile users. This eliminates the need for end-users to trust the NOs and VASPs to produce accurate usage records, which in turn eliminates complex accounting and billing procedures. As long as the service is provided, the user will release payment tokens into the network. The same applies in the reverse direction, whereby the NO or VASP will stop providing the service if valid payment tokens are not received. The payment tokens are redeemable only by the specified entity which prevents them from being double-spent by cheating nodes. There is no requirement to maintain an online connection to a TTP to verify the validity of payment tokens by an accepting entity. The requirement for a home network has also been eliminated by removing the need for subscription based billing and location management. The requirement that end-users have to pay a location management server to obtain the current location of roaming mobiles minimizes the number of unsolicited messages being sent to a node. In addition, the protocol allows signaling messages transmitted by a MN to be quickly and efficiently authenticated, by employing lightweight cryptographic techniques.

The MobPay scheme was implemented using the *ns-2* Cellular IP extensions and made use of the OpenSSL libraries to implement the cryptographic protocols. Performance measurements showed that the average overhead for both UDP (e.g. voice telephony) and TCP (e.g. Web browsing) is in the order of microseconds. Thus the protocol has a minimal impact on the end-user applications, and at the same time provides secure payment for network services and authentication of routing updates of roaming mobiles. To summarize, the authentication and accounting protocol for next-generation mobile networks eliminates any long-lived trust relationships from the system, reduces the online communications overhead of contacting a distant home network operator, and allows real-time payment for usage of network resources anywhere by anyone who holds valid payment tokens.

One important aspect of next-generation networks will be the presence of mobile ad hoc networks. The MobPay protocol was extended into a multi-party micropayment protocol to provide for remuneration of nodes for packet relaying in an ad hoc network scenario. The resulting AdPay scheme allows each of the nodes in the path between the target and destination nodes to be paid in real time for packet forwarding without the requirement of contacting a TTP for payment verification. In addition, the solution allows changes to occur in the relay path in the ad hoc network, again without the need to contact a TTP to construct a new payment contract or to buy new node specific payment tokens. The source node can use unused chains from the UOBT and can immediately start paying the new nodes in the path towards the destination node. Also, unlike the previous schemes, no restrictions are imposed in terms of any fixed or special administrative nodes being part of the routing path. The simulation results showed that the AdPay multi-party micropayment protocol imposed a minimal overhead over the base DSR protocol.

Current ad hoc routing protocols implicitly trust all the participants to cooperate in the relaying of datagrams in the network, and are not able to cope with network disruptions due to malicious behavior of one or more of the participants. To counteract such attacks, a lightweight cryptographic secure routing and monitoring protocol was developed. The SecAd protocol allows a node wishing to find a route to a destination to authenticate the identity of each node in the chosen path. Each node also monitors its neighbor's transmissions, maintains a rating for all other nodes that it encounters in the network through the route discovery process, and makes use of the highest rated path to the destination even if a shorter path exists. This results in a higher percentage of datagrams being delivered to their destination in the face of adversaries in the network.

Taking into account the protocols developed in Chapters 4, 5 and 6 and the above discussion, it becomes apparent that there is a crucial requirement for authentication and accounting procedures in next-generation mobile networks. As mentioned throughout this thesis, such networks will consist of networking technologies primarily based around infrastructure and ad hoc networks. Since these are radically different networking paradigms, the authentication and accounting procedure required in each will be substantially different, as can be noted from the MobPay and SecAd protocol descriptions. Thus mobile nodes in next-generation networks will be required to intelligently select the appropriate payment and authentication protocols for the type of network they find themselves in at any given time.

To be certain that the proposed cryptographic extensions could be executed quickly and efficiently on constrained mobile devices such as PDAs, a number of benchmarking tests were carried out on a WinCE Pocket PC 2002 device. The performance analysis proved that the current generation of hand held mobile devices are able to efficiently execute basic cryptographic algorithms such as creating and verifying digital signatures. The analysis also showed that the bulk hash chain generation techniques provide the best overall memory and computational performance.

In summary, a review of authentication and accounting techniques in current mobile networks exposed a number of open problems, which have been addressed by developing a real-time micropayment protocol. The resulting scheme is an improvement over traditional billing as it removes any long-lived trust relationships between the various entities in the network, and hence the need for complex roaming agreements. A mobile user can pay any network operator in any location without the ability to overspend. Real-time payment without the need for long-lived contracts eliminates fraud, and in turn allows independent network operators and service providers to flourish, by providing faster and cheaper network access to end-users. In turn, mobile nodes will be able to pick and choose access networks and service providers wherever they roam. In situations where a mobile node is not directly within the communications range of a fixed node such as base station or access point, it will be able to route its packet securely through other mobile nodes and compensate them for their services in real time. In conclusion, it is envisaged that in the future there will be a large number of micro- and pico-cellular networks based on IP, in tandem with islands of ad hoc nodes, which will provide the next generation of telecommunications services to a very large user population. These networks will require secure and scalable AAA provisioning and secure routing strategies. The solutions proposed in this thesis provide an efficient means of authentication of signaling messages and accounting of resources in next-generation mobile networks.

8.2 Directions for Future Research

A number of avenues are possible to further this research. This thesis represents a first step in at providing authentication and accounting in next-generation mobile networks. It is likely that there will be further advances in areas of Quality of Service (QoS) and mobility management, such as the development of new micromobility architectures to support fast handover in the access network. The whole area of adaptive terminals and base stations using software radio techniques will also have to be explored further [MRPS04]. The recent SHA-1 collision attack has highlighted the requirement for further work to be carried out in the development of new and efficient, collision resistant hash algorithms.

Mobile devices in next-generation networks will have to evolve into highly sophisticated and intelligent nodes in order for end-users to gain optimal performance across a variety of wireless environments. With large numbers of heterogeneous access network technologies in use, nodes will be required to employ intelligent mobility management schemes [AXM04]. Users will have to rely on their mobile device to intelligently pick different access networks and location management servers based on a combination of mobility, call history profile, tariffs and QoS requirements. Sophisticated algorithms, based on artificial intelligence techniques, could be employed to address this important issue.

In the micropayment protocols developed in Chapters 4 and 5, not a great deal of attention was paid to the actual pricing contracts and how tariffs were set. Ideally, prices should remain fixed for the duration of a typical call, although this may be subject to change if the volume of data transferred or received exceeds the agreed pricing contract. Suitable economic theory could be applied here to develop adaptive pricing algorithms to allow dynamic tariffs to be set. Also, it would be prudent to perform a formal mathematical proof of the protocols developed in Chapters 4, 5, and 6, prior to deployment.

The performance analysis carried out in Chapter 7 was only done on a number of cryptographic primitives. It would be prudent to implement the MobPay, AdPay and SecAd protocols in a real testbed and evaluate their performance with a large population of mobile nodes. Most of the effort would be required in porting the adapted CIP and DSR protocols to the WinCE Pocket PC 2002 environment. Alternatively one could use a version of Linux on the PDAs [Handhelds] and then port the protocols. The latter would be the preferred approach as the CIP and DSR code base was originally developed on the UNIX platform.

Finally, in this thesis an initial attempt at developing a multi-party payment scheme and a secure routing protocol for ad hoc networks has been made. Further work could be carried out in integrating the two protocols more tightly. The area of ad hoc networks is rapidly expanding with new routing protocols and applications being proposed on a regular basis. Undoubtedly, other novel methods for effecting payment and securing routes in ad hoc networks will be developed in the future.

Appendix A Cryptographic Terms and Notation

This appendix provides the reader with an overview of the cryptographic terms, operators and the notation used to illustrate the various protocol exchanges in this thesis. More detailed descriptions of the various cryptographic algorithms and techniques can be found in [Sch96, MOV96, RSALab, OPT01].

Most cryptosystems can be classified under one of the following two categories, namely *secret-key* and *public-key* cryptosystems. In secret-key cryptography, also referred to as symmetric-key cryptography, the same key is used for both encryption and decryption. On of the most popular secret-key cryptosystems is the Data Encryption Standard (DES) [NIST99]. However NIST now recommends the use of the Advanced Encryption Standard (AES) [NIS00] which is the successor to DES and is considered to be more secure. In public-key cryptography, each user has a *public key* and a *private key*. The public key is made widely available while the private key remains secret. The RSA public-key cryptosystem is one of the most popular forms of public-key cryptography, where RSA stands for Rivest, Shamir and Adleman, the inventors of the RSA cryptosystem.

The Digital Signature Algorithm (DSA) [NIST00b] is another popular public-key technique, though it can only be used for generating digital signatures, not encryption. Hash functions such as MD5 [Riv92a] and SHA-1 [NIST05b] can be used for producing Message Authentication Codes (MACs), and in conjunction with the RSA cryptosystems for generating digital signatures. This appendix begins by defining a number of cryptographic terms used within this thesis. This is followed with more details on the cryptographic techniques, operators, and notation used throughout this thesis.

A.1 Cryptographic Definitions

A number of cryptographic definitions are provided below:

A.1.1 Authentication

Authentication is a term which is used in a very broad sense. By itself it has little meaning other than to convey the idea that some means has been provided to guarantee that entities are who they claim to be, or that information has not been manipulated by unauthorized parties. A *digital signature* is a cryptographic means through which the above may be verified. Digital signature techniques are elaborated on in detail in the following sections.

A.1.2 Identification

Identification is the process through which one ascertains the identity of another person or entity. Identification requires that the verifier check the information presented against all the entities it knows about., while authentication requires that the information be checked for a single, previously identified entity. In addition, while identification must, by definition, uniquely identify a given entity, authentication does not necessarily require uniqueness.

A.1.3 Non-repudiation

In a cryptographic context, the word *repudiation* refers to the act of denying association with a message (i.e. claiming it was sent by a third party).

A.1.4 Security Association

A Security Association (SA) is a relationship between two or more entities that describes how the entities will utilize security services to communicate securely. This relationship is represented by a set of information

that can be considered a contract between the entities. The information must be agreed upon and shared between all the entities. For example the attributes specified for an IP Security (IPsec) SA include, the authentication mechanism, cryptographic algorithm, algorithm mode, key length, and Initialization Vector (IV) [MSST98].

A.2 Secret-Key Cryptography

Secret-key cryptography is also referred to as symmetric cryptography where a single shared key is used to encrypt and decrypt a message. Secret-key cryptosystems can be used for encryption as well as authentication. The main problem with secret-key cryptosystems is key management, i.e. getting the sender and receiver to agree on the secret key without anyone else finding out. However, secret-key cryptography is generally faster than public-key cryptography. The most common techniques used in secret-key cryptography are block ciphers, stream ciphers and message authentication codes.

A.2.1 Block Ciphers

Iterated block ciphers encrypt a plaintext block (user data) by a process that has several rounds into ciphertext (encrypted text). A Feistel cipher is a special class of iterated block cipher where the ciphertext is calculated from the plaintext by repeated application of the same transformation function. Feistel ciphers are also sometimes called *DES-like* ciphers. Figure A-1 shows the general structure of a Feistel cipher.

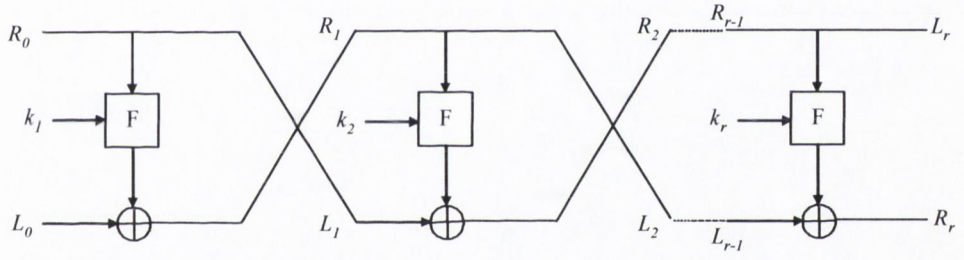


Figure A-1 Feistel Cipher

In a Feistel cipher, the plaintext being encrypted is split into two halves. The round function *f* is applied to the right half using a subkey and the output *f* is XORed with the left half. The set of subkeys is usually derived from a user-provided secret key by a special function. The right and left halves are then swapped. Each round follows the same pattern except the last round where there is no swap. An important feature of a Feistel cipher is that encryption and decryption are structurally identical. However the subkeys used during encryption at each round are used in the reverse order during the decryption operation.

A.2.2 Stream Ciphers

A stream cipher is a type of symmetric encryption and is usually faster than a block cipher. While block ciphers operate on large blocks of data, stream ciphers operate on bits of data. A stream cipher generates a keystream which is a sequence of bits used as a key. Encryption is accomplished by combining the keystream with the plaintext, usually with a biwise XOR operation. One of the most widely used stream ciphers is the RC4 cipher [Riv92b]. RC stands for Ron's Code, but officially it is an abbreviation of "Rivest Cipher".

A.2.3 Message Authentication Codes

A Message Authentication Code (MAC) is a key-dependent one-way hash function. Hash functions are discussed in detail in Section A.3 of this Appendix. Unlike digital signatures, MACs are computed and

verified using the same key. Message authentication codes can be created in a number of ways. One of the most popular methods is to use a shared symmetric key in conjunction with a hash function, also known as Hash function based MACs (HMACs). Another popular method is to make use of a block cipher. A symmetric cipher such as DES can be used to encrypt the message blocks in Chain Block Cipher (CBC) mode. The final block in the ciphertext is used as the checksum or MAC. Other methods using One-Time Pads and stream ciphers are also possible [RSALab].

A.3 Hash Functions

A hash function H is a transformation that takes an input m and returns a fixed-sized string which is called the hash value h where $h = H(m)$. A hash function H is said to be one-way if it is hard to invert, i.e. it is computationally hard to find some input x such that $H(x) = h$. The resulting digest is a strong *digital fingerprint* of the message. A *strongly collision- free* hash function H is one for which it is computationally infeasible to find any two messages x and y such that $H(x) = H(y)$. Finally, even a small change in the input to the hash function should result in a significant change in the output, a phenomenon known as the *avalanche effect*. Examples of some well known hash functions are MD5 [Riv92a], SHA-1 and SHA-256 [NIST05b].

In this thesis a hash operation on a number of parameters is represented by the following notation:

$$h = H(X \| Y \| Z)$$

where the quantities X, Y and Z are concatenated together prior to being passed to the hash function. Concatenation is depicted using the || operator.

The main role of hash functions is in the provision of message authentication codes. Hash functions can also be used in conjunction with the RSA algorithm to produce digital signatures. More details about digital signatures can be found below.

A.3.1 Hash Chains

Lamport [Lam81], proposed the repeated evaluation of a one-way hash function to generate a chain of values allowing many user authentications. A *hash chain* of length n is constructed by applying a hash function n times to a random value labeled x_n. The value x_n is called the *root value* of the hash chain. A hash chain can be derived using a hash function H recursively as:

$$H^n(y) = H(H^{n-1}(y))$$
$$H^0(y) = x_n$$

where $H^n(y)$ is the result of applying a hash function repeatedly n times to an original value y. The final hash value, or *anchor*, of the hash chain after applying the hash function n times is $x_0 = H^n(x_n)$. The hashes are numbered in increasing order from the chain anchor x_0, such that $H(x_1) = x_0$, and $H(x_2) = x_1$.

A.3.2 Unbalanced One-way Binary Tree

A number of hash chain schemes have been proposed that modify the chain structure to produce a tree or graph structure, and allow more efficient storage and computation of large numbers of hash values. One such scheme of particular interest is the Unbalanced One-way Binary Tree (UOBT) scheme [YHH99]. With an Unbalanced One-way Binary Tree (UOBT) each sub-chain forms a series of right branches in the tree, while the backbone chain values continue forming left branches. In Figure A-2 the logical right branches are depicted as vertical (solid) lines while the logical left branches form horizontal (dashed) lines. Since each value in a UOBT has at most two children nodes, and since each node in a sub-chain has only one child, the

tree structure forms an unbalanced binary tree. It is one-way due to the use of one-way hash functions which allow branches to be traversed in one direction only, given a certain node value.

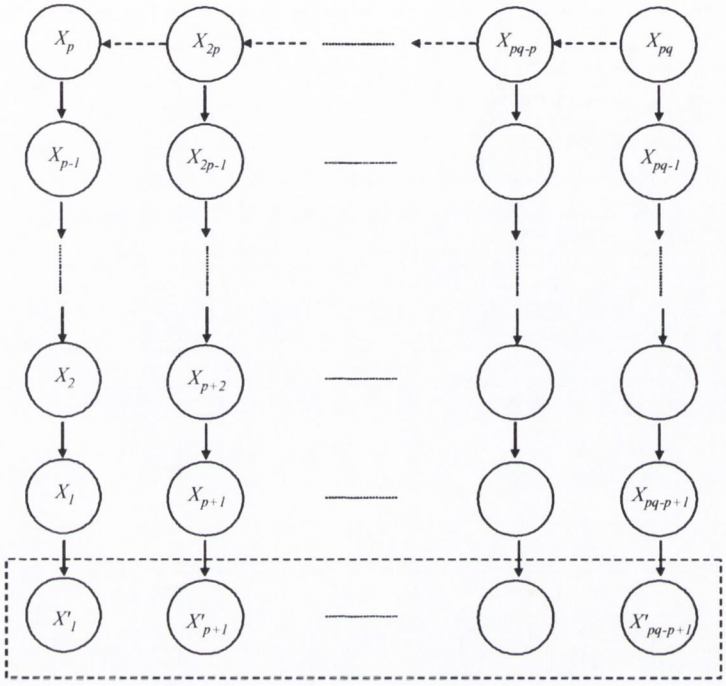


Figure A-2 Generic UOBT

However, to construct a one-way binary tree two different one-way hash functions are required. If the same hash function is used for the left directed (dashed arrows) and right directed (solid arrows) subtrees, for example $X_p = h(X_{2p})$ and $X_{2p-1} = h(X_{2p})$. Under this construction, given Xp, the node X_{2p-1} and all of its children can be obtained. Instead, during the UOBT construction, $h_1()$ (e.g. SHA-1) is used to generate the left directed subtree and $h_2()$ (e.g. MD5) is used for the right directed subtree. The following formula can then be used to compute the value of any node X_i when given the tree root X_{pq} and the index i:

$$X_i = h_2^b(h_1^a(X_{pq}))$$

where $a = q - \lceil i/p \rceil$

$b = (p - (i \bmod p) \bmod p)$

If a UOBT with a backbone chain length equal to the length of each sub-chain is used, it can be shown that the average computational overhead to compute the next hash is $n^{1/2} - 1$, where n is the number of values in the UOBT and $n^{1/2}$ is the square root of n. This gives an efficiency improvement from $O(n)$ to $O(n^{1/2})$.

A.4 Public-Key Cryptography

Diffie and Hellman [DH76] proposed the concept of public-key cryptography in order to address the key management problem in secret-key cryptosystems. In a public-key cryptosystem, each user generates or securely obtains a pair of keys. One is called the public key and is widely published. The other is called the private key and is kept secret in a device such as a smart card. The need for the sender and receiver to share secret information is eliminated as all communications can be performed using the public keys. Private keys

are not required to be transmitted or shared in the system. The keys are mathematically linked such that encryption with the public key can only be reversed with the corresponding private key.

Figure A-3 depicts the situation where the sender (Alice) obtains the recipient's (Bob) public key (PK_B) to send him an encrypted message. The message can only be decrypted with Bob's private key (SK_B) which is only known to Bob. The only requirement is that public keys be associated with their users in a trusted manner. The concept of a Public Key Infrastructure (PKI) is discussed in more detail below.

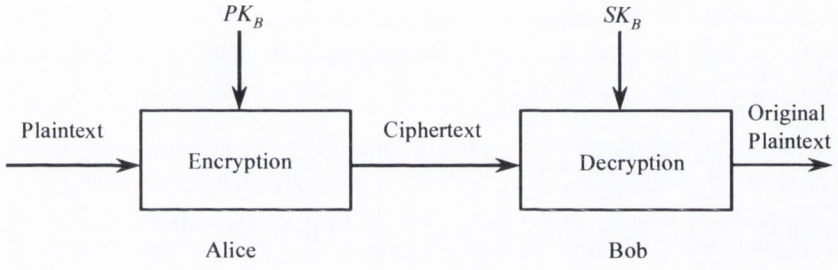


Figure A-3 Public-key Encryption

There are a number of well known public-key cryptosystems in use today each with varying capabilities. For example the RSA algorithm can be used for encryption, decryption, generating digital signatures and key-exchange purposes. The Digital Signature Standard (DSS) which incorporates the Digital Signature Algorithm (DSA) [NSIT00b] on the other hand can only be used for creating digital signatures, while the Diffie-Hellman [DH76] public-key algorithm can only be used for authenticated key-exchange purposes. In this thesis we use the following notation to represent encryption with the public key:

$$PK_B(Pram1, Pram2, Pram3)$$

where PK_B is the public key of the recipient (Bob) and *Pram1*, *Pram2* and *Pram3* are the quantities to be encrypted.

A.4.1 Digital Signatures

A digital signature is an electronic rather than a written signature that can be used by someone to authenticate the identity of the sender of a message or of the signer of a document. It can also be used to ensure that the original content of the message or document that has been conveyed is unchanged. A digital signature must also be unforgeable and must satisfy the property of non-repudiation. There are a number of public-key algorithms that can be used for generating digital signatures. The basic signature generation protocol is as follows:

1. Alice encrypts the document or message with her private key, thereby signing the document
2. Alice then sends the signed document to Bob
3. Bob decrypts the document with Alice's public key, thereby verifying the signature

In practical implementations using the RSA algorithm, in order to reduce the computation time, one-way hash functions are employed. Instead of signing an arbitrarily large document *M*, Alice passes the document through the hash function *H* to produce a fixed length quantity and encrypts this with her private key. Also the original message can be encrypted with the public key of the recipient (Bob) to create a digital signed envelope. Figure A-4 shows the overall process of generating a digital signature using the RSA algorithm.

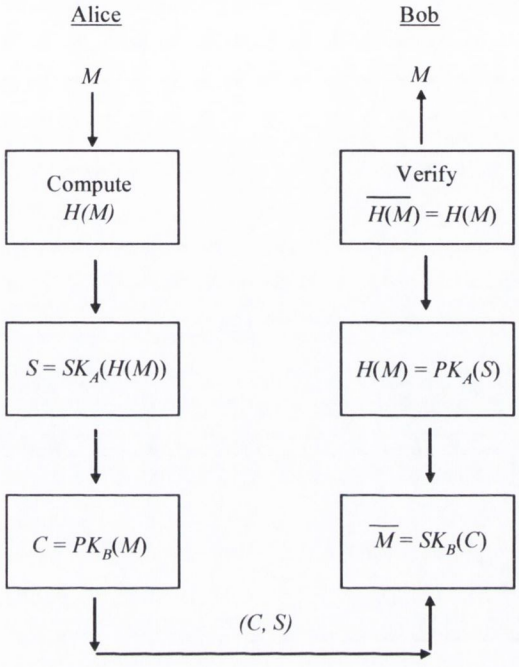


Figure A-4 Enveloped and Signed Data

In this thesis the following notation is used to denote a digital signature generated using the RSA algorithm:

$$Sig_X(M, N, O, P) = (M, N, O, P, SK_x(H(M \| N \| O \| P))$$

where the hash on the concatenation of the parameters is encrypted with the senders private key. The original message parameters are then sent along with the appended hash as the digital signature to the recipient. The digital signature can be optionally encrypted with the public key of the recipient.

A.4.2 Public Key Infrastructure (PKI)

Public-key cryptography is based on the idea that an individual will generate a key pair, keep one component secret, and publish the other component. Other users on the network must be able to retrieve this public key, associate it with an identity of some sort and use it to communicate securely with, or authenticate messages from the user claiming that identity. If an attacker can convince a user that a bogus public key is associated with a valid identity, then the attacker can easily masquerade as the person with that identity. The simplicity of this attack demonstrates that public-key cryptography can only work when users can associate a public key with an identity in a trusted fashion.

A.4.3 X.509 Certificates

One way to form a trusted association between a key and an identity is to enlist the services of a Trusted Third Party (TTP). This is an individual or organization which all users of a system can trust. In an identification scheme, it could be a government organization, or in a payment system, it is likely to be a financial institution. This TTP will construct a message referred to as a *Certificate* which contains a number of fields, the most important of which are a user identity and the associated public key. The TTP signs this certificate using its private key, in the process guaranteeing that the public key is associated with the named

user. This guarantee is made subject to a defined security policy. This could be quite lax, and involve the user forwarding the public key to the TTP for certification, or it could be an involved process requiring the physical presence of the user together with the presentation of multiple forms of identification.

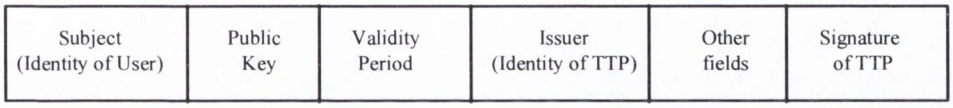


Figure A-5 X.509 Digital Certificate

The certificate is used when a message recipient wishes to gain access to the sender's public key. He can either consult some on-line directory service to obtain this or alternatively, the sender may append their certificate to the message. It is assumed that every user in the system is first equipped with the public key of the TTP. Using this, the signature on the certificate can be verified, and if it passes the test, the public key contained in the certificate can be trusted.

A.4.4 Certification Hierarchy

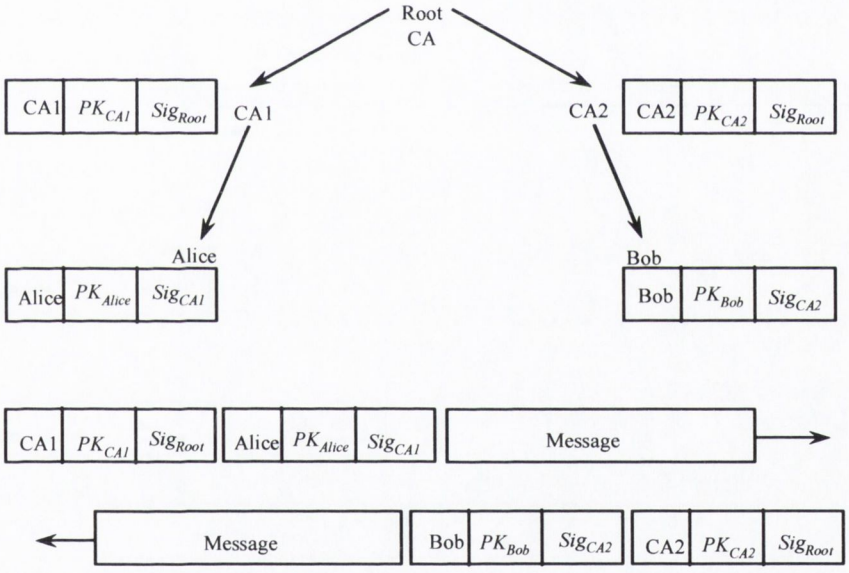


Figure A-6 Certification Hierarchy

TTPs that issue certificates are referred to as Certification Authorities (CAs), and when the population of users becomes large, it is unlikely that a single CA can serve the entire user base. This means that either each user must acquire the public keys of each independent CA, or the CAs can organize into a hierarchy. The root of the hierarchy is a CA that issues certificates only to other CAs, which then certify users of the system. There may of course be more levels than this but the principles are the same. Each user of the system need only hold the public key of the root CA, and when sending a message they include a copy of all certificates in the path between them and the root.

Figure A-6 shows a simple certification hierarchy where Alice has been certified by CA1 and Bob by CA2. The two CA's use a common root CA which has issued certificates for both CA1 and CA2, and all users of the system are equipped with the public key of this root CA. When Alice sends a message to Bob, she

includes her own certificate, signed by CA1 and CA1's certificate signed by the root CA. When Bob received this message, he uses the PK_{Root} to verify PK_{CA1}, PK_{CA1} to verify PK_{Alice} and PK_{Alice} to authenticate the message. This is called traversing the trust chain of certificates, and a similar process can be undergone for messages in the reverse direction.

In cases where the certification hierarchy is extensive, including all certificates with each message can be a substantial overhead. This can be alleviated by each user keeping a copy of the certificates they receive. Rather than including the certificates in the message, the sender includes a message digest of the certificate, called a "thumbprint" in its place. The sender compares this thumbprint with a digest of each certificate that it has a copy of, and if it cannot find a match, it will ask the sender to forward a copy.

If a user's secret key becomes compromised, then the certificate associated with the public key must be revoked. The CA's in the system keep Certificate Revocation Lists (CRLs) which are available for users of the system. In order to completely trust the authenticity of the message, the CA for each certificate in the trust chain must be contacted to check that none have been revoked since they were issued. The extent of this problem will depend on two things, firstly, the number of compromised keys, and secondly, the normal period of validity of a certificate.

Appendix B MobPay Implementation & Measurement Details

In this appendix some of the modifications that were made to the CMIS Cellular IP implementation to allow for authentication and accounting in the MobPay version are highlighted. A substantial redesign of the CIP modules was required to incorporate the cryptographic extensions. New subroutines had to be added to generate the UOBT and to verify the payment parameters at each node. The NS simulator makes use of OTcl scripts to instantiate the nodes in the network. It required considerable effort to be able to add a new broker object to the network as there are complex interactions between the C++ and the OTcl code. Finally, the NS trace routines produce very detailed results, and it took a while before the relevant parameters were identified prior to parsing the trace files. A number of Perl scripts were developed to do the same.

B.1 UOBT Generation

The code below shows the generation of a UOBT by making calls to the OpenSSL API in the file watchdog.cc.

```
// Generate tree root - uobt[0][10]
RAND_bytes(*(*(uobt)+UOBT_LEN), 20);

// Generate backbone chain using SHA1
j = UOBT_LEN-1;
for (i=UOBT_LEN; i>1; i--)
{
  EVP_Digest(*(*(uobt)+i), SHA1_LEN, *(*(uobt)+j), NULL, EVP_sha1(), NULL);
  j--;
}

// Generate sub-chains of UOBT using MD5
for (i=UOBT_LEN; i>0; i--)
  for (j=0; j<UOBT_LEN; j++)
    EVP_Digest(*(*(uobt+j)+i), MD5_LEN, *(*(uobt+j+1)+i), NULL, EVP_md5(), NULL);
```

B.2 MobPay Header

The MobPay header as defined in agent.h

```
struct hdr_mobpay {
        int chainno_;
        int hashno_;
        unsigned char hashval_[MD5_LEN];
        PKCS7 *anchors_;                // Anchors to be signed by broker
        PKCS7 *sig_anchors_;            // Anchors signed by broker

        int chainno() { return (chainno_); }
        int hashno() { return (hashno_); }
        unsigned char* hashval() { return (hashval_); }
};
```

B.3 Protocol Overhead Calculations

Calculation of protocol overhead when a datagram is received by a CIP node in classifier-cip.cc

```
void CipAddressClassifier::recv(Packet* p, Handler*h)
{
.....
// MobPay - Recived page or route update - Update Cache
```

```
int i, j, k, vrfy, hashno, chainno;
unsigned char hashval[MD5_LEN], preimage[MD5_LEN], temp[MD5_LEN];

// Variables to hold crypto start and end times
static struct timezone tz;
static struct timeval _tstart, _tend;

// Start clock
crypto_time = 0;
gettimeofday(&_tstart, &tz);

// Access payment header
hdr_mobpay *mp = (hdr_mobpay*)p->access(off_mobpay_);
hashno  = mp->hashno_;
chainno = mp->chainno_;

if (hashno) {
  if (ch->ciptype_ != 4) {
    for (i=0; i<MD5_LEN; i++)
      hashval[i] = temp[i] = mp->hashval_[i];  // Make a copy of received hash value

    if (lastchainno != mp->chainno_) {         // New sub-chain of the UOBT
      for (i=0; i<MD5_LEN; i++)
        lasthash[i] = anchors[chainno][i];     // Load chain anchor as last hash value

      lastchainno = mp->chainno_;
    }
  k = hashno;
  vrfy = 1;

  while (k>0 && vrfy) {
    EVP_Digest(temp, MD5_LEN, preimage, NULL, EVP_md5(), NULL);

    if (strncmp((char *)preimage, (char *)lasthash, MD5_LEN) == 0)
      vrfy = 0;                                    // We may not always hash to the anchor
    else {                                         // Allows for greater efficiency
      for (j=0; j<MD5_LEN; j++)
        temp[j] = preimage[j];
      k--;
    }
  }
}

if (vrfy && ch->ciptype_!= 4) {
  // Drop packet and prevent cache update
  printf("MobPay - Error in verifying token (CipAddressClassifier::recv)\n");
  Packet::free(p);   // Free packet
  return;
}

// If we hashed to the anchor then we have a new sub-chain
// of the UOBT or we are a new node in the path from MN to GW
if (ch->ciptype_ != 4) {
  lasthashno = hashno;
  for (i=0; i<MD5_LEN; i++)
    lasthash[i] = hashval[i];                  // Store received hash
}

// Stop clock
gettimeofday(&_tend, &tz);
```

```
// Calculate crypto overhead in milliseconds
double t1, t2;
t1 = (double)_tstart.tv_sec + (double)_tstart.tv_usec/(1000*1000);
t2 = (double)_tend.tv_sec  + (double)_tend.tv_usec/(1000*1000);

if (ch->ciptype_ != 4) {
  crypto_time = t2-t1;
  if (!crypto_measurement)
  {
    printf("Crypto overhead is %.6f\n", crypto_time);
  }
}

//uplink
Tcl::instance().evalf("%s update_routing_table %d %d %d %d %d", name(), iph->src().addr_,
iph->dst().addr_, ch->cipnodeid_, ch->ciptype_, ch->cipsemisoft_);

ch->cipnodeid_ = cipnodeid_;
nextHop = Tcl::instance().result();
if (strcmp(nextHop, "-1"))
{
  NsObject* node = (NsObject*)TclObject::lookup(nextHop);

  // MobPay - Add crypto_time overhead to scheduler
  if (crypto_time)
  {
    Scheduler::instance().schedule(node, p->copy(), crypto_time);
    if (!crypto_measurement)
    {
      printf("Adding crypto_time of %.6f to scheduler\n", crypto_time);
      printf("*******************************************************\n");
    }
    crypto_time = 0;
  }
  else
     node->recv(p, h);

return;
.....
}
```

B.4 Perl Script for Processing MobPay Log Files

Perl script for extracting the relevant fields from the *ns-2* trace file mobpay-udp.tr

```
#!/usr/bin/perl

# This is a Perl script for extracting the relevant fields from the
# ns-2 trace file mobpay-udp.tr in the ns21b6 (MobPay) version

while (<>)
{
  # Start time (cbr)
  if (/^r -t (\S+) -Hs 8.*-It cbr -Il 70 -If (\S+) -Ii (\S+)/)
  {
        $start[$3] = $1;                # Start Time
  }

  if (/^r -t (\S+) -Hs 10.*-It cbr -Il 70 -If (\S+) -Ii (\S+)/)
  {
        $start[$3] = $1;                # Start Time
  }

  if (/^r -t (\S+) -Hs 11.*-It cbr -Il 70 -If (\S+) -Ii (\S+)/)
  {
```

```
        $start[$3] = $1;                # Start Time
    }

    if (/^r -t (\S+) -Hs 12.*-It cbr -Il 70 -If (\S+) -Ii (\S+)/)
    {
        $start[$3] = $1;                # Start Time
    }

    # End time
    if (/^\+ (\S+) 0 6 cbr 70 ------- (\S+) 1.0.2.2 0.0.0.0 (\S+) (\S+)/)
    {
        $stop[$4] = $1;                 # End Time
    }
}

# Create log file
$i = 0;
open(FP, ">mobpay-udp.log");
foreach $j (0 .. $#stop)
{
    if ($start[$j] > 0 && $stop[$j] > 0)
    {
      printf "Update# %d \n", $i;
      printf "-------------\n";
      printf "Start Time      ... %.6f\n", $start[$j];
      printf "End Time        ... %.6f\n", $stop[$j];
      printf "Elapsed Time    ... %.6f\n\n", $stop[$j] - $start[$j];

      printf FP "%.6f\n", $stop[$j] - $start[$j];
      $i++;
    }
}
close (FP);
printf "Number of Route Updates or ACKs is %d\n", $i;
```

Appendix C AdPay Implementation & Measurement Details

The majority of the changes to transform the base DSR protocol in NS into the AdPay protocol were restricted to the DSR module. However the DSR implementation for *ns-2* is complex in nature and the majority of the time was spent in understanding the flow through the code. Once there was a better understanding of the inner-workings of the code, a number of subroutines were developed to implement the micropayment protocol. A number of Perl scripts (one of which is presented below) were also developed to extract the relevant data out of the NS trace files. A summary of the measured data from which the graphs in Chapter 5 were derived is also presented in this appendix.

C.1 Perl Script for Processing AdPay Log Files

Perl script for extracting the relevant fields from the *ns-2* trace file output.tr.

```
!/usr/bin/perl
# This is a Perl script for extracting the relevant fields from the
# ns-2 trace file output.tr in the ns226 (AdPay) version

$runs = 50;
@Nodes = (10, 30, 50);
@Pause_Times = (0, 30, 60, 120, 300, 600, 900);

# Initalize variables to zero
foreach $i (0 .. $#Nodes) {
  foreach $j (0 .. $#Pause_Times) {
    $cmd = "\$nodes_$i_$j_latency = 0";
    eval $cmd;
    $cmd = "\$nodes_$i_$j_pkt_del_ratio = 0";
    eval $cmd;
  }
}

while (<>)
{
  if (/^nodes (\S+) -pause_time (\S+) -version (\S+) -sent (\S+) -rcvd (\S+) -del_ratio
(\S+) -latency (\S+)/)
  {
    $cmd = "\$nodes_$1_$2_latency = \$nodes_$1_$2_latency + $7";
    eval $cmd;

    $cmd = "\$nodes_$1_$2_pkt_del_ratio = \$nodes_$1_$2_pkt_del_ratio + $6";
    eval $cmd;
  }
}

open (LOG, ">result.log");
printf LOG "Number of runs is %d\n\n", $runs;
foreach $i (0 .. $#Nodes) {
  foreach $j (0 .. $#Pause_Times) {
    printf LOG "Nodes $Nodes[$i] Pause Time $Pause_Times[$j]\n";
    printf LOG "----------------------\n";

    printf LOG "Latency ";
    $cmd = "\$nodes_$Nodes[$i]_$Pause_Times[$j]_latency / $runs";
    printf LOG "%.2f\n", eval $cmd;

    printf LOG "Pkt Delivery Ratio ";
```

```
    $cmd = "\$nodes_$Nodes[$i]_$Pause_Times[$j]_pkt_del_ratio / $runs";
    printf LOG "%.2f\n\n\n", eval $cmd;
  }
}
close (LOG);
```

C.2 AdPay Measurements

10 Nodes

Pause Time	DSR	AdPay
0	0.85	0.81
30	0.83	0.79
60	0.84	0.8
120	0.81	0.79
300	1	1
600	1	1
900	1	1

Pause Time	DSR	AdPay
0	0.9	0.94
30	0.7	0.73
60	0.53	0.53
120	0.48	0.49
300	0.01	0.01
600	0.01	0.02
900	0.01	0.01

30 Nodes

Pause Time	DSR	AdPay
0	0.97	0.84
30	0.97	0.84
60	0.97	0.87
120	0.98	0.9
300	1	0.99
600	1	0.99
900	1	0.99

Pause Time	DSR	AdPay
0	0.32	0.46
30	0.28	0.29
60	0.22	0.23
120	0.12	0.12
300	0.04	0.06
600	0.05	0.06
900	0.05	0.06

50 Nodes

Pause Time	DSR	AdPay
0	0.97	0.86
30	0.97	0.85
60	0.98	0.87
120	0.99	0.91
300	0.99	0.98
600	1	0.99
900	1	0.98

Pause Time	DSR	AdPay
0	0.38	0.41
30	0.28	0.33
60	0.24	0.26
120	0.17	0.21
300	0.08	0.09
600	0.07	0.08
900	0.06	0.07

Packet Delivery Ratio **Average End-to-End Delay**

Table B.1 PDR and Latency Measurements for the AdPay Protocol

Appendix D SecAd Implementation & Measurement Details

This appendix provides extracts of the cryptographic extensions to the *ns-2* code base. As with the AdPay implementation, the majority of the changes were restricted to the DSR module. In particular it shows how to create and add a RDSR object to the NS code base. Most of the routines developed add the required cryptographic functionality to the code, such as digitally co-signing the route reply parameters, verifying the hash values etc. Finally, a number of Perl scripts were developed to extract the relevant data out of the NS trace files. This appendix also presents a summary of the measured data from which the graphs in Chapter 6 were derived.

D.1 RDSR Agent Modifications

Changes made to file ns-2.26/tcl/lib/ns-default.tcl

```
# Secad - RDSRAgent
Agent/RDSRAgent set sport_ 255
Agent/RDSRAgent set dport_ 255
```

Changes made to file ns-2.26/tcl/lib/ns-mobilenode.tcl

```
# Secad - RDSRAgent
Class RSRNodeNew -superclass Node/MobileNode
```

Changes made to file ns-2.26/dsr/dsragent.cc

```
// SecAd - RDSRAgent
static class RDSRAgentclass : public TclClass {
public :
        RDSRAgentclass() : TclClass("Agent/RDSRAgent") {}
        TclObject* create(int, const char*const*) {
                return (new RDSRAgent);
        }
} class_RDSRAgent;

RDSRAgent::RDSRAgent() : DSRAgent() {}

/ SecAd - Rogue DSR Agent
void RDSRAgent::handleForwarding(SRPacket &p)
  /* forward packet on to next host in source route, snooping as appropriate */
{
  hdr_sr *srh =  hdr_sr::access(p.pkt);
  hdr_ip *iph = hdr_ip::access(p.pkt);
  hdr_cmn *ch =  hdr_cmn::access(p.pkt);
  bool flowOnly = !srh->num_addrs();

  if (srh->flow_header())
    handleFlowForwarding(p);
  else if (!srh->num_addrs())
    handleDefaultForwarding(p);

  if (flowOnly)
    return;
  assert(p.pkt); // make sure flow state didn't eat the pkt

  // first make sure we are the ``current'' host along the source route.
  // if we're not, the previous node set up the source route incorrectly.
  assert(p.route[p.route.index()] == net_id
         || p.route[p.route.index()] == MAC_id);

  // SecAd
```

```
int i, j, pos, reply_len;

reply_len = srh->route_reply_len();
for (i=0; i<reply_len-1; i++)
  if (srh->reply_addrs()[i].addr == net_id.addr)
    pos = i;                                          // Note position of node

// Skip if we are the target node
if (srh->route_reply() && net_id != p.src && net_id != p.dest)
{
  // Generate new chain for this flow
  opt_chain_gen();
  if (pos == reply_len-2)                             // Node one hop away from dest
  {
    // Add anchor of own chain to list
    srh->anchors()[net_id.addr].node_id_ = net_id.addr;
    for (i=0; i<MD5_LEN; i++)
    {
      srh->anchors()[net_id.addr].anchor_[i] = anchor[i];
    }
  }
  else
  {
    // Read previous anchors and add own to list
    for (i=MAX_SR_LEN-1; i>0; i--)
    {
      if (srh->anchors()[i].node_id_ != 0)
      {
        anchor_list_[i].node_id_ = srh->anchors()[i].node_id_;
        for (j=0; j<MD5_LEN; j++)
        {
          anchor_list_[i].anchor_[j] = srh->anchors()[i].anchor_[j];
        }}}

    // Add anchor of own chain to list
    srh->anchors()[net_id.addr].node_id_ = net_id.addr;
    for (i=0; i<MD5_LEN; i++)
    {
      srh->anchors()[net_id.addr].anchor_[i] = anchor[i];
    }
  }
}

if (p.route.index() >= p.route.length())
  {
    fprintf(stderr,"dfu: ran off the end of a source route\n");
    trace("SDFU:  ran off the end of a source route\n");
    drop(p.pkt, DROP_RTR_ROUTE_LOOP);
    p.pkt = 0;
    // maybe we should send this packet back as an error...
    return;
  }
// if there's a source route, maybe we should snoop it too
if (dsragent_snoop_source_routes)
  route_cache->noticeRouteUsed(p.route, Scheduler::instance().clock(), net_id);

// sendOutPacketWithRoute will add in the size of the src hdr, so
// we have to subtract it out here
ch->size() -= srh->size();

// we need to manually decr this, since nothing else does.
```

```
  if (!iph->ttl()--) {
    drop(p.pkt, DROP_RTR_TTL);
    p.pkt = 0;
    return;
  }

  // SecAd - Forward the packet without any hash tokens
  int part_of_route = 0;
  for (i=0; i<srh->num_addrs(); i++)
    if (srh->addrs()[i].addr == net_id.addr)
      part_of_route = 1;

  if (srh->route_reply() || srh->lh_req()|| srh->lh_resp())
    sendOutPacketWithRoute(p, false);
  else
  {
    unsigned char rand[1];
    RAND_bytes(rand, 1);        // An unsigned char has value between 0 & 255

    if (rand[0] > 10 && srh->hash_num() > 1)     // Do not drop first data pkt
      p.pkt = 0;  // Drop packet
    else         // Forward packet
    {
      if (part_of_route == 1)   // If we are part of the of the route then forward pkt
      {
        if (cur_hash == srh->hash_num())
          opt_hash_gen();
        else
        {
          for(i=srh->hash_num(); i>cur_hash; i--)
          {
            opt_hash_gen();
          }
          cur_hash = srh->hash_num();
        }
        if (current == CHAIN_LEN)
        {
         for (i = 0; i < MD5_LEN; i++)
          {
            srh->hash_val()[i]  =  auth_cache_[srh->hash_num()  %  AUTH_CACHE_LEN].sent[i]  =
srh->hash_val()[i] ^ root[i];
          }
        }
        else
        {
          for (i = 0; i < MD5_LEN; i++)
           {
             srh->hash_val()[i] =   auth_cache_[srh->hash_num() % AUTH_CACHE_LEN].sent[i] =
srh->hash_val()[i] ^ th[i];
           }
        }
        sendOutPacketWithRoute(p, false);
      }
      else                                  // If not part of the flow then drop pkt
      {
        p.pkt = 0;
        return;
       }
      } } }
```

D.2 SecAd Measurements

10 Nodes

Pause Time	DSR	DSR+Mal	SecAd
0	0.63	0.42	0.45
30	0.66	0.48	0.49
60	0.57	0.46	0.5
120	0.58	0.52	0.58
300	0.78	0.7	0.78
600	0.87	0.81	0.86
900	1	0.91	0.96

Pause Time	DSR	DSR+Mal	SecAd
0	0.01	0.01	2.53
30	0.01	0.01	3.28
60	0.01	0.01	3.28
120	0.01	0.01	1.08
300	0.01	0.01	1.52
600	0.01	0.01	0.34
900	0.01	0.01	0.02

30 Nodes

Pause Time	DSR	DSR+Mal	SecAd
0	0.91	0.52	0.43
30	0.91	0.57	0.51
60	0.84	0.5	0.53
120	0.9	0.58	0.59
300	0.96	0.82	0.82
600	0.99	0.87	0.78
900	1	0.9	0.93

Pause Time	DSR	DSR+Mal	SecAd
0	0.02	0.01	5.36
30	0.02	0.01	6.65
60	0.02	0.01	0.21
120	0.02	0.01	4.01
300	0.02	0.01	1.01
600	0.01	0.01	0.04
900	0.02	0.02	0.02

Packet Delivery Ratio

Average End-to-End Delay

Table C.1 PDR and Latency Measurements for the SecAd Protocol

Appendix E Cryptographic Performance Analysis

This appendix presents the reader with the code details of the Optimal Hash Traversal Scheme that was employed in Chapter 6 and the associated Makefile. Specifically it provides the WinCE version of the code that was used to derive the measurements in Chapter 7.

E.1 Makefile

```
#
# Makefile for compilation on Windows 32 for Windows CE systems.
#
#
# Run the following command on the desktop machine. Make sure the path is # correct in the
file wcearm.bat
# c:\program files\microsoft embedded tools\evc\wce300\bin\wcearm.bat
#
# To run the make file type in 'nmake'
#
CC         = clarm
CPP        = clarm /TP
LD         = link
EVC        = "C:\Program Files\Microsoft eMbedded Tools\EVC"
WCEROOT    = "C:\Windows CE Tools"
PLATFORM   = "Pocket PC 2002"
OSVERSION  = WCE300
CEVERSION  = 300
CESUBSYS   = WINDOWSCE
MACHINE    = /machine:arm

INCLUDE    = -I$(WCEROOT)/$(OSVERSION)/$(PLATFORM)/include -I"." -
I"c:\pocketconsole\include" -I"c:\wceopenssl\include"

LIB        = -libpath:$(WCEROOT)/$(OSVERSION)/$(PLATFORM)/lib/arm -
libpath:"c:\pocketconsole\lib" -libpath:"c:\wceopenssl\lib"

CFLAGS     = -nologo -D_ARM_ -DARM -D_arm_ -DPOCKET_SIZE -DPALM_SIZE \
             -D_WINCE_ -D_WINCE -D_WIN32_WCE=$(CEVERSION) -DUNDER_CE=$(CEVERSION) -
             DUNICODE -D_UNICODE -w

LIBS       = commctrl.lib coredll.lib winsock.lib portlib.lib libeay32.lib
             ssleay32.lib

ENTRY      = -entry:mainACRTStartup


all:           isit

clean:
               del /F *.obj *.exe

isit.obj:      isit.c
               $(CC) $(CFLAGS) $(INCLUDE) -c isit.c
isit:          isit.obj
               $(LD) -subsystem:$(CESUBSYS) $(MACHINE) $(ENTRY) $(LIB) \
               -out:isit.exe isit.obj $(LIBS)
# EOF
```

E.2 Optimal Hash Sequence Traversal

```
/* Implementation of Markus Jakobsson's fractal hash tree traversal */

#include <stdio.h>
#include <stdlib.h>
#include <windows.h>
#include <openssl/evp.h>
#include <openssl/ssl.h>

#define MD5_LEN 16
```

```
#define CHAIN_LEN 1024
#define ITERATIONS  100000
#define OUTPUT_FILE "isit.txt"
#define COPY(s,d) {s.pos=d.pos;s.si=d.si;s.di=d.di;s.dest=d.dest;s.value=d.value;}

static const char rnd_seed[] = "string to make the random number generator think it has
entropy";

// Optimal hash chain traversal technique
void grid (unsigned char *root, DWORD *total_time)
{
  int i, j, k;
  EVP_MD_CTX ctx;
  int n, NRP, current;
  DWORD tv1, tv2, time;
  unsigned int num_bytes;

  struct pebbles
  {
    int pos;
    int si;
    int di;
    int dest;
    unsigned char value[MD5_LEN];
  };

  struct pebbles p[10];
  unsigned char hashchain[CHAIN_LEN+1][MD5_LEN];
  unsigned char th[MD5_LEN], tmp1[MD5_LEN], tmp2[MD5_LEN];

  n=NRP=10;

  for (j=0; j<MD5_LEN; j++)
   hashchain[CHAIN_LEN][j] = root[j];

  for (i = CHAIN_LEN, j = CHAIN_LEN-1; i > 0; i--, j--)
  {
       EVP_DigestInit(&ctx, EVP_md5());
       EVP_DigestUpdate(&ctx, *(hashchain+i), MD5_LEN);
    EVP_DigestFinal(&ctx, *(hashchain+j), &num_bytes);
  }

  // System pre-calculations
  for (i=0; i<NRP; i++)
  {
    p[i].si = 3 * (1 << (i + 1));
    p[i].di = 2 * (1 << (i + 1));
    p[i].pos = (1 << (i + 1));
    p[i].dest = (1 << (i + 1));

    // Copy chain root
    for (j=0; j<MD5_LEN; j++)
         tmp1[j] = root[j];

       // Hash to pebble+1 and store
    for (j=CHAIN_LEN; j>(1<<(i+1))-1; j--)
    {
      EVP_DigestInit(&ctx, EVP_md5());
          EVP_DigestUpdate(&ctx, tmp1, MD5_LEN);
      EVP_DigestFinal(&ctx, tmp2, &num_bytes);
```

```
        for (k=0; k<MD5_LEN; k++)
          tmp1[k] = tmp2[k];
    }
    for (k=0; k<MD5_LEN; k++)
        (p+i)->value[k] = tmp1[k];
  }

  current = 0;

tv1 = GetTickCount();          // Start Clock
  while (current != (1 << n))
  {
    current++;

    for (i=0; i<NRP; i++)
        {
          if (p[i].pos != p[i].dest)
          {
            p[i].pos -= 2;
                EVP_DigestInit(&ctx, EVP_md5());
              EVP_DigestUpdate(&ctx, (p+i)->value, MD5_LEN);
          EVP_DigestFinal(&ctx, tmp1, &num_bytes);

                EVP_DigestInit(&ctx, EVP_md5());
              EVP_DigestUpdate(&ctx, tmp1, MD5_LEN);
          EVP_DigestFinal(&ctx, (p+i)->value, &num_bytes);
          }
        }

    if ((current & 1) == 0)
        {
          EVP_DigestInit(&ctx, EVP_md5());
          EVP_DigestUpdate(&ctx, p->value, MD5_LEN);
      EVP_DigestFinal(&ctx, th, &num_bytes);
    }
    else
        {
          for(k=0; k<MD5_LEN; k++)
            th[k] = p[0].value[k];

          p[0].pos += p[0].si;
          p[0].dest += p[0].di;

          if (p[0].dest >= (1 << n))
            p[0].dest = p[0].pos = 1000;
          else
          {
            for (i=1; i<NRP; i++)
                  if (p[i].pos == p[0].pos)
                  {
                        for(k=0; k<MD5_LEN; k++)
                      p[0].value[k] = p[i].value[k];
                    break;
                  }
          }

          // sort
          for (i=0; i<NRP; i++)
            for (j = i + 1; j < NRP; j++)
              if (p[i].pos > p[j].pos)
                  {
```

```
                struct pebbles tmp;

tmp.pos=p[i].pos;tmp.si=p[i].si;tmp.di=p[i].di;tmp.dest=p[i].dest;for(k=0;  k<MD5_LEN;  k++)
tmp.value[k]=p[i].value[k];

p[i].pos=p[j].pos;p[i].si=p[j].si;p[i].di=p[j].di;p[i].dest=p[j].dest;for(k=0;    k<MD5_LEN;
k++) p[i].value[k]=p[j].value[k];

p[j].pos=tmp.pos;p[j].si=tmp.si;p[j].di=tmp.di;p[j].dest=tmp.dest;for(k=0;  k<MD5_LEN;  k++)
p[j].value[k]=tmp.value[k];
                }
        }
  }
  tv2 = GetTickCount();               // Stop Clock
  time = (tv2 - tv1);
  *total_time = *total_time + time;
}

int
main ()
{
  int i;
  FILE *output;
  DWORD total_time;
  unsigned char root[MD5_LEN];

  total_time = 0;
  RAND_seed(rnd_seed, sizeof(rnd_seed));

  for (i=0; i<ITERATIONS; i++)
  {
    RAND_bytes(root, MD5_LEN);
    grid (root, &total_time);
  }

  output = fopen (OUTPUT_FILE, "w");
  fprintf (output, "%d\n", total_time);
  fclose(output);
  return (EXIT_SUCCESS);
}
```

Appendix F OpenSSL X.509 Certificates

This appendix provides details of the structure of a X.509 certificate, and how to derive the same using the OpenSSL development environment.

F.1 Certificate Creation with OpenSSL

```
# Info about a certificate:
openssl x509 -in client.pem -text -noout

# Create a root CA certificate (self-signed):
openssl req -new -x509 -nodes -keyout cakey.pem -out cacert.pem -days 3650

# Create a new private key and a certificate request:
openssl req -new -nodes -keyout bk.pem -out bk.pem -days 2000

# Sign the above certificate request and create a certificate:
openssl ca -policy policy_anything -out bk-cert.pem -infiles bk.pem
```

F.2 X.509 Certificate

```
Certificate:
    Data:
        Version: 3 (0x2)
        Serial Number: 0 (0x0)
        Signature Algorithm: sha1WithRSAEncryption
        Issuer: C=IE, ST=Ireland, L=Dublin, O=Trinity College Dublin, OU=NTRG,
CN=CA/emailAddress=ca@tcd.ie
        Validity
            Not Before: Aug 11 13:37:20 2003 GMT
            Not After : Aug  8 13:37:20 2013 GMT
        Subject: C=IE, ST=Ireland, L=Dublin, O=Trinity College Dublin, OU=NTRG,
CN=BK/emailAddress=bk@tcd.ie
        Subject Public Key Info:
            Public Key Algorithm: rsaEncryption
            RSA Public Key: (2048 bit)
                Modulus (2048 bit):
                    00:cd:05:a5:09:56:34:d8:21:0a:95:c8:1f:d2:3c:
                    cb:3d:ec:2a:9d:a8:ad:79:c3:ca:bc:02:ca:ed:7a:
                    42:be:6c:a8:22:17:be:0c:f6:9c:33:39:b4:30:71:
                    1a:c2:5a:18:31:9e:fe:b6:09:ea:71:7d:27:f7:24:
                    7b:12:22:1f:54:f8:5f:e7:02:71:b6:d4:96:a5:d8:
                    57:8a:e3:d1:00:82:85:91:36:93:9a:40:a3:eb:ce:
                    33:43:24:95:d1:8f:02:32:4d:de:d9:a6:ab:04:a1:
                    bc:25:c5:4d:cc:74:10:78:99:08:34:06:98:87:3b:
                    dc:b2:ef:b6:9f:30:c3:57:75:6f:ff:70:95:2f:c3:
                    1d:c8:ef:42:8a:74:f4:6d:de:ca:f1:c9:da:13:ee:
                    3e:4e:7b:c8:2d:13:57:f0:76:34:56:da:41:5c:3d:
                    19:dd:5b:8f:16:2c:13:68:67:82:34:5a:9f:89:4d:
                    ee:6a:69:2e:bf:92:63:34:ee:cb:72:39:9a:d3:9b:
                    04:57:37:92:d6:1b:cb:e3:80:3d:89:51:e0:18:40:
                    bd:59:95:e0:30:0f:9e:e3:f6:4d:15:c0:34:a9:1f:
                    75:b0:8e:79:78:0d:f3:19:b9:28:5e:d3:32:db:ed:
                    b2:8f:9e:d6:b3:4b:9c:12:68:c5:69:70:0f:be:9e:
                    3d:cf
                Exponent: 65537 (0x10001)
        X509v3 extensions:
            X509v3 Basic Constraints:
            CA:FALSE
            Netscape Comment:
            Certificate issued by NTRG-CA
            X509v3 Subject Key Identifier:
            13:F8:A5:FC:1D:45:B4:D9:59:60:3C:7A:6E:24:27:1B:EF:BC:7B:64
            X509v3 Authority Key Identifier:
            keyid:18:67:31:DB:23:D7:16:D8:95:93:D0:6C:43:7B:B1:F7:D7:30:03:E7
            DirName:/C=IE/ST=Ireland/L=Dublin/O=Trinity College
Dublin/OU=NTRG/CN=CA/emailAddress=ca@tcd.ie
            serial:00
```

2048-bit RSA Public Key

```
            Netscape Base Url:
            https://ntrg.cs.tcd.ie/ca/
            Netscape CA Revocation Url:
            https://ntrg.cs.tcd.ie/ca/ntrg-ca.crl
            Netscape Revocation Url:
            https://ntrg.cs.tcd.ie/ca/revocation.html?
            Netscape Renewal Url:
            https://ntrg.cs.tcd.ie/ca/renewal.html?
            Netscape CA Policy Url:
            https://ntrg.cs.tcd.ie/ca/policy.html
            X509v3 Issuer Alternative Name:
            URI:https://ntrg.cs.tcd.ie/ssl/ntrg.crt
            X509v3 CRL Distribution Points:
            URI:https://ntrg.cs.tcd.ie/ca/ntrg-ca.crl

    Signature Algorithm: sha1WithRSAEncryption
        a0:19:6b:21:6e:d8:de:72:d7:05:3f:ce:28:82:99:74:b2:6a:
        1a:fe:27:7d:fb:f4:f7:c5:ef:e7:f1:bc:f1:28:c6:04:58:b0:
        da:ca:74:45:a8:00:d1:33:03:72:8c:66:eb:13:9f:ff:7b:6f:
        76:76:49:c8:f6:fa:89:c2:52:0e:d6:69:45:0b:ac:57:b4:5e:
        fa:5e:62:6e:f1:bd:80:7a:82:99:45:28:dc:4e:11:79:b0:fa:
        c0:b2:c3:6a:cc:d9:8c:89:78:4a:c4:6e:3f:54:ac:bf:94:46:
        20:56:68:7c:85:59:30:42:d4:6c:63:f5:76:d0:07:9a:45:f7:
        c4:4a:c1:bb:bd:5f:f0:4d:85:44:95:3d:44:80:26:e3:57:85:
        41:e6:21:7b:cd:33:60:9a:b1:2c:48:ee:11:8b:f2:83:62:2a:
        69:55:99:e9:5f:fd:b1:09:f5:8e:34:8d:11:bc:52:c1:d5:4a:
        21:bb:44:a0:51:3e:3e:4e:8d:9a:28:d8:f8:1b:a9:85:6c:2a:
        ef:39:7b:e2:2f:96:a8:82:9a:51:b0:71:0e:b4:06:a4:17:1e:
        f5:6b:6b:f7:4b:f9:8b:42:5f:04:e0:cd:a1:7d:11:c1:a2:e9:
        6c:73:23:f7:cf:85:21:79:3f:2d:ca:e4:1e:e3:25:c7:b9:2f:
        66:a7:72:07
-----BEGIN CERTIFICATE-----
MIIGEjCCBPqgAwIBAgIBADANBgkqhkiG9w0BAQUFADCBhzELMAkGA1UEBhMCSUUx
EDAOBgNVBAgTB0lyZWxhbmQxDzANBgNVBAcTBkRlYmxpbjEfMB0GA1UEChMWVHJp
bml0eSBDb2xsZWdlIER1YmxpbjENMAsGA1UECxMETlRSRzELMAkGA1UEAxMCQ0Ex
GDAWBgkqhkiG9w0BCQEWCWNhQHRjZC5pZTAeFw0wMzA4MTExMzM3MjBaFw0xMzA4
MDgxMzM3MjBaMIGHMQswCQYDVQQGEwJJRTEQMA4GA1UECBMHSXJlbGFuZDEPMA0G
A1UEBxMGRHVibGluMR8wHQYDVQQKExZUcmluaXR5IENvbGxlZ2UgRHVibGluMQ0w
CwYDVQQLEwROVFJHMQswCQYDVQQDEwJCSzEYMBYGCSqGSIb3DQEJARYJYmtAdGNk
LmllMIIBIjANBgkqhkiG9w0BAQEFAAOCAQ8AMIIBCgKCAQEAzQWlCVY02CEKlcgf
0jzLPewqnaitecPKvALK7XpCvmyoIhe+DPacMzm0MHEawloYMZ7+tgnqcX0n9yR7
EiIfVPhf5wJxttSWpdhXiuPRAIKFkTaTmkCj684zQySV0Y8CMk3e2aarBKG8JcVN
zHQQeJkINAaYhzvcsu+2nzDDV3Vv/3CVL8MdyO9CinT0bd7K8cnaE+4+TnvILRNX
8HY0VtpBXD0Z3VuPFiwTaGeCNFqfiU3uamkuv5JjNO7Lcjma05sEVzeS1hvL44A9
iVHgGEC9WZXgMA+e4/ZNFcA0qR91sI55eA3zGbkoXtMy2+2yj57Ws0ucEmjFaXAP
vp49zwIDAQABo4IChTCCAoEwCQYDVR0TBAIwADAsBglghkgBhvhCAQ0EHxYdQ2Vy
dGlmaWNhdGUgaXNzdWVkIGJ5IE5UUkctQ0EwHQYDVR0OBBYEFBP4pfwdRbTZWWA8
em4kJxvvvHtkMIG0BgNVHSMEgawwgamAFBhnMdsj1xbYlZPQbEN7sffXMAPnoYGN
pIGKMIGHMQswCQYDVQQGEwJJRTEQMA4GA1UECBMHSXJlbGFuZDEPMA0GA1UEBxMG
RHVibGluMR8wHQYDVQQKExZUcmluaXR5IENvbGxlZ2UgRHVibGluMQ0wCwYDVQQL
EwROVFJHMQswCQYDVQQDEwJDQTEYMBYGCSqGSIb3DQEJARYJY2FAdGNkLmllggEA
MCkGCWCGSAGG+EIBAgQcFhpodHRwczovL250cmcuY3MudGNkLmllL2NhLzA0Bglg
hkgBhvhCAQQEJxYlaHR0cHM6Ly9udHJnLmNzLnRjZC5pZS9jYS9udHJnLWNhLmNy
bDA5BglghkgBhvhCAQMELBYqaHR0cHM6Ly9udHJnLmNzLnRjZC5pZS9jYS9yZXZv
Y2F0aW9uLmh0bWw/MDYGCWCGSAGG+EIBBwQpFidodHRwczovL250cmcuY3MudGNk
LmllL2NhL3JlbmV3YWwuaHRtbD8wNAYJYIZIAYb4QgEIBCcWJWh0dHBzOi8vbnRy
Zy5jcy50Y2QuaWUvY2EvcG9saWN5Lmh0bWwwLgYDVR0SBCcwJYYjaHR0cHM6Ly9u
dHJnLmNzLnRjZC5pZS9zc2wvbnRyZy5jcnQwNgYDVR0fBC8wLTAroCmgJ4YlaHR0
cHM6Ly9udHJnLmNzLnRjZC5pZS9jYS9udHJnLWNhLmNybDANBgkqhkiG9w0BAQUF
AAOCAQEAoBlrIW7Y3nLXBT/OKIKZdLJqGv4nffv098Xv5/G88SjGBFiw2sp0RagA
0TMDcoxm6xOf/3tvdnZJyPb6icJSDtZpRQusV7Re+l5ibvG9gHqCmUUo3E4RebD6
wLLDaszZjIl4SsRuP1Ssv5RGIFZofIVZMELUbGP1dtAHmkX3xErBu71f8E2FRJU9
RIAm41eFQeYhe80zYJqxLEjuEYvyg2IqaVWZ6V/9sQn1jjSNEbxSwdVKIbtEoFE+
Pk6NmijY+BuphWwq7zl74i+WqIKaUbBxDrQGpBce9Wtr90v5i0JfBODNoX0RwaLp
bHMj98+FIXk/LcrkHuMlx7kvZqdyBw==
-----END CERTIFICATE-----
```

X.509 Digital Certificate

Bibliography

(A complete online listing of all the references below can found at – http://www.cs.tcd.ie/~htewari/bib.html)

[3GPP] Third Generation Partnership Project, http://www.3gpp.org/

[3GPP03a] 3GPP TS 23.002, "3rd Generation Partnership Project; Technical Specification Group Services and System Aspects; Network Architecture", version. 6.3.0, Dec. 2003.

[3GPP03b] 3GPP TS 32.015, "3rd Generation Partnership Project; Technical Specification Group Services and System Aspects; Telecommunication Management; Charging Management; Call and Event Data for the Packet Switched Domain", version .3.12.0, Dec. 2003.

[3GPP04] 3GPP TS 22.115, "3rd Generation Partnership Project; Technical Specification Group Services and System Aspects Service Aspects; Charging and Billing, version 6.3.0, Jan. 2004.

[3GPP2] Third Generation Partnership Project 2, http://www.3gpp2.org/

[3GPT00] 3GPP2 P.R0001: Wireless IP Architecture Based on IETF Protocols, version 1.0.0, July 2000.

[3GPT01] 3GPP2 S.R0005-B: Network Reference Model for cdma2000 Spread Spectrum Systems, version 1.0, April 2001.

[3GPT03] 3GPP2 X.S0011-001-C: cdma2000 Wireless IP Network Standard: Introduction, version 1.0.0, Aug. 2003.

[4GT] The 4th Generation Telephony Concept, http://ntrg.cs.tcd.ie/4genhome.php.

[AAA] Authentication, Authorization and Accounting (AAA) Working Group Charter, http://www.ietf.org/html.charters/aaa-charter.html.

[Abo99] B. Aboba, "The Network Access Identifier", IETF RFC 2486, Jan. 1999.

[AESL05] AES Lounge, http://www.iaik.tu-graz.ac.at/research/krypto/AES/

[AH05] G. Antell and W. Harris, *Economics: Institutions and Analysis*, AMSCO School Publications, New York, 2005, http://www.amscopub.com/frameset.htm

[AHP03] K. Ahmavaara, H. Haverinen and R. Pichna, "Interworking Architecture between 3GPP and WLAN Systems", *IEEE Communications*, vol. 41, no. 11, Nov. 2003, pp. 74-81.

[AJSW97] N. Asokan, P. Jason, M. Steiner and M. Waidner, "The State of the Art in Electronic Payment Systems", *IEEE Computer*, vol. 30, no. 9, Sep. 1997, pp. 28-35.

[ANSI T1] T1 Committee of the Telecommunications of the American National Standards Institute, http://www.t1.org/

[APPS00] G. Apostolopoulos, V. Peris, P. Paradhan and D. Saha, "Securing Electronic Commerce: Reducing the SSL Overhead", *IEEE Network*, vol. 14, no. 4, Aug. 2000, pp. 8-16.

[ARIB] Association of Radio Industries and Business, Japan, http://www.arib.or.jp/english/index.html.

[ASWZ02] W. Arabaugh, N. Shankar, Y. Wan and K. Zhang, "Your 802.11 Wireless Network Has No Clothes", *IEEE Wireless Communications*, vol. 9, no. 6, Dec. 2002, pp. 44-51.

[AVTO04] P. Argyroudis, R. Verma, H. Tewari and D. O'Mahony, "Performance Analysis of Cryptographic Protocols on Handheld Devices", *Proceedings of the 3rd IEEE International Symposium on Network Computing and Applications*, Cambridge, MA, Aug 30 - Sep 1, 2004.

[AXM04] I. Akyildiz, J. Xie and S. Mohanty, "A Survey of Mobility Management in Next-Generation All-IP Based Wireless Systems", *IEEE Wireless Communications*, vol. 11, no. 4, Aug. 2004, pp. 16-28.

[BA01] S. Bhargava and D. Agrawal, "Security Enhancements in AODV protocol for Wireless Ad Hoc Networks", *Vehicular Technology Conference*, Atlantic City, Oct. 7-11, 2001.

[Bar05] P. Barreto, "Hashing Function Lounge", http://paginas.terra.com.br/informatica/paulobarreto/hflounge.html

[BB02] S. Buchegger and J-Y. Le Boudec, "Performance Analysis of the CONFIDANT Protocol (Cooperation of Nodes – Fairness in Dynamic Ad-hoc NeTworks)", *Proceedings of the 3rd Symposium on Mobile Ad Hoc Networking and Computing (MobiHoc 2002)*, ACM Press, 2002, pp. 226-236.

[BBC+94] J. Boly, A. Bosselaers, R. Cramer, R. Michelsen, S. Mjølsnes, F. Muller, T. Pedersen, B. Pfitzmann, P. Rooij, B. Schoenmakers, M. Schunter, L. Vallée and M Waidner, "The ESPRIT project CAFE - High Security Digital Payment Systems", *Proceedings of Computer Security - ESORICS '94*, LCNS, vol. 875, Springer-Verlag, Berlin, 1994, pp. 217-230.

[BBCD+99] S. Blake, D. Black, M. Carlson, E. Davies, Z. Wang and W. Weiss, "An Architecture for Differentiated Services", IETF RFC 2475, Dec. 1999.

[BBCG+01] L. Blazevic, L. Buttyan, S. Capkun, S. Giordano, J-P. Hubaux and J-Y Le Boudec, "Self-Organization in Mobile Ad Hoc Networks: The Approach of Terminodes", *IEEE Communications*, vol. 39, no. 6, June 2001, pp. 166-174.

[BBT02] R. Berezdivin, R. Breinig and R. Topp, "Next-Generation Wireless Communications Concepts and Technologies", *IEEE Communications*, vol. 40, no. 3, March 2002, pp. 108-116.

[BCHL+03] M. Buddhikot, G. Chandranmenon, S. Han, Y-W Lee, S. Miller and L. Salgarelli, "Design and Implementation of a WLAN/CDMA2000 Interworking Architecture", *IEEE Communications*, vol. 41, no. 11, Nov. 2003, pp. 90-100.

[BCJ00] K. Basu, A. Campbell and A. Joseph (Editors). Special Issue on IP-Based Mobile Telecommunications Networks. *IEEE Personal Communications*, vol. 7, no. 4, Aug. 2000.

[BGQS+01] A. Bria, F. Gessler, O. Queseth, R. Stridh, M. Unbehaun, J. Wu, J. Zander and M. Flament, "4th-Generation Wireless Infrastructure: Scenarios and Research Challenges", *IEEE Personal Communications*, vol. 8, no. 6, Dec. 2001, pp. 25-31.

[BH01] L. Buttyán and J.-P. Hubaux, "Nuglets: A Virtual Currency to Stimulate Cooperation in Self-Organized Mobile Ad Hoc Networks", Tech Report DSC/2001/001, Swiss Federal Institute of Technology, Lausanne, 2001.

[BIS00] Bank of International Settlements, "Statistics on Payment Systems in the Group of Ten Countries", Basel, Switzerland, Feb. 2000, http://www.bis.org/publ.

[BL01] L. Bos and S. Leroy, "Toward and All-IP-Based UMTS Architecture", *IEEE Network*, vol. 15, no. 1, Jan. 2001, pp. 36-45.

[BMA02] M. Benzaid, P. Minet and K. Al Agha, "Integrating Fast Mobility in the OLSR Routing Protocol", *Proceedings of the 4th IEEE International Workshop on Mobile and Wireless Communications Network*, Stockholm, Sep. 2002, pp. 217-221.

[Boingo] Boingo Wireless, http://www.boingo.com/

[Boingo03] "Towards Ubiquitous Wireless Broadband", Boingo Wireless Industry White Paper, Sep. 2003, http://www.boingo.com/wi-fi_industry_basics.pdf.

[Bos05] A. Bosselaers, "Fast Implementation on the Pentium", http://www.esat.kuleuven.ac.be/~bosselae/

[BPIM+01] L. Becchetti, F. Priscoli, T. Inzerlli, P. Mahonen and L. Munoz, "Enhancing IP Service Provision over Heterogeneous Wireless Networks: A Path toward 4G", *IEEE Communications*, vol. 39, no. 8, Aug. 2001, pp. 74-81.

[Bra01] E. Bradford, "RunTime: High-performance Programming Techniques on Linux and Windows 2000 – Setting Up Timing Routines", IBM developerWorks, April 2001, http://www-106.ibm.com/developerworks/library/l-rt1/

[BRAN] Broadband Radio Access Networks, http://portal.etsi.org/bran/Summary.asp.

[BT] Bluetooth, http://www.bluetooth.com/

[BVE99] C. Bettstetter, H-J Vogel and J Eberspacher, "GSR Phase 2+ General Packet Radio Service GPRS: Architecture, Protocols, and Air Interface", *IEEE Communications Surveys*, vol. 2, no. 3, 1999, pp. 2-14.

[BW97] G. Brasche and B. Walke, "Concepts, Services and Protocol of the New GSM Phase 2+ General Packet Radio Service", *IEEE Communications*, vol. 35, no. 8, Aug. 1997, pp. 94-104.

[BWDD+03] N. Banerjee, W. Wu, S. K. Das, S. Dawkins and J. Pathak, "Mobility Support in Wireless Internet", *IEEE Wireless Communications*, vol. 10, no. 5, Oct. 2003, pp. 54-61.

[BWorld] Billing World, http://www.billingworld.com/

[CEPS00a] Common Electronic Purse Specifications – Business Requirements", Version 7.0, March 2000, http://www.cepsco.com/

[CEPS00b] "Common Electronic Purse Specifications – Functional Requirements", Version 6.3, Sep. 1999, http://www.cepsco.com/

[CEPS00c] "Common Electronic Purse Specifications – Technical Specifications", Version 2.2, May 2000, http://www.cepsco.com/

[CG97] J. Cai and D. Goodman, "General Packet Radio Service in GSM", *IEEE Communications*, vol. 35, no. 10, Oct. 1997, pp. 122-131.

[CGKV+00] A. Campbell, J. Gomez, S. Kim, A. Valko, C.-Y. Wan and Z. Turanyi, "Design, Implementation, and Evaluation of Cellular IP", *IEEE Personal Communications*, vol. 7, no. 4, Aug. 2000, pp. 42-49.

[CGKW+02] A. Campbell, J. Gomez, S. Kim, C.-Y. Wan, Z. Turanyi and A. Valko, "Comparison of IP Micromobility Protocols", *IEEE Wireless Communications*, vol. 9, no. 1, Feb. 2002, pp. 72-82.

[CH00] J. Cushine and D. Hutchison, "Charging and Billing for GSM and Future Mobile Internet Services", *Proceedings of the 1st Prostgraduate Symposium- PGNet*, Liverpool, UK, June 2000, http://www.cms.livjm.ac.uk/pgnet/

[Cha82] D. Chaum, "Blind Signatures for Untraceable Payments", *Proceedings of Advances in Cryptology – CRYPTO '82*, Plenum Press, New York, 1983, pp. 199-203.

[Cha85] D. Chaum, "Security Without Identification: Transaction Systems to Make Big Brother Obsolete", *Communications of the ACM*, vol. 28, no. 10, Oct. 1985, pp. 1030-1044.

[Cha92] D. Chaum, "Achieving Electronic Privacy", Scientific American (invited), Aug. 1992, pp. 96-101.

[Cha04] David Chaum's Home Page, http://www.chaum.com/

[CIP] Columbia Cellular IP Protocol, http://comet.ctr.columbia.edu/cellularip/

[CISCO01] "Billing for Mobile Wireless Data Services", Cisco Systems, White Paper, 2001, http://www.cisco.com/

[CJ02] D. Coppersmith and M. Jakobsson, "Almost Optimal Hash Sequence Traversal", *Proceedings of the 6th Intl Financial Cryptography Conference*, Bermuda, March, 2002.

[CJ03] T. Clausen and P. Jacquet Eds., "Optimized Like State Routing Protocol (OLSR)", IETF RFC 3626, Oct. 2003.

[CJPH04] P Calhoun, T. Johansson, C. Perkins and T. Hiller, "Diameter Mobile IPv4 Application", draft-ietf-aaa-diameter-mobileip-16.txt, work in progress, Feb. 2004.

[CKK02] F. Chiussi, D. Khotimsky and S. Krishnan, "Mobility Management in Third-Generation All-IP Networks", *IEEE Communications*, vol. 40, no. 9, Sep. 2002, pp. 124-135.

[CLGZ+03] P. Calhoun, J. Loughney, E. Guttman, G. Zorn and J. Arkko, "Diameter Base Protocol", IETF RFC 3588, Sep. 2003.

[CMIS] Columbia IP Micromobility Software, http://comet.columbia.edu/micromobility/

[COBU96] ACTS Project AC031 COBUCO. Cordless Business Communications System (COBUCO) – System Concept and Architecture. COBUCO Deliverable 01, Feb. 1996.

[COBU98] ACTS Project AC031 COBUCO – Final Report. COBUCO Deliverable 20, Oct. 1998.

[CPP99] K. Chen, R. Prasad and H. Poor (Editors). Special Issue on Software Radio. *IEEE Personal Communications*, vol. 6, no. 4, Aug. 1999.

[Crypto05] Cryptography Research, Hash Collisions Q&A, March 2005 http://www.cryptography.com/cnews/hash.html.

[CS97] L. Camp and M. Sirbu, "Critical Issues in Internet Commerce", *IEEE Communications*, vol. 35, no. 5, May 1997, pp. 58-62.

[CWTS] China Wireless Telecommunications Standards Group, China, http://www.cwts.org/english/index.php.

[DARPA] The Defense Advanced Research Projects Agency, http://www.darpa.mil/

[DB99] N. Daswami and D. Boneh, "Experimenting with Electronic Commerece on the PalmPilot", *Proceedings of Eurocrypt '99*, LNCS 1648, Springer Verlag, Feb. 1999, pp. 1-16.

[DGA01] S. Dixit, Y. Guo and Z. Antoniou, "Resource Management and Quality of Service in Third-Generation Wireless Networks", *IEEE Communications*, vol. 39, no. 2, Feb. 2001, pp. 125-133.

[DH76] W. Diffie and M. Hellman, "New Directions in Cryptography", *IEEE Transactions on Information Theory*, vol. 22, Nov. 1976, pp. 74-78.

[DIDATA03] "Telecommunications Fraud", Dimension Data, White Paper, March 2003, http://www.didata.com/

[DLA02] H. Deng, W. Li and D. Agrawal, "Routing Security in Wireless Ad Hoc Networks", *IEEE Communications*, vol. 40, no. 10, Oct. 2002, pp. 70-75.

[DMDM+02] S. Das, A. McAuley, A. Dutta, A. Misra, K. Chakraborty and S. Das, "IDMP: An Intradomain Mobility Management Protocol for Next-Generation Wireless Networks", *IEEE Wireless Communications*, vol. 9, no. 3, June 2002, pp. 38-45.

[DTNA+03] A. Doufexi, E. Tameh, A. Nix, S. Armour and A. Molina, "Hotspot Wireless LANs to Enhance the Performance of 3G and Beyond Cellular Networks", *IEEE Communications*, vol. 41, no. 7, July 2003, pp. 58-65.

[EComm] M. Merkow, "MasterCard's Response to the Online Payments Quandary", News Article ecommerce-guide.com, http://www.ecommerce-guide.com/news/trends/article.php/952181

[Edd04] W. Eddy, "At What Layer Does Mobility Belong?", *IEEE Communications*, vol. 42, no. 10, Oct. 2004, pp. 155-159.

[EHHK+01] I. Elsen, F. Hartung, U. Horn, M. Kampmann and L. Peters, "Streaming Technology in 3G Mobile Communications Systems", *IEEE Computer*, vol. 34, no. 9, Sep. 2001, pp. 46-52.

[EMV00a] *EMV2000 Integrated Circuit Card Specification for Payment Systems*, Book 1 – Application Independent ICC to Terminal Interface Requirements, Version 4.0, Dec. 2000, http://www.emvco.com/

[EMV00b] *EMV2000 Integrated Circuit Card Specification for Payment Systems*, Book 2 – Security and Key Management, Version 4.0, Dec. 2000, http://www.emvco.com/

[EMV00c] *EMV2000 Integrated Circuit Card Specification for Payment Systems*, Book 3 – Application Specification, Version 4.0, Dec. 2000, http://www.emvco.com/

[EMV00d] *EMV2000 Integrated Circuit Card Specification for Payment Systems*, Book 4 – Cardholder, Attendant, and Acquirer interface Requirements, Version 4.0, Dec. 2000.

[EPS1] Electronic Payment Schemes, http://www.w3.org/ECommerce/roadmap.html.

[EPS2] Network Payment Mechanisms, http://ntrg.cs.tcd.ie/mepeirce/project.html.

[ETSI] European Telecommunications Standards Institute, http://www.etsi.org/

[ETSI00a] ETSI TS 101 393: Digital Cellular Telecommunications System (phase 2+); General Packet Radio Service (GPRS); GPRS Charging, version 7.5.0, June 2000.

[Euro99] Europay International, "Mobile Commerce New Opportunities for SET", Oct. 1999.

[FCC03] U.S. Federal Communications Commission, "Quarterly Report on Informal Consumer Enquiries and Complaints Released", May 2003, http://www.fcc.gov/cgb/

[For03] B. Forouzan, *TCP/IP Protocol Suite*, 2nd Ed., McGraw Hill Publishers, 2003.

[FLP04] S. Faccin, P. Lalwaney and B. Patil, "IP Multimedia Services: Analysis of Mobile IP and SIP Interactions in 3G Networks", *IEEE Communications*, vol. 42, no. 1, Jan. 2004, pp. 113-120.

[FSTC] Financial Service Technology Consortium, http://www.fstc.org/

[FV03] K. Fall and K. Vardhan, "The ns Manual", Dec. 2003, http://www.isi.edu/nsnam/ns/ns-documentation.html

[Gar02] L. Garber, "Will 3G Really be the Next Big Wireless Technology", *IEEE Computer*, vol. 35, no. 1, Jan. 2002, pp. 26-32.

[GEN01] A. Grilo, P. Estrela and M. Nunes, "Terminal Independent Mobility (TIMIP)", *IEEE Communications*, vol. 39, no. 12, Dec. 2001, pp. 34-41.

[GG01] V. Gupta and S. Gupta, "Securing the Wireless Internet", *IEEE Communications*, vol. 39, no. 12, Dec. 2001, pp. 68-74.

[GHJP00] S. Glass, T. Hiller, S. Jacobs and C. Perkins, "Mobile IP Authentication, Authorization and Accounting Requirements", IETF RFC 2977, Oct. 2000, http://www.ietf.org/html.charters/OLD/mobileip-charter.html.

[Gin00] P. Ginzboorg, "Seven Comments on Charging and Billing", *Communications of the ACM*, vol. 43, no. 11, Nov. 2000, pp. 89-92.

[GIS01] S. Ghazizadeh, O. Ilghami and E. Sirin, "Security-Aware Adaptive Dynamic Source Routing Protocol", *Proceedings of IEEE Local Computer Networks*, Florida, Nov. 2002.

[GJP03] E. Gustafsson, A. Jonsson and C. Perkins, "Mobile IPv4 Regional Registration", Internet Draft, draft-ietf-mobileip-reg-tunnel-08, work in progress, Nov. 2003, http://www.ietf.org/internet-drafts/draft-ietf-mobileip-reg-tunnel-08.txt.

[GKFK01] V. Gazis, M. Koutsopoulou, C. Farmakis and A. Kaloxylos, "A Flexible Charging & Billing Approach for the Emerging UMTS Network Operator Role", *Proceedings. of ATS 2001*, Seattle, Washington, April 2001, http://www.cnl.di.uoa.gr/cnluk/cnlindex.php.

[GL02] S. Giordano and W. Lu (Editors). Special Issue on Challenges in Mobile Ad Hoc Networking. *IEEE Communications*, vol. 39, no. 6, June 2001.

[GMAG+95] S. Glassman, M. Manasse, M. Abadi, P. Gauthier and P. Sobalvarro, "The Millicent Protocol for Inexpensive Electronic Commerce", *Proceedings of the 4th International. World Wide Web Conference*, Boston, MA, Dec. 1995, pp. 603-618.

[Gre02] M. Greis, "Tutorial for the Network Simulator ns", June 2002, http://www.isi.edu/nsnam/ns/tutorial/index.html.

[GSMA] GSM Association, http://www.gsmworld.com/

[GSMA04a] GSM Association, "Membership and Market Statistics", Dec 2004, http://www.gsmworld.com/news/statistics/index.shtml.

[GSMA04b] GSM Association, "Tapping the Potential of Roaming", March 2004, http://www.gsmworld.com/using/billing/potential.shtml.

[HA03] R. Housley and W. Arbaugh, "Security Problems in 802.11-Based Networks", *Communications of the ACM*, vol. 46, no. 5, May 2003, pp. 31-34.

[Handhelds] handhelds.org, http://www.handhelds.org/geeklog/index.php

[Her03] A. Herzberg, "Payments and Banking with Mobile Personal Devices", *Communications of the ACM*, vol. 46, no. 5, May 2003, pp. 53-58.

[HC98] D. Harkins and D. Carrel, "The Internet Key Exchange Protocol (IKE)", IETF RFC 2409, Nov. 1998.

[HGBV01] J-P. Hubaux, T. Gross, J-Y Le Boudec and M. Vetterli, "Towards Self-Organized Mobile Ad Hoc Networks: The Terminodes Project", *IEEE Communications*, vol. 39, no. 1, Jan. 2001, pp. 118-124.

[HJP02] Y-C. Hu, D. B. Johnson and A. Perrig, "SEAD: Secure Efficient Distance Vector Routing in Mobile Wireless Ad Hoc Networks", *Proceedings of the 4th IEEE Workshop on Mobile Computing Systems and Applications (WMCSA'02)*, IEEE Press, 2002, pp.3-13.

[HL02] P. Henry and H. Lou, "WiFi: What's Next?", *IEEE Communications*, vol. 40., no. 12, Dec. 2002, pp. 66-72.

[HMP03] O. Haase, K. Murakami and T. LaPorta, "Unified Mobility Manager: Enabling Efficient SIP/UMTS Mobile Network Control", *IEEE Wireless Communications*, vol. 10, no. 4, Aug. 2003, pp. 66-75.

[HMT02] H. Haverinen, J. Mikkonen and T. Takamaki, "Cellular Access Control and Charging for Mobile Operator Wireless Local Area Networks", *IEEE Wireless Communications*, vol. 9, no. 6, Dec. 2002, pp. 52-60.

[HP04] Y-C Hu and A. Perrig, "A Survey of Secure Wireless Ad Hoc Routing", *IEEE Security & Privacy*, vol. 2, no. 3, June, 2004, pp. 28-39.

[HPJ02] Y-C. Hu, A. Perrig and D. B. Johnson, "Ariadne: A Secure On-demand Routing Protocol for Ad Hoc Networks", *Proceedings of MobiCom'02*, Atlanta, Georgia, Sep. 2002.

[HPNL02] H. Honlasalo, K. Pehkonen, M. Niemi and A. Leino, "WCDMA and WLAN for 3G and Beyond", *IEEE Wireless Communications*, April 2002, vol. 9, no. 2, pp. 14-18.

[HSW96] R. Hauser, M. Steiner, and M. Waidner, "Micro-payments Based on iKP", *Proceedings of the 14th Worldwide Congress on Computer and Communications Security Protection*, Paris, 1996, pp. 67-82.

[HY03] S. Hui and K. Yeung, "Challenges in the Migration to 4G Mobile Systems", *IEEE Communications*, vol. 41, no. 12, Dec. 2003, pp. 54-59.

[IEC] International Electrotechnical Commission, http://www.iec.ch/

[IEEE802] IEEE 802 Standards, http://standards.ieee.org/getieee802/

[IETF] Internet Engineering Task Force, http://www.ietf.org/

[IPsec] IP Security Protocol Working Group Charter, http://www.ietf.org/html.charters/ipsec-charter.html.

[ISC04] Internet Systems Consortium – Internet Domain Survey Host Count, Jan. 2004, http://www.isc.org/index.pl?/ops/ds/

[IST00] IST-2000-25350 SHAMAN, http://www.ist-shaman.org/

[ITFacts] IT Facts, http://www.itfacts.biz/

[ITU] International Telecommunications Union, http://www.itu.int/

[IWS04] Internet World Stats, http://www.internetworldstats.com/stats.htm.

[Jak02] M. Jakobsson, "Fractal Hash Sequence Representation and Traversal", *Proceedings of the IEEE International Symposium on Information Theory, ISIT'02*, Lausanne, Switzerland, June 2002, http://www.rsasecurity.com/rsalabs/staff/bios/mjakobsson/

[Jam03] A. Jamalipour, "*The Wireless Mobile Internet – Architectures, Protocols and Services*", John Wiley & Sons, NJ, USA, 2003.

[JL03] A. Jamalipour and P. Lorenz (Editors). Special Issue on Merging IP and Wireless Networks. *IEEE Wireless Communications*, vol. 10, no. 5, Oct. 2003.

[JMH03] D. Johnson, D. A. Maltz and Y-C. Hu, "The Dynamic Source Routing Protocol for Mobile Ad Hoc Networks", Internet Draft, draft-ietf-manet-dsr-08.txt, work in progress, Feb. 2003.

[Jon96] T. Jones, "The Future of Money as It Affects the Payment Systems in the U.S. and Abroad", *Submission to the U.S. House of Representatives*, June 1996.

[JPA03] D. Johnson, C. Perkins and J. Arkko, "Mobility Support in IPv6", Internet Draft, draft-ietf-mobileip-ipv6-24, work in progress, June 2003.

[JT01] A. Jamalipour and S. Tekinay (Editors). Special Issue on Fourth Generation Wireless Networks and Interconnecting Standards. *IEEE Personal Communications*, vol. 8, no. 5, Oct. 2001.

[KA98] S. Kent and R. Atkinson, "Security Architecture for the Internet Protocol", IETF RFC 2401, Nov. 1998.

[Kar04] S. Karnouskos, "Mobile Payment: A Journey Through Existing Procedures and Standardization Initiatives", *IEEE Communications Surveys & Tutorials*, vol. 6, no. 4, Fourth Quarter 2004.

[Kau04] C. Kaufman, "Internet Key Exchange (IKEv2) Protocol", Internet Draft, draft-ietf-ipsec-ikev2-17.text, Sep. 2004, http://www.ietf.org/internet-drafts/draft-ietf-ipsec-ikev2-17.txt.

[KFK96] P. Kocker, A. Freier and P. Karlton, "The SSL Protocol Version 3.0", Netscape Communications Corp., March 1996, http://home.netscape.com/eng/ssl3/index.html.

[KGDD02] T. Kwon, M. Gerla, S. K. Das and S. Das, "Mobility Management for VoIP Service: Mobile IP vs. SIP", *IEEE Wireless Communications*, vol. 9, no. 5, Oct. 2002, pp. 66-75.

[KH03] G. Koien and T. Haslestad, "Security Aspects of 3G-WLAN Interworking", *IEEE Communications*, vol. 41, no. 11, Nov. 2003, pp. 82-88.

[KJCH+03] Y. Kim, B. Jeong, J. Chung, C-S. Hwang, J. Tyu, Ki-Ho. Kim and Y. K. Kim, "Beyond 3G: Vision, Requirements, and Enabling Technologies", *IEEE Communications*, vol. 41, no. 3, March 2003, pp. 120-124.

[KKAM+04] M. Koutsopoulou, A. Kaloxylos, A. Alonistioti, L. Merakos and K. Kawamura, "Charging, Accounting and Billing Management Schemes in Mobile Telecommunications Networks and the Internet", *IEEE Communications Surveys & Tutorials*, vol. 6, no. 1, First Quarter 2004.

[Knu05] L. Kundsen, "Fast Software Encryption", http://www2.mat.dtu.dk/people/Lars.R.Knudsen/

[KS02a] H. Knospe and S. Schwiderski-Grosche, "Future Mobile Networks: Ad-hoc Access Based online Payment with Smartcards", *13th IEEE International Symposium on Personal, Indoor and Mobile Radio Communications, PIMRC'02*, pp. 197-200, Sep. 2002.

[KS02b] H. Knospe and S. Schwiderski-Grosche, "Online Payment Access to Hetrogeneous Mobile Networks", IST *Mobile and Wireless Telecommunications Summit 2002*, Thessaloniki, Greece, pp. 748-752.

[KS04] H. Knospe and S. Schwiderski-Grosche, "Secure Mobile Commerce", *In C. J. Mitchell, Editor, Security for Mobility, IEE Press*, pp. 325-346, Jan. 2004.

[Lam81] L. Lamport, "Password Authentication with Insecure Communication", *Communications of the ACM*, vol. 24, no. 11, Nov. 1981, pp. 770-772.

[LCR00] Y-B. Lin, M-F. Chang and H. Rao, "Mobile Prepaid Phone Service", *IEEE Personal Communications*, vol. 7, no. 3, June 2000, pp. 6-14.

[Lip05] H. Lipmaa, "Fast implementations", http://home.cyber.ee/helger/implementations/

[LJP03] H-Y Lach, C. Janneteau and A. Petrescu, "Network Mobility in Beyond-3G Systems", *IEEE Communications*, vol. 41, no. 7, July 2003, pp. 52-57.

[Llo03] D. Lloyd, "International Roaming Fraud Trends and Prevention Techniques", Fair Isaac Corporation, 2003, http://www.fairisaac.com/fairisaac/

[LPHC02] Yi-B Lin, A-C. Pang, Y-R Haung and I. Chlamtac, "An All-IP Approach for UMTS Third-Generation Mobile Networks", *IEEE Network*, vol. 16, no. 5, Sep. 2002, pp. 8-19.

[LPW03] B. Lamparter, K. Paul and D. Westhoff, "Charging Support for Ad Hoc Stub Networks", *Elsevier Journal of Computer Communications*, vol. 26, no. 13, Aug. 2003, pp. 1504-1514.

[LWW05] A. Lenstra, X. Wang and B. Weger, "Colliding X.509 Certificates", March 2005, http://www.win.tue.nl/~bdeweger/CollidingCertificates/CollidingCertificates.pdf.

[MAGMA] Multicast and Anycast Group Membership (MAGMA) Working Group Charter, http://www.ietf.org/html.charters/magma-charter.html.

[MAGM+03] V. Marques, R. Aguiar, C. Garcia, J. Moreno, C. Beaujean, E. Melin and M. Liebsch, "An IP-Based QoS Architecture for 4G Operator Scenarios", *IEEE Wireless Communications*, vol. 10, no. 3, June 2003, pp. 54-62.

[MANET] Mobile Ad Hoc Network (MANET) Working Group Charter, http://www.ietf.org/html.charters/manet-charter.html.

[Mep00] M. Peirce, "Multi-Party Electronic Payments for Mobile Communications", PhD Thesis, Trinity College Dublin, Ireland, Oct. 2000.

[Met99] C. Metz, "AAA Protocols: Authentication, Authorization and Accounting for the Internet", *IEEE Internet Computing*, vol. 3, no. 6, Dec. 1999, pp. 75-79.

[Met00] C. Metz, "IP over 2000: Where Have We Been and Where Are We Going?", *IEEE Internet Computing*, vol. 4, no. 1, Jan. 2000, pp. 83-87.

[MGLB00] S. Marti, T.J. Giuli, K. Lai and M. Baker, "Mitigating Route Misbehavior in Mobile Ad Hoc Networks", *Proceedings of ACM Mobicom '00*, Boston, Massachusetts, Aug. 2000, pp. 255-265, http://www.argreenhouse.com/mobicom2000/

[MH00] P. McCann and T. Hiller, "An Internet Infrastructure for Cellular CDMA Networks Using Mobile IP", *IEEE Personal Communications*, vol. 7, no. 4, Aug. 2000, pp. 26-32.

[MIP] IP Routing for Wireless/Mobile Hosts (mobileip) Working Group Charter, http://www.ietf.org/html.charters/mobileip-charter.html.

[MIP4] Mobility for IPv4 (mip4) Working Group Charter, http://www.ietf.org/html.charters/mip4-charter.html.

[MIP6] Mobility for IPv6 (mip6) Working Group Charter, http://www.ietf.org/html.charters/mip6-charter.html.

[MMDM04] N. Milanovic, M. Malek, A. Davidson, and V. Milutinovic, "Routing and Security in Mobile Ad Hoc Networks", *IEEE Computer*, vol. 37, no. 2, Feb. 2004, pp. 61-65.

[MNP04] A. Mishra, K. Nadkarni and A. Patcha, "Intrusion Detection in Wireless Ad Hoc Networks", *IEEE Wireless Communications*, vol. 11, no. 1, Feb. 2004, pp. 48-60.

[Mondex] Mondex USA, http://www.mondexusa.com/

[MOV96] A. Menezes, P. van Oorschot and S. Vanstone, *Handbook of Applied Cryptography*, CRC Press, Oct. 1996, http://www.cacr.math.uwaterloo.ca/hac/

[MP01] P. Mahonen and G. Polyzos (Editors). Special Issue on European R&D on Fourth-Generation Mobile and Wireless IP Networks. *IEEE Personal Communications*, vol. 8, no. 6, Dec. 2001.

[MPLS] Multiprotocol Label Switching (MPLS) Working Group Charter, http://www.ietf.org/html.charters/mpls-charter.html.

[MRPS04] P. Mahonen, J. Riihijarvi, M. Petrova and Z. Shelby, "Hop-by-Hop Toward a Future Mobile Broadband IP", *IEEE Communications*, vol. 42, no. 3, March 2004, pp. 138-146.

[MS95] P. Metzger and W. Simpson, "IP Authentication using Keyed MD5", IETF RFC 1828, Aug. 1995.

[MSST98] D. Maughan, M. Schertler, M. Schneider and J. Turner, "Internet Security Association and Key Management Protocol (ISAKMP)", IETF RFC 2408, Nov. 1998.

[Nam] The Network Animator (Nam), http://www.isi.edu/nsnam/nam/

[New04] P. Newman, "In Search of the All-IP Mobile Network", *IEEE Radio Communications*, vol. 42. no. 4, Dec. 2004, pp. s3-s8. (Quarterly Supplement to IEEE Communications Magazine)

[Nic03] S. Vaughan-Nichols, "The Challenge of Wi-Fi Roaming", *IEEE Computer*, vol. 36, no. 7, July 2003, pp. 17-19.

[NIST95] NIST Federal Information Processing Standards Publication (FIPS) 180-1: Secure Hash Standard (SHS). U.S. Department of Commerce, April 1995, http://csrc.nist.gov/

[NIST99] NIST Federal Information Processing Standards Publication (FIPS) 46-3: Data Encryption Standard (DES). U.S. Department of Commerce, Oct. 1999.

[NIST00a] Advanced Encryption Standard, NIST, http://csrc.nist.gov/CryptoToolkit/aes/

[NSIT00b] Digital Signature Standard, NIST, http://csrc.nist.gov/CryptoToolkit/tkdigsigs.html

[NIST05a] NIST Brief Comments on Recent Cryptanalytic Attacks on SHA-1, Feb. 2005, http://csrc.nist.gov/news-highlights/NIST-Brief-Comments-on-SHA1-attack.pdf

[NIST05b] National Institute of Standards and Technology (NIST), FIPS Approved Algorithms for Secure Hashing, http://csrc.nist.gov/CryptoToolkit/tkhash.html.

[NISTPKI] NIST Public Key Infrastructure Program, http://csrc.nist.gov/pki/

[NNS96] T. Narten, E. Nordmark and W. Simpson, "Neighbor Discovery for IP Version 6 (IPv6)", IETF RFC 1970, Aug. 1996.

[NS] The Network Simulator – ns -2, http://www.isi.edu/nsnam/ns/

[Orbis] Orbiscom, http://www.orbiscom.com/

[Orm98] H. Orman, "The OAKLEY Key Determination Protocol", IETF RFC 2412, Nov. 1998.

[Opp98] R. Oppliger, "Security at the Internet Layer", *IEEE Computer*, vol. 31, no. 9, Sep. 1998, pp. 43-47.

[OPT01] D. O'Mahony, M. Peirce and H. Tewari, *Electronic Payment Systems for E-Commerce*, 2nd Ed., Artech House Publishers, Boston/London, 2001.

[OSSL] The OpenSSL Project, http://www.openssl.org/

[Oye01] T. Oyedele, "Charging Requirements for UMTS Packet-Switched Data Services", Masters Thesis, Chalmers University of Technology, Sweden, Feb. 2001, http://db.s2.chalmers.se/download/masters/master_EX001_2001.pdf.

[Pan96] P. Panurach, "Money in Electronic Commerce: Digital Cash, Electronic Funds Transfer, and Ecash", *Communications of the ACM*, vol. 39, no. 6, June 1996, pp. 45-50.

[Par02] J-H. Park, "Wireless Internet Access for Mobile Subscribers Based on the GPRS/UMTS Network", *IEEE Communications*, vol. 40, no. 4, April 2002, pp. 38-49.

[PayBox] PayBox, http://www.paybox.net/

[PBSP01] A. Pras, B-J. Beijnum, R. Sprenkels and R. Parhonyi, "Internet Accounting", *IEEE Communications*, vol. 39, no. 5, May 2001, pp. 108-113.

[PC04] C. Perkins and P Calhoun, "AAA Registration Keys for Mobile IPv4", Internet Draft, darft-ietf-mip4-aaa-key-03.txt, work in progress, Feb. 2004.

[PCAG+04] C. Poltis, K Chew, N, Akhtar, M. Georgiades, R. Tafazolli and T. Daguiklas, "Hybrid Multilayer Mobility Management with AAA Context Transfer Capabilities for All-IP Networks", *IEEE Wireless Communications*, vol. 11, no. 4, Aug. 2004, pp. 76-88.

[PCCA04] A-C. Pang, J-C. Chen, Y-K Chen and P. Agrawal, "Mobility and Session Management: UMTS vs. CDMA 2000", *IEEE Wireless Communications*, vol. 11, no. 4, Aug. 2004, pp 30-43.

[PD00] G. Patel and S. Dennett, "The 3GPP and 3GPP2 Movements Toward and All-IP Mobile Network", *IEEE Personal Communications*, vol. 7, no. 4, Aug. 2000, pp. 62-64.

[PD03] J. Park and D. Dicoi, "WLAN Security: Current and Future", *IEEE Internet Computing*, vol. 7, no. 5, Oct. 2003, pp. 60-65.

[Per96a] C. Perkins ed., "IP Mobility Support", IETF RFC 2002, Oct. 1996.

[Per96b] C. Perkins, "IP Encapsulation within IP", IETF RFC 2003, Oct. 1996.

[Per97] C. Perkins, "Mobile IP", *IEEE Communications*, vol. 35, no. 5, May 1997, pp. 84-99.

[Per98] C. Perkins, "Mobile Networking Through Mobile IP", *IEEE Internet Computing*, vol. 2, no. 1, Jan. 1998, pp. 58-69.

[Per00] C. Perkins, "Mobile IP Joins Forces with AAA", *IEEE Personal Communications*, vol. 7, no. 4, Aug. 2000, pp. 59-61.

[Per02a] C. Perkins ed., "IP Mobility Support for IPv4", IETF RFC 3344, Aug. 2002.

[Per02b] C. Perkins, "Mobile IP", *IEEE Communications*, Anniversary Commemorative Issue, vol. 40, no. 5, May 2002, pp. 66-82.

[PH02a] P. Papadimitratos and Z. J. Haas, "Secure Routing in Mobile Ad Hoc Networks", *Proceedings of the Communication Networks and Distributed Systems Modeling and Simulation Conference (CNDS'02)*, San Antonio, Texas, Jan. 2002.

[PH02b] P. Papadimitratos and Z. J. Haas, "Performance Evaluation of Secure Routing for Mobile Ad Hoc Networks", *Proceedings of 1st ACM Workshop on Wireless Security*, Sep. 2002.

[PJ01] C. Perkins and D. Johnson, "Route Optimization in Mobile IP", Internet Draft, draft-ietf-mobileip-optim-11.txt, work in progress, Sep. 2001.

[PK00] R. Perlman and C. Kaufman, "Key Exchange in IPSec: Analysis of IKE", *IEEE Internet Computing*, vol. 4, no. 6, Dec. 2000, pp. 50-56.

[PK01] K. Pahlavan and P. Krishnamurthy, *Principles of Wireless Networks*, Prentice Hall, New Jersey, 2001.

[PKIX] Public-Key Infrastructure (X.509) (pkix) Working Group Charter, http://www.ietf.org/html.charters/pkix-charter.html.

[PO99] M. Peirce and D. O'Mahony, "Flexible Real-Time Payment Methods for Mobile Communications", *IEEE Personal Communications*, vol. 6, no. 6, Dec. 1999, pp. 44-55.

[PRD03] C. Perkins, E. Royer and S. Das, "Ad Hoc On Demand Distance Vector (AODV) Routing", IETF RFC 3561, July 2000, http://www.ietf.org/html.charters/manet-charter.html.

[PPSW97] A. Pfitzmann, B. Piftzmann, M. Schunter and M. Waidner, "Trusting Mobile User Devices and Security Modules", *IEEE Computer*, vol. 30, no. 2, Feb. 1997, pp. 61-68.

[PPYS+99] C. Polyzois, K. Purdy, P-F. Yang, D. Schrader, H. Sinnreich, F. Menard and H. Schulzrinne, "From POTS to PANS: A Commentary on the Evolution of Internet Telephony", *IEEE Internet Computing*, vol. 3, no. 3, June 1999, pp. 83-91.

[RB03] P. Reinbold and O. Bonaventure, "IP Micro-Mobility Protocols", *IEEE Communications Surveys*, vol. 5, no. 1, Third Quarter 2003, pp. 40-57.

[RHD03] A. Roos, M. Hartman and S. Duntnall, "Critical Issues for Roaming in 3G", *IEEE Wireless Communications*, vol. 10, no.1, Feb. 2003, pp. 29-35.

[RHKS02] C. Rensing, Hasan, M. Arsten and B. Stiller "AAA: A Survey and a Policy-Based Architecture and Framework", *IEEE Network*, vol. 16, no. 6, Dec. 2002, pp. 22-27.

[Rig97] C. Rigney, "RADIUS Accounting", IETF RFC 2139, April 1997, http://www.ietf.org/html.charters/OLD/radius-charter.html.

[Riv92a] R. Rivest, "The MD5 Message-Digest Algorithm", IETF RFC 1321, April 1992.

[Riv92b] R. Rivest, "The RC4 Encryption Algorithm", RSA Data Security Inc., March 1992.

[RS96] R. Rivest and A. Shamir, "PayWord and MicroMint: Two Simple Micropayment Schemes", *Proceedings of the 4th Security Protocols International Workshop (Security Protocols)*, LCNS, vol. 1189, Berlin: Spriger-Verlag, 1996, pp. 69-87.

[RPST+00] R. Ramjee, T. F. La Porta, L. Salgarelli, S. Thuel and K. Varadhan, "IP-Based Access Network Infrastructure for Next-Generation Wireless Communications Data Networks", *IEEE Personal Communications*, vol. 7, no. 4, Aug. 2000, pp. 34-41.

[RPTV+99] R. Ramjee, T. La Porta, S. Thuel, K. Varadhan and S.Y. Wang, "HAWAII: A Domain-based Approach for Supporting Mobility in Wide-area Wireless Networks", *Proceedings of the Seventh International Conference on Network Protocols ICNP'99*, Nov. 1999.

[RR02] R. Ramanathan and J. Redi, "A Brief Overview of Ad Hoc Networks: Challenges and Directions", *IEEE Communications*, 50th Anniversary Commemorative Issue, vol. 40, no. 5, May 2002, pp. 20-22.

[RSA78] R. Rivest, A. Shamir and L. Adleman, "A Method for Obtaining Digital Signatures and Public-Key Cryptosystems", *Communications of the ACM*, vol. 21, no. 2, 1978, pp. 120-126.

[RSALab] RSA Laboratories' Frequently Asked Questions About Today's Cryptography, Version 4.1, May 2000, http://www.rsasecurity.com/rsalabs/node.asp?id=2152

[RSCJ+02] J. Rosenberg, H. Schulzrinne, G. Camarillo, A. Johnston, J. Peterson, R. Sparks, M. Handley and E. Schooler, "SIP: Session Initiation Protocol", IETF RFC 3261, June 2002.

[RT99] E. Royer and C-K. Toh, "A Review of Current Routing Protocols for Ad Hoc Mobile Wireless Networks", *IEEE Personal Communications*, vol. 6, no. 2, April 1999, pp. 46-55.

[RWO95] S. Redl, M. Weber and M. Oliphant, *An Introduction to GSM*, Artech House Publishers, Boston/London, 1995.

[RWRS00] C. Rigney, S. Willens, A. Ruben and W. Simpson, "Remote Authentication Dial In User Service", IETF RFC 2865, June 2000, http://www.ietf.org/html.charters/OLD/radius-charter.html.

[Sal04] A. Salkintis, "Interworking Techniques and Architectures for WLAN/3G Integration Toward 4G Mobile Data Networks", *IEEE Wireless Communications*, vol. 11, no. 3, June, 2004, pp. 50-61.

[Sar00] B. Sarikaya, "Packet Mode in Wireless Networks: Overview of Transition to Third Generation", IEEE Communications, vol. 38, no. 9, Sep. 2000, pp. 164-172.

[SBGP+03] L. Salgarelli, M. Buddhikot, J. Garay, S. Patel and S. Miller, "Efficient Authentication and Key Distribution in Wireless IP Networks", *IEEE Wireless Communications*, vol. 10, no. 6, Dec. 2003, pp. 52-61.

[SBHJ03] N. Salem, L. Buttyan, J. Hubaux and M. Jaokobsson, "A Charging and Rewarding Scheme for Packet Forwarding in Multi-hop Cellular Networks", *Proceedings of the 4th Symposium on Mobile Ad Hoc Networking and Computing*, ACM Press, 2003, pp. 13-24.

[Sch96] B. Schneier, *Applied Cryptography – Protocols, Algorithms, and Source Code in C*, 2nd Ed., Wiley, New York, 1996.

[Sch05] B. Schneier, Schneier on Security, A Weblog Covering Security and Security Technology, http://www.schneier.com/blog/archives/2005/02/index.html.

[SDLS+02] K. Sanzigri, B. Dalhill, B. Levine, C. Shields and E. Royer, "A Secure Routing Protocol for Ad Hoc Networks", *Proceedings of IEEE ICNP '02*, Paris, France, Nov. 2002.

[Sen00] J. Senn, "The Emergence of M-Commerce", *IEEE Computer*, vol. 33, no. 12, Dec. 2000, pp. 148-150.

[SET97a] MasterCard and Visa Corporations, *Secure Electronic Transaction (SET) Specification – Book 1: Business Description Version 1.0*, May 1997.

[SET97b] MasterCard and Visa Corporations, *Secure Electronic Transaction (SET) Specification – Book 2: Programmers Guide Version 1.0*, May 1997.

[SET97c] MasterCard and Visa Corporations, *Secure Electronic Transaction (SET) Specification – Book 3: Formal Protocol Definition Version 1.0*, May 1997.

[SFP02] A. Salkintzis, C. Fors and R. Pazhyannur, "WLAN GPRS Integration for Next-Generation Mobile Data Networks", *IEEE Wireless Communications*, vol. 9, no, 5, Oct. 2002, pp. 112-124.

[SIP1] Session Initiation Protocol (SIP) Working Group Charter, http://www.ietf.org/html.charters/sip-charter.html.

[SIP2] Columbia Session Initiation Protocol (SIP), http://www.cs.columbia.edu/sip/

[SITI01a] "GPRS Billing: Getting Ready for UMTS", SITICOM Group White Paper, March 2001, http://www.siticom.fr/siti_gb/presse/revue_press_hi.htm.

[SITI01b] "Billing for 3G Access ... or Actual Service?", SITICOM Group White Paper, June 2001, http://www.siticom.fr/siti_gb/presse/revue_press_hi.htm.

[SK02] S. Schwiderski-Grosche and H. Knospe, "Secure Mobile Commerce", *In C. J. Mitchell (Editor): Special Issue of IEE Electronics and Communications Engineering Journal on Security and Mobility*, vol. 14, no. 5, pp. 228-238, Oct. 2002.

[SMAD00] S. Das, A. Misra, P. Agrawal and S. K. Das, "TeleMIP: Telecommunications-Enhanced Mobile IP Architecture for Fast Intradomain Mobility", *IEEE Personal Communications*, vol. 7, no. 4. Aug. 2000, pp. 50-58.

[SR99] H. Schulzrinne and J. Rosenberg, "The IETF Internet Telephony Architecture and Protocols", *IEEE Network*, vol. 13, no. 3, June 1999, pp. 18-23.

[SR00] H. Schulzrinne and J. Rosenberg, "The Session Initiation Protocol: Internet-Centric Signaling", *IEEE Communications*, vol. 38, no. 10, Oct. 2000, pp. 134-141.

[SS02] P Silva and H Sirisena, "A Mobility Management Protocol for IP-Based Cellular Networks", *IEEE Wireless Communications*, vol. 9, no. 3, June 2002, pp. 31-37.

[SSM04] M. Shi, X. Shen and J. Mark, "IEEE 802.11 Roaming and Authentication in Wireless LAN/Cellular Mobile Networks", *IEEE Wireless Communications*, vol. 11, no. 4, Aug. 2004, pp. 66-75.

[SMMC04] D. Saha, A. Mukherjee, I. Misra, M. Chakraborty, "Mobility Support in IP: A Survey of Related Protocols", *IEEE Network*, vol. 18, no. 6, Dec. 2004, pp. 34-40.

[SVP02] S. Salsano, L. Veltri and D. Papalilo, "SIP Security Issues: The SIP Authentication Procedure and its Processing Load", *IEEE Network*, vol. 16, no. 6, Dec. 2002, pp. 38-44.

[SW00] H. Schulzrinne and E. Wedlund, "Application-Layer Mobility Using SIP", *Mobile Computing and Communications Review*, vol. 4, no. 3, July 2000, pp. 29-36.

[Tan03] A. Tanenbaum, *Computer Networks*, 4th Ed., Prentice Hall PTR, New Jersey, 2003.

[TCL] Tcl/Tk Information Page, http://hegel.ittc.ukans.edu/topics/tcltk/

[TDG98] R. Thayer, N. Doraswamy and R. Glenn, "IP Security Document Roadmap", IETF RFC 2411, Nov. 1998.

[TN96] S. Thomson and T. Narten, "IPv6 Stateless Address Autoconfiguration", IETF RFC 1971, Aug. 1996.

[TO02] H. Tewari and D. O'Mahony, "Lightweight AAA for Cellular IP", *Proceedings of European Wireless '02*, Florence, Italy, Feb. 2002, pp. 301-306.

[TO03a] H. Tewari and D. O'Mahony, "Real-Time Payments for Mobile IP", *IEEE Communications*, vol. 41, no. 2, Feb. 2003, pp. 126-136.

[TO03b] H. Tewari and D. O'Mahony, "Multiparty Micropayments for Ad Hoc Networks", *Proceedings of IEEE WCNC'03*, New Orleans, Louisiana, March 2003, http://www.wcnc.org/2003/

[TSC03] Y-C. Tseng, C-C. Shen and W-T. Chen, "Integrating Mobile IP with Ad Hoc Networks", *IEEE Computer*, vol. 36, no. 5, May 2003, pp. 48-55.

[TTA] Telecommunications, Technology Association, South Korea, http://www.tta.or.kr/English/new/main/index.htm.

[TTC] Telecommunication Technology Committee, Japan, http://www.ttc.or.jp/e/

[UMTS03] "Mobile Evolution – Shaping the Future", UMTS Forum, White Paper, Aug. 2003, http://www.umts-forum.org/

[UK03] M. Ulema and B. Kozbe (Editors). Special Issue on Management of Next-Generation Wireless Networks and Services. *IEEE Communications*, vol. 41, no. 2, Feb. 2003.

[Usk03] S. Uskela, "Key Concepts for Evolution Toward Beyond 3G Networks", *IEEE Wireless Communications*, vol. 10, no. 1, Feb. 2003, pp. 43-48.

[Var02] U. Varshney, "Mobile Payments", *IEEE Computer*, vol. 35, no. 12, Dec. 2002, pp. 120-121.

[Var03] U. Varshney, "The Status and Future of 802.11-Based WLANs", *IEEE Computer*, vol. 36, no. 6, June 2003, pp. 102-105.

[Visa03] "Electronic Payments and Economic Growth", Visa International, White Paper, June 2003, http://www.corporate.visa.com/mc/documentdownloads/

[VINT] The Virtual InterNetwork Testbed, http://www.isi.edu/nsnam/vint/

[VLLX02] J. Vriendt, P. Laine, C. Lerouge and X. Xu, "Mobile Network Evolution: A Revolution on the Move", *IEEE Communications*, vol. 40, no. 4, April 2002, pp. 104-111.

[VVV03] K. Venken, I. Vinagre and J. Vriendt, "Analysis of the Evolution to an IP-Based UMTS Terrestrial Radio Access Network", *IEEE Wireless Communications*, vol. 10, no.5, Oct. 2003, pp. 46-53.

[WAGZ+03] D. Wisely, H. Aghvami, S. Gwyn, T. Zahariadis, J. Manner, V. Gazis, N. Houssos and N. Alonistioti, "Transparent IP Radio Access for Next-Generation Mobile Networks", *IEEE Wireless Communications*, vol. 10, no. 4, Aug. 2003, pp. 26-35.

[Wal00] J. Walker, "Unsafe at Any Key Size: An Analysis of the WEP Encapsulation", IEEE 802.11 Task Group E, Oct. 2000.

[WiFi] Wi-Fi Alliance, http://www.wi-fi.org/

[WinCE] Windows CE, http://msdn.microsoft.com/embedded/windowsce/default.aspx

[Wro97] J. Wroclawski, "The use of RSVP with IETF Integrated Services", IETF RFC 2210, Sep. 1997.

[WS99] E. Wedlund and H. Schulzrinne, "Mobility Support Using SIP", *Proceedings of the 2nd ACM/IEEE Intl Conference on Wireless and Mobile Multimedia, WoWMoM'99*, Seattle, Washington, Aug. 1999.

[WYY05] X. Wang, Y. Yin and H. Yu, "Collision Search Attacks on SHA1", http://theory.csail.mit.edu/~yiqun/shanote.pdf, Feb. 2005.

[YHH99] S. Yen, L. Ho and C. Huang, "Internet Micropayment Based on Unbalanced One-way Binary Tree", *Proceedings. of CrypTEC'99*, Hong Kong, July 1999, pp.155-162.

[YLYL+04] H. Yang, H. Luo, F. Ye, S. Lu and L. Zhang, "Security in Mobile Ad Hoc Networks: Challenges and Solutions", *IEEE Wireless Communications*, vol. 11, no. 1, Feb. 2004, pp. 38-47.

[ZA02] M. Zapata and N. Asokan, "Securing Ad Hoc Routing Protocols", *Proceedings of the ACM Workshop on Wireless Security (WiSe)*, ACM Press, 2002, pp. 1-10.

[ZAB99] M. Zeng, A. Annamalai and V. Bhargava, "Recent Advances in Cellular Communications", *IEEE Communications*, vol. 37, no. 9, Sep. 1999, pp. 128-138.

[ZAH04] T. Zahariadis (Editor). Special Issue on Migration Toward 4G Wireless Communications, *IEEE Wireless Communications*, vol. 11, no. 3, June 2004,

[ZCY03] S. Zhong, J. Chen and Y. Yang, "Sprite: A Simple, Cheat-Proof, Credit-Based System for Mobile Ad Hoc Networks", *Proceedings of IEEE Infocom'03*, San Francisco, CA, April 2003, pp. 1987-1997.

[ZH99] L. Zhou and Z. J. Haas, "Securing Ad Hoc Networks", *IEEE Network*, vol. 13, no. 6, Dec. 1999, pp. 24-30.

[ZK03] T. Zahariadis and D. Kazakos (Editors). Special Issue on (R)Evolution Toward 4G Mobile Communication Systems. *IEEE Wireless Communications*, vol. 10, no. 4, Aug. 2003.

[ZVTZ+02] T. Zahariadis, K. Vaxevanakis, C. Tsantilas, N. Zervos and N. Nikolaou, "Global Roaming in Next-Generation Networks", *IEEE Communications*, vol. 40, no. 2, Feb. 2002, pp. 145-151.