ABSTRACT

AD HOC NETWORK SECURITY AND MODELING WITH STOCHASTIC PETRI NETS

by
Congzhe Zhang

Advances in wireless technology and portable computing along with demands for high user mobility have provided a major promotion toward the development of ad hoc networks. These networks feature dynamic topology, self-organization, limited bandwidth and battery power of a node. Unlike the existing commercial wireless systems and fixed infrastructure networks, they do not rely on specialized routers for path discovery and traffic routing. Security is an important issue in such networks. Typically, mobile nodes are significantly more susceptible to physical attacks than their wired counterparts.

This research intends to investigate the ad hoc network routing security by proposing a performance enhanced Secure ad hoc On-demand Routing protocol (SOR). Specifically, it presents a method to embed "Security Level" into ad hoc on-demand routing protocols using node-disjoint multipath, and to use "maximum hopcount" to restrict the number of routing packets in a specific area. The proposed scheme enables the use of security as a marked factor to improve the relevance of the routes discovered by ad hoc routing protocols. It provides customizable security to the flow of routing protocol messages. In general, SOR offers an alternative way to implement security in on-demand routing protocols.

Ad hoc network is too complex to allow analytical study for explicit performance expressions. This research presents a Stochastic Petri net-based approach to modeling and analysis of mobile ad hoc network. This work illustrates how this model is built as a scalable model and used to exploit the characteristics of the networks. The proposed scheme is a powerful analytical model that can be used to derive network performance much more easily than a simulation-based approach. Furthermore, the proposed model is extended to study the performance of ad hoc network security by adding multipath selection and security measurement parameters. This research gives a quantificational measurement to analyze the performance of a modified SPN model under the effect of multipath and attack of a hypothetical compromised node.

AD HOC NETWORK SECURITY AND MODELING WITH STOCHASTIC PETRI NETS

by
Congzhe Zhang

A Dissertation
Submitted to the Faculty of
New Jersey Institute of Technology
In Partial Fulfillment of the Requirements for the Degree of
Doctor of Philosophy in Electrical Engineering

Department of Electrical and Computer Engineering

January 2004

APPROVAL PAGE

AD HOC NETWORK SECURITY AND MODELING WITH STOCHASTIC PETRI NETS

Congzhe Zhang

Dr. MengChu Zhou, Dissertation Advisor — Date
Professor of Electrical and Computer Engineering, NJIT

Dr. Constantine Manikopoulos, Committee Member — Date
Associate Professor of Electrical and Computer Engineering, NJIT

Dr. Hongya Ge, Committee Member — Date
Associate Professor of Electrical and Computer Engineering, NJIT

Dr. Symeon Papavassiliou, Committee Member — Date
Assistant Professor of Electrical and Computer Engineering, NJIT

Dr. Steven Chien, Committee Member — Date
Associate Professor of Civil and Environmental Engineering, NJIT

BIOGRAPHICAL SKETCH

Author: Congzhe Zhang

Degree: Doctor of Philosophy

Date: January 2004

Undergraduate and Graduate Education:

- Doctor of Philosophy in Electrical Engineering
  New Jersey Institute of Technology, Newark, NJ, 2004
- Master of Science in Automatic Control
  Tsinghua University, Beijing, P. R. China, 2000
- Bachelor of Science in Civil Engineering
  Tsinghua University, Beijing, P. R. China, 1997

Major: Electrical Engineering

Presentations and Publications:

C. Z. Zhang and M. C. Zhou, "A Stochastic Petri Net Approach to Modeling and Analysis of Ad Hoc Network," *submitted to IEEE Trans. on Systems, Man, and Cybernetics*: Part A, Oct. 2003.

C. Z. Zhang and M. C. Zhou, and S. Papavassiliou, "Ad Hoc Network Security: A Review," *submitted to Intl. Journal of Communication Systems*, October 2003.

C. Z. Zhang and M. C. Zhou, "Performance Enhanced Secure Ad Hoc On-Demand Routing Protocol," *submitted to IEEE Trans. on Systems, Man, and Cybernetics*: Part B, Nov. 2003.

C. Z. Zhang and M. C. Zhou, "Ad-Hoc Network Security: Review", *IEEE Intl. Conf. on Systems, Man, and Cybernetics*, October, 2003, Washington, D.C., USA.

C. Z. Zhang and M. C. Zhou, "A Stochastic Petri Net Approach to Modeling and Analysis of Ad Hoc Network", *Intl. Conf. on Information Technology: Research and Education*, August 10-13, 2003, Newark, New Jersey, USA

C. Z. Zhang and M. C. Zhou, "Security Enhanced Ad Hoc On-Demand Routing Protocol", *3rd Annual IEEE Information Assurance Workshop*, June 2002.

C. Z. Zhang, M. C. Zhou, and J. L. Gao, "Conversion Between Discrete Images and Organized 3D File Formats", *IEEE Intl. Conf. on Systems, Man, and Cybernetics*, vol. 5, pp. 2835-2839, 2001.

J. L. Gao, M. C. Zhou, H. M. Wang, and C. Z. Zhang, "Three Dimensional Surface Warping for Plastic Surgery Planning", *IEEE Intl. Conf. on Systems, Man, and Cybernetics*, vol. 3, pp. 2016-2021, 2001.

J. L. Gao, M. C. Zhou, and C. Z. Zhang, "2D and 3D Medical Image Database Design", *IEEE Intl. Conf. on Systems, Man, and Cybernetics*, vol. 3, pp. 2011-2015, 2001.

C. Z. Zhang and Y. Zhang, "Using Multi-variable Membership of Function to Classify the Quality of Tone of the Measurement Point of the Auditorium", *System Engineering Theory and Practice*, vol. 21, no. 11, 2001.

C. Z. Zhang and Y. Zhang, "A Generalized Ray-Tracing Algorithm for Integrated Visual and Auditory Rendering in Room Acoustics", *Civil Engineering and Environmental Systems*, vol. 18, no. 1, pp. 19-34, 2001.

C. Z. Zhang and Y. Zhang, "Visual and Auditory Rendering Integrated Ray-Tracing Algorithm", *Chinese Science Abstracts*, vol.6, no.5, pp. 636-637, 2000.

C. Z. Zhang, Y. Zhang, and C. D. Geng, "Virtual Reality Technology in CAD", *Computer World*, vol. 5, Feb. 1998.

C. Z. Zhang and Y. Zhang, "Virtual Reality Technology in Architectural Robot", *Computer World*, vol. 5, Feb. 1998.

Y. Zhang and C. Z. Zhang, "Virtual Reality Technology in Civil Engineering", *Theory and Practice in Multimedia Aided Engineering*, Publishing House of Electronics Industry, pp. 104-133, 1998.

To Remember My Beloved Mother

TABLE OF CONTENTS

TABLE OF CONTENTS
(Continued)

Chapter **Page**

TABLE OF CONTENTS
(Continued)

LIST OF TABLES

LIST OF FIGURES

LIST OF FIGURES
(Continued)

Figure **Page**

CHAPTER 1

INTRODUCTION

1.1 Motivation

In the last few years, there was a surge of interest in mobile ad hoc networks. The readers might have already been familiar with present Internet vision of mobile wireless networks, such as the cellular networks and wireless Local Area Networks (LANs), in which wireless nodes on the edge of the network cloud are typically connected and supported by a single wireless hop to the fixed, wired infrastructure. An ad hoc network expands this vision. Basically, an ad hoc network is a collection of wireless mobile nodes dynamically forming a network without the use of any existing network infrastructure or centralized administration. It can be formed, merged or partitioned into separate networks, without necessarily relying on a fixed infrastructure to manage their operation. Hence, there is no fixed infrastructure such as base stations in a cellular network. Each node is capable of moving independently and functioning as a router that discovers and maintains routes and forwards packets to other nodes. The topology of interconnections may be quite dynamic. The rapidly deployable, self-organizing nature is the primary factor that differentiates ad hoc network from the commercial cellular networks.

In contrast to traditional wire line or wireless network, the ad hoc network technology is still at its early stage. Much research efforts are being made to address the issues that primarily differentiate it from its infrastructure-oriented cousin. Typical differences and operational characteristics for ad hoc networks include distributed operation, dynamic network topology, fluctuating link capacity, and low-power devices.

Moreover, Quality of Service (QoS), routing, security, mobility, and scalability issues are also major concerns of ad hoc networks and areas requiring in depth research.

Communications among nodes in an ad hoc network come with the support from routing protocols. Routing protocol security is an important issue in ad hoc networks. After a route is successfully found, the information to be sent over it must be protected against malicious attacks. Since they use the same wireless transmission medium, both data and control messages need to be well protected. So far many protocols have been proposed and their performance has been well researched. But the security issues and concerns have not been addressed in depth. In general, there are two sources of threats: external and internal attacks. Typically, an attacker can eavesdrop routing information and inject erroneous routing information, replay old routing information, or distort routing information. An attacker can successfully partition a network or introduce excessive traffic load into the network by causing retransmission and inefficient routing. More severe kind of threat comes from compromised nodes within the internal network. A compromised node can broadcast incorrect routing information to other nodes. Detection of such nodes through routing information is difficult in an ad hoc network because of its dynamically changing topology: an invalid piece of routing information can be generated by a compromised node, or because of the result of topology changes. It is difficult to distinguish between them. On the other hand, false routing information generated by compromised nodes can be considered the same as the outdated information. As long as there are sufficiently many correct nodes, the routing protocol should be able to find new routes that pass by these compromised nodes.

Ad hoc network is too complex to allow analytical study for explicit performance expressions. One of the widely used approaches is simulation methods. However, there are two main drawbacks to using simulation: first, it may be time consuming to execute the necessary simulations. Imaging in a highly variable scenario, with number of nodes ranging from tens to thousands, node mobility varying from zero to tens of m/s, the simulation time of most current systems will increase dramatically to an unacceptable level. Second, it may be difficult to achieve results that are precise enough. In order to get a reliable value, one has to run simulation tens of iterations with different seed values of a random generator.

Due to the advantage in quick construction and numerical analysis of Petri nets, it has been used for the network performance analysis. Yet ad hoc network has not been explored because of its specific characteristics different from other networks. To present an approach for the modeling and analysis of large-scale ad hoc network systems using Petri nets, there are two requirements in advance. First, a model should be detailed enough to describe some important network characteristics that have a significant impact on performance. Second, it should be simple enough to be scalable and analyzable.

1.2 Objectives

This research intends to enhance the ad hoc network security and investigate the overall network security performance using Petri net approaches. The specific objects are:

1) To investigate the "Security Level" concept to enhance the ad hoc network security by using a multi-path scheme.

2) Based on an ad hoc on-demand routing protocol and "Security Level" concept, to propose a performance enhanced routing protocol embedding a "maximum hopcount" factor to restrict the number of routing packets for a particular purpose.
3) To use Stochastic Petri Nets (SPN) to analyze the data transmission and develop a SPN model for network performance analysis as a function of various parameters.
4) To investigate the ad hoc network security using a Petri net approach, and find a better way to describe the characteristics of security.

1.3 Organization

The dissertation is organized as follows: Chapter 1 gives the motivation and objectives of the research work. Chapter 2 makes the literature review of the current research issues and existing methods on the related research subjects. Chapter 3 proposes a performance enhanced secure ad hoc on-demand routing protocol addressing on the ad hoc network security. Chapter 4 develops a stochastic Petri net model for ad hoc network and makes the network performance analysis based on this model. Chapter 5 proposes a method embedding security and multi-path aspects into the Petri net model. Finally, Chapter 6 presents the conclusions and some future research directions.

CHAPTER 2

LITERATURE REVIEW

2.1 Ad Hoc Network

Advances in wireless technology and portable computing along with demands for further user mobility have promoted the development of ad hoc networks. Such networks are characterized by dynamic topology due to node mobility, and self-organizing. Their nodes have only limited bandwidth and limited battery power. Unlike the existing commercial wireless systems and fixed infrastructure networks, ad hoc networks reply on no specialized routers for path discovery and traffic routing. In such a network, each mobile node operates not only as a host but also as a router, sending and forwarding packets to other mobile nodes in the network that may not be within its direct wireless transmission range. Every node in a network complies with an ad hoc routing protocol that allows it to discover "multi-hop" paths, which means a packet from a source node to a destination node can go through several nodes throughout the network.

Ad hoc networks have largely evolved from the packet radio network program of the U.S. Defense Advanced Research Projects Agency (DARPA) [Jubin and Tornow, 1987]. They intend to play an important role in military and commercial settings where mobile access to a wired network is either ineffective or impossible. In the early 1990s, a series of new developments signaled a new phase in ad hoc networking. The Department of Defense funded the Near-term Digital Radio program [Ruppe et al., 1997], which developed a two-tier ad hoc network architecture. The clustering and link-state routing was used in it. Meanwhile, within the Internet Engineering Task Force (IETF), the

Mobile Ad Hoc Networking (MANET) working group was born and began to standardize routing protocols for ad hoc networks [MANET, 2003].

Ad hoc networks have their own characteristics that are significantly different from those of fixed networks:

1) Dynamic topologies: Node mobility in an ad hoc network causes frequent changes of network topology. Adjusting transmission and reception parameters such as power can also affect the topology. Thus, the node needs to collect connectivity information from other nodes periodically. An implication of this is an increased message overhead in collecting topology information. In general, strategies designed to support internetworking in ad hoc networks should handle such topological changes with minimal overhead by limiting the scope of control packets that may have to be generated and propagated after a change in topology.

2) Bandwidth-constrained, variable capacity, possibly asymmetric links: wireless links have significantly lower capacity than their hardwired counterparts. One effect of these relatively low to moderate link capacities is that congestion happens more often than wired networks. This also makes Quality of Service (QoS) harder to implement for some applications, which require support of a certain QoS for optimal performance in a wireless environment.

3) Energy-constrained operation: Most of the ad hoc nodes run on batteries. That is, network overhead needs to be kept at the minimum level so that energy is conserved. Moreover, in order to conserve energy, nodes may power themselves off. This requirement is contradictory to the need for topology update messages.

4) Wireless vulnerabilities and limited physical security: Mobile wireless networks are generally more prone to information and physical security threats than fixed, hardwired networks. Nodes roaming in a hostile environment such as a battlefield have non-negligible probability of being compromised. That is, a malicious attack may be launched from within the network by compromised nodes.

For the ad hoc network security, the following attributes: availability, confidentiality, integrity, authentication, and non-repudiation need to be considered [Zhou and Haas, 1999; Kärpijoki, 2001].

Availability ensures the survivability of network services despite denial-of-service attacks. An attack can be launched at any layer of an ad hoc network. An adversary could employ jamming to interfere with communication on the physical and media access control layers; disrupt the routing protocol and disconnect the network on the network layer; bring down high-level services on the higher layer. One useful measure is the key management service [Fumy and Landrock, 1993], not only for ad hoc networks, but also for traditional networks.

Confidentiality ensures that certain information is never disclosed to unauthorized entities. Such information includes network transmission of sensitive information and routing information. Due to the inherent characteristic of ad hoc networks, each node acts as a router, and sensitive information needs multi-hop paths through network to other nodes, thereby enlarging the possibility of leaking routing information.

Integrity requires that messages should not be altered or corrupted during transmission. A message could be altered by a benign or malicious attack on the network.

Authentication means that the participants somehow prove that their identities are what they claim them to be. Authentication can be done by something users know, embody or possess. For instance, something known can be a password, something embodied can be a fingerprint, and something possessed can be a smart card.

Non-repudiation guarantees that the origin of a message cannot deny having sent the message and the receiver cannot deny the reception. Non-repudiation is useful for detection and isolation of compromised nodes. For instance, when node A receives an erroneous message from node B, non-repudiation allows A to accuse B using this message and to convince other nodes that B is compromised.

Based on the above-mentioned requirements, several traditional security mechanisms still play important roles in achieving the above attributes. However, these mechanisms are not as same as before [Mäki, 2000]. Changes need to be added to these mechanisms. This research focuses on routing protocols and encryptions.

Encryption can be used to hide the information during transmission or to store information more safely. It is assumed to change the information in such a way that only authorized users can interpret it. Therefore, encryption is used to gain confidentiality.

Protocols. Encryption alone does not accomplish security. It works as a part of the security protocol used in a network. The protocol defines the steps how, for example, the parties authenticate each other, and what infrastructure is needed for the authentication. Protocols involve key management, and they may often require the use of certificates.

2.2 Routing Protocols and Encryptions

2.2.1 Encryption

Cryptosystems can be divided into symmetric and asymmetric ones. In a symmetric cryptosystem the encryption and decryption keys are identical. In asymmetric cryptosystems, or public key systems, they are different from each other, and the decryption key should not be derivable from the one used for encryption. The encryption key of the parties is public, while the decryption keys are personal secrets of the participants. For instance, in an encryption scheme, one party, *the sender*, generates a ciphertext for a message with another party's public key. The other party, called *the recipient*, decrypts the ciphertext with the corresponding private key to obtain the original message. This encryption scheme can provide confidentiality of a message.

Beritelli et al. (2000) proposed a multiplayer chaotic encryption system based on chaotic models with a dynamic key. The algorithm is symmetric with a dynamic chaotic key. The key used for a masking sequence is made up of two factors: state and control parameter. The combination of these two elements allows an encrypted message to be decrypted. The system dynamically updates both the state and control parameter of the chaotic series used for masking. The laws governing the update of the initial state and control parameter are also of the chaotic type. As its extension to a more general stage, a multiplayer key generation structure is proposed based on the most appropriate Chaotic Map (CM). The initial state and control parameter of every stage are generated in the previous stage following a chaotic law.

Asymmetric cryptosystem gains wider use than symmetric one on the Internet due to an increasing security requirement. Sander and Tschudin (1998) proposed a

homomorphic public-key encryption scheme that allows for non-interactive addition or multiplication of two encrypted messages by manipulating ciphertext only. In this way, the host can compute any function $g(\cdot, y)$ on a hidden input x that is represented by a polynomial. This approach was later improved by Sander et al. (1999) to non-interactive evaluation of all functions $g(\cdot, y)$ on a hidden input x that can be represented by circuits of logarithmic depth. However, this method is considered infeasible by Algesheimer et al. (2001) using a privacy of a mobile shopping agent application example. They indicate that any scheme in which some host is to learn information that depends on the agent's current state cannot be secure.

Algesheimer et al. (2001) proposed a *generic secure computation service*. It performs some cryptographic operations on behalf of a mobile code. Here, the mobile code is a program which is produced by one entity and subsequently transferred to a second entity through the network. This secure computation service resembles other generic security services like a public-key infrastructure (PKI) or an anonymous re-mailer. Their basic idea is based on [Yao, 1986]: original sender O constructs an encrypted circuit C computing two outputs ξ and z. It sends C to the host H, encrypts all keys in "key pairs" K for a third party T and does not include the key pairs in "key pairs" U which correspond to ξ (denoted by U) so that H will not learn anything about ξ. Next H selects from K the encrypted keys representing input bit y and invokes T to decrypt them in a single round of interaction. Then H evaluates the circuit and obtains z; it also returns the keys in the circuit output representing ξ to O, who can determine ξ from this.

Taking the advantage of the inherent redundancy in ad hoc network - multiple routes between nodes, Zhou and Haas (1999) proposed a distributed, asynchronous key-

management service to defend routing against denial-of-service attacks. Because nodes are possibly compromised, the network should not have any central entities but a distributed architecture. The authors present a distributed key management system where the private key of a trusted service is divided to *n* servers. To create a signature with the private key, at least *k* out of the *n* servers need to combine their knowledge. Combining the shares would not reveal the actual private key. The correctness of the signature would, as usual, be verifiable with the public key of the service. The method is called *threshold cryptography* [Desmedt, 1994]: an *(n, k)* threshold cryptography scheme allows *n* parties to share the ability to perform a cryptographic operation (e.g., creating a digital signature). Any *k* parties can perform the operation jointly, whereas it is infeasible for a less number of parties than *k*. If at most *k-1* servers can be compromised at a time, a false signature cannot be created.

2.2.2 Routing Protocol

Many different protocols have been proposed to solve the multi-hop routing problem in ad hoc networks based on different assumptions and intuitions. An ad hoc network routing must be simple, robust, and should minimize the control message exchanges. Ad hoc routing needs to be simple because it is performed by generic mobile hosts that have limited resources and power. Routing algorithms that consume excessive bandwidth for routing control message exchanges may not be appropriate for wireless networks. The topology of an ad hoc network is inherently volatile and routing algorithms must be robust against frequent topology changes caused by host movements [Lee and Kim, 2000].

Ad hoc routing can be classified into proactive and reactive ones based on when routes are determined. The former continuously makes routing decisions so that routes are immediately available when packets need to be transmitted with no regard to when and how frequently such routes are desired. It relies on an underlying routing table update mechanism that involves the constant propagation of routing information. Its example protocols are Destination-Sequenced Distance Vector (DSDV) [Perkins and Bhagwat, 1994], and Wireless Routing Protocol (WRP) [Murthy and Garcia-Luna-Aceves, 1996]. The reactive routing determines routes on an on-demand basis: when a node has a packet to transmit, it queries the network for a route. Its example protocols are Temporally-Ordered Routing Algorithm (TORA) [Park and Corson, 1997], Ad hoc On-Demand Distance Vector (AODV) [Perkins and Royer, 1999], and Dynamic Source Routing (DSR) [Johnson and Maltz, 1996]. Each has its disadvantages: proactive routing consumes a great deal of resources to exchange routing information while reactive routing may rapidly lose their validity in an ad hoc network because its topology changes rapidly. Multipath on-demand protocols try to alleviate these problems by computing multiple paths in a single path discovery attempt. Multiple paths can be used to balance load by forwarding data packets on multiple paths at the same time or as a backup to route discovery, i.e., new route discovery is needed only when all possible paths fail. The former one will be discussed in the following chapters.

Recently, the use of geographic positions as a means of routing has become increasingly popular in mobile ad-hoc networks. One important advantage of using positions for routing is its inherent ability to alleviate the need for the development of separate complicated techniques for mobility management. In addition, position-based

routing strategies, since they do not require the exchanges of routing tables, are especially attractive in highly mobile environments where topological changes are frequent and routing tables become obsolete very quickly [Hou and Li, 1985]. In general, using geographic positions as a means of routing/mobility management has the following advantages:

1) It makes feasible the realization of a flat routing architecture, thus, eliminating the complexities associated with maintaining rigid, multiple-tier hierarchical architectures, therefore making the network robust and 'fluid' in nature,
2) It minimizes the amount of initial manual configuration required to set-up the network, thus making the network completely self-organizing, and
3) It makes feasible the process of routing packets with geographic destinations.

Because some of the protocols' characteristics will be utilized later, a brief review of some important routing protocols is presented below.

Destination-Sequenced Distance Vector (DSDV)

DSDV [Perkins and Bhagwat, 1994] is a multihop distance vector routing protocol requiring each node to broadcast routing updates periodically. It is based on the idea of the classical Bellman-Ford routing algorithm. Each node in the system maintains a routing table containing the next-hop information for each reachable destination. Each route has a sequence number. If a new entry is given, it prefers the route with the greatest sequence number, or if different routes' sequence numbers are the same, it chooses the metric with the lowest value. Each node advertises an increasing even sequence number for itself. When the source node determines that the destination node is unreachable, it advertises the next odd sequence for the route that has failed with an infinite metric count

number. Any node receiving this infinite metric count updates its table for the matching route and waits until a greater sequence number with non-infinite metric count is received. Its advantage over traditional distance vector protocols is that it guarantees loop-freedom [Perkins et al., 2001].

Wireless Routing Protocol (WRP)

WRP [Murthy and Aceves, 1996] is a vector routing protocol. Each node in the network maintains a table containing routing, distance, link cost information and a message retransmission list. Distance table of a source node S contains the distance of each destination node D via each neighbor N of S. It also contains the downstream neighbor of N through which this path is realized. The routing table of S contains the distance of each destination node D from node S, and the predecessor and successor of S on this path. It also contains a tag to identify if the entry is a simple path, a loop or invalid. Storing predecessor and successor into the table helps in detecting loops and the counting-to-infinity problem. Link cost table contains cost of link to each neighbor of the node and the number of timeouts since last error-free message was received from that neighbor. Message transmission list contains information about which neighbor has not acknowledged its update message, allowing a node to retransmit the update message to that neighbor.

Nodes exchange routing tables with their neighbors periodically or when links change. Nodes in a message transmission list must acknowledge back to update messages. If there is no change in a routing table since the last update, a node has to send an idle hello message to ensure connectivity. On receiving an updated message, a node modifies its distance table and looks for better paths using this new information. Any newly found

path is sent back to original nodes so that they can update their tables. A node also updates its routing table if a new path is better than the existing one. When a node receives ACK, it updates its message transmission list. This algorithm checks for consistency of all its neighbors every time it detects a change to any of its neighbors. It eliminates looping and converges fast [Celebi, 2001].

Dynamic Source Routing (DSR)

DSR [Johnson and Maltz, 1996] uses source routing rather than hop-by-hop routing, with each packet to be routed carrying in its header the complete, ordered list of nodes through which the packet must pass. The DSR protocol consists of two mechanisms: Route Discovery and Route Maintenance. To perform the former, the source node S broadcasts a route request packet that is flooded through the network in a controlled manner and answered by a route reply packet from either the destination node or another that knows a route to the destination. To reduce the cost of the former, each node maintains a cache of source routes it has learned or overheard. This node uses cache to limit the frequency and propagation of route requests. Route Maintenance is the mechanism by which a packet's sender S detects if the network topology has changed such that it can no longer use its route to the destination D because two nodes listed in the route have moved out of range of each other. When Route Maintenance indicates a source route is broken, S is notified with a route error packet. S can then attempt to use any other route to D already in its cache or can invoke Route Discovery again to find a new route. DSR's key advantage is that intermediate nodes do not need to maintain up-to-date routing information in order to route the packets they forward, since the packets themselves already contain all the routing decisions. This feature, coupled with the on-

demand nature of the protocol, eliminates the need for the periodic route advertisement and neighbor detection packets present in other protocols.

Ad hoc On-Demand Distance Vector (AODV)

AODV [Perkins and Royer, 1999] is based on a hop-by-hop routing approach. When a source node needs a route to a destination, it broadcasts a Route Request (RREQ) message to its neighbors. Each node receiving the message creates a reverse route to the source. This message is flooded until the information required is complete by either meeting the destination or meeting a node that has a known route to the destination. Route Reply (RREP) message is sent back to the source. Duplicate copies of the RREQ packet received at any node are discarded. Each node receiving the reply message creates a forward route to the destination. Thus each node remembers only the next hop required to reach any of the hosts, not the whole route. Once the source node receives RREP, it may begin to forward packets to the destination. If the source later receives a RREP message containing a shorter path, it may update its routing information for that destination and begin using the shorter route.

Each RREQ carries a parameter named hopcount that represents the number of hops from the source node to the node handling the request. Any node that wants to forward RREQ will increase the hopcount by one. After some nodes have successfully found a route to the destination node, the flood is still ongoing throughout the whole network until the predetermined time is out.

Position-Based Routing

Position information can be used in conjunction with 'on-demand' routing protocols in order to limit the scope of the flood search that occurs when a source

attempts to find a route to a destination. Location-Aided Routing (LAR) [Ko and Vaidya, 1998] couples DSR with position knowledge, by using the last known destination position as the origin of an uncertainty circle, which is called the 'request zone', and limiting the flood search for a destination route within that request zone. In [Basagni et al., 1999], global node position knowledge is used as a means of constructing a network connectivity graph, which is subsequently used to construct a source route from a given source to a given destination; thus, there is no need to perform flood search as in DSR. However, it should be noted that the connectivity graph constructed using exclusively the position knowledge might not necessarily correspond to the actual network connectivity due to the existence of terrain obstructions. In addition, the drawback associated with the source routing in general, is that the route computed originally at the source may become obsolete as the packet is 'in transit', resulting in packet loss.

Distance Routing Effect Algorithm for Mobility (DREAM) [Basagni et al., 1998] and Fisheye State Routing (FSR) [Pei et al., 2000] are two protocols that are based on the existence of multiple routing events, and the exploitation of the 'near-far' routing effect. DREAM is a position-based routing protocol that utilizes an array of position-update triggering timers, where timer[k] has a larger period (expiration interval) than timer[k-1]; therefore, position updates triggered by the expiration of timer[k] have a larger dissemination radius than the updates triggered by the expiration of timer[k-1]. FSR is a link-state routing protocol that utilizes the same update triggering/dissemination mechanism, but instead of position updates, it sends link-state updates.

In [Xu et al., 2000; Xu et al., 2000], the Position-guided Sliding-window Routing (PSR) was proposed and evaluated. PSR is also a position-based routing protocol that

exploits the 'near-far' routing effect. It, however, differs from DREAM and FSR in that it actively combines link-state routing with position-based routing. The link-state component of PSR provides for QoS routing and proactive bandwidth management within the local vicinity of a node (referred to as the 'core' zone), while the position-based component provides the position-update dissemination control mechanism, which is necessary for scalability.

Since Broch et al. (1998) first introduced a quantitative analysis method comparing the performance of a variety of multi-hop ad hoc network routing protocols, much related work [Boukerche, 2001; Perkins et al., 2001; Celebi, 2001; Cano and Manzoni, 2000; Ahuja et al., 2000; Royer and Toh, 1999] has been done. Royer and Toh (1999) analyzed the time complexity, communication complexity, and other parameters of eight different ah hoc network protocols, while Broch et al. (1998) quantized routing overhead, packet delivery ratio, and other parameters using a modified discrete event simulator - ns [Fall and Varadha, 1997]. Each protocol has its own advantages and disadvantages. None of them can be claimed as absolutely better than the others. For instance, Broch et al. (1998) found that DSDV delivered virtually all data packets when node mobility rate and movement speed were low, but failed to converge as node mobility increased. TORA delivered over 90% of the packets in scenarios when overhead was low, but when overhead was high, it was unable to handle all of the traffic generated by the routing protocol and a significant fraction of data packets were dropped. Perkins et al. (2001) found that the attribute of aggressive caching helped DSR at low loads and also kept its routing load down. The overhead of DSR is potentially larger than that of AODV since each DSR packet has to carry full routing information, whereas in AODV packets

need contain only the destination address. Combining the related work [Boukerche, 2001; Perkins et al., 2001; Celebi, 2001; Broch et al., 1998; Royer and Toh, 1999], Table 2.1 summarizes the comparison results of four typical routing protocols in terms of their time complexity, communication complexity, packet delivery ratio, average packet delay, routing load, power consumption, multicast capability and security.

Table 2.1 Comparison of Four Routing Protocols

	DSDV	TORA	DSR	AODV
Time complexity	Better	Good	Good	Good
Communication complexity	Better	Good	Good	Good
Packet delivery ratio	Good	Worse	Good	Better
Average packet delay	Better	Worse	Good	Better
Routing load	Good	Worse	Good	Good
Power consumption	Better	Good	Worse	Good
Multicast capability*	N/C	N/C	N/C	Yes
Security	N/C	Good potential	To be improved	To be improved

*: N/C – Not considered

2.2.3 Protocol Security

Routing protocols should be robust against both dynamically changing topology and malicious attacks. There is no standard protocol for ad hoc networks because ad hoc network itself is under research.

There are two sources of threats to routing protocols: external and internal attacks. Typically, an attacker can eavesdrop routing information and inject erroneous routing

information, replay old routing information, or distort routing information. An attacker can successfully partition a network or introduce excessive traffic load into the network by causing retransmission and inefficient routing.

More severe kind of threat comes from compromised nodes within the internal network. A compromised node can broadcast incorrect routing information to other nodes. Detection of such nodes through routing information is difficult in an ad hoc network because of its dynamically changing topology. An invalid piece of routing information can be generated by a compromised node, or as the result of topology changes. It is inherently difficult to distinguish between them. On the other hand, false routing information generated by compromised nodes can be considered the same as the outdated information. As long as there are sufficiently many correct nodes, the routing protocol should be able to find routes that bypass these compromised nodes. Therefore, the protocols that are capable of finding multiple paths such as AODV, TORA, and DSR, have an advantage.

Diversity coding [Ayanoglu, 1993] uses multiple paths to transmit data, and does not make any retransmission. Its basic idea is to transmit redundant information through additional routes for error detection and correction. For example, if there are n disjoint routes between two nodes, $n - r$ channels can be used to transmit data and other r channels used to transmit redundant information. Even if certain routes are compromised, the receiver may still be able to validate messages and recover them from errors using the redundant information from the additional r channels. This method is developed for static network, and applicable for ad hoc network.

Smith et al. (1997) studied vulnerabilities and provided countermeasures for distance-vector routing protocols. They managed to do it by using the predecessor information specified in the path-finding class of distance-vector proposals. The same method can be adopted for securing Mobile Ad-Hoc Network distance-vector protocols. As a result, routing messages should be protected as well as the user data, and multiple path transmission should always be used.

Network management is a process of controlling a complex data network so as to maximize its efficiency and productivity [Leinwand and Fang, 1993; Xu et al., 2000]. It can be functionally divided into five areas defined by the International Standards Organization (ISO): fault management, configuration management, security management, performance management, and accounting management. Simple network management protocol (SNMP) [Stallings, 1998] is the most widely deployed management protocol standard for the management of IP-based networks and Internets. SNMPv1 defined a protocol for exchanging information between one or more management systems and a number of agents, provides a framework for formatting and storing management information, and defines a number of general-purpose management information variables or objects.

The current typical protocols in ad hoc network do not pay much attention to the security issue yet. Future improvements about these protocols need to focus on the security aspect, as done in current standard protocols of Internet such as Secure Sockets Layer [Netscape, 2001], and authentication header and encryption technology used in IPv6 [2001]. Some detailed measures include neighbor-to-neighbor digital signature of routing updates, the addition of sequence numbers and timestamps to the updates, the

addition of acknowledgments and retransmission of routing updates [Kumar, 1993], cryptographic protection of the hop information in a path, and IPSec [IPSEC, 2001] authentication headers deployed along with the necessary key management to distribute keys to the members of the ad hoc network.

In general, network attacks range from passive information steal to active message impersonation and distortion. The security provisioning mechanism can be improved based on multipath traffic dispersion against network attacks along the information path. The approach proposed in [Yang and Papavassiliou, 2001] can be used to improve network security so that the possibility of unauthorized information leakage induced by attacks along the path is minimized. Moreover a heuristic multipath traffic dispersion scheme has been proposed in order to minimize the impact of network attacks. This approach seems applicable for the mobile ad hoc networks where many different paths are often available.

2.3 Ad Hoc Network Performance Analysis

2.3.1 Network Performance Metrics

There are several performance metrics used in the evaluation of network performance and routing protocol performance:

1) Throughput: packets received by the destinations, usually measured in Mb/s. This metric can be overall throughput or average per flow throughput.

2) Average end-to-end delay of data packets: It includes all possible delays caused by buffering during route discovery latency, queuing at the interface queue,

retransmission delays at the Medium Access Control (MAC), and the propagation and transfer times.

3) Packet delivery ratio: The ratio of the data packets delivered to the destinations to those generated by the sources. This metric is always less than one.

4) Normalized routing load: The number of routing packets transmitted per data packet delivered at the destination. Each hop-wise transmission of a routing packet is counted as one transmission.

The first two metrics are the most important for best-effort traffic. The routing load metric evaluates the efficiency of the routing protocol. Note that these metrics are not completely independent. For instance, lower packet delivery fraction means that the delay metric is evaluated with fewer samples.

The networking "context" in which a protocol's performance is measured is considered [MANET, 2003]. Essential parameters that should be varied include:

1) Network size: measured in the number of nodes.

2) Network connectivity: the average degree of a node (i.e. the average number of neighbors of a node).

3) Topological rate of change: the speed with which a network's topology changes.

4) Link capacity: effective link speed measured in bits/second, after accounting for losses due to multiple access, coding, framing, etc.

5) Fraction of unidirectional links: how effectively does a protocol perform as a function of the presence of unidirectional links?

6) Traffic patterns: how effective is a protocol in adapting to non-uniform or busty traffic patterns?

7) Mobility: when, and under what circumstances, is temporal and spatial topological correlation relevant to the performance of a routing protocol? In these cases, what is the most appropriate model for simulating node mobility in an ad hoc network?

8) Fraction and frequency of sleeping nodes - how does a protocol perform in the presence of sleeping and awakening nodes?

2.3.2 System Measurement Metrics

Three ways are available to evaluate a system.

1) Measurement-based evaluation: This is the most accurate method, but could be the most time-consuming and expensive, since one has to have the real system to take the measurements. The method may be infeasible for some early-stage designs and ideas.

2) Discrete-event simulation: it is perhaps the most common way these days to evaluate a system, especially when it comes to a network system in the design phase. Although one does not need to construct a real system, it may take a long time to construct a simulator to represent a real one. The comprehensive simulation runs also cost much, sometimes unaffordable computing resources.

3) Analytical modeling: the less costly and more efficient method is perhaps the third one. It has been a common belief that well-built analytical models are able to shed more lights into the system than the previous two methods. They, however, may require simulation or small-scale pilot system run to support their accuracy and balance the believability.

Despite the nice features of analytical modeling approaches, one needs to be cautious that it may be at the expense of accuracy and believability because of simplification in an attempt to make the model analytically tractable. So, it is indeed very important for us to use alternative methods to check the correctness of the models.

Simulation methods are the widely used methods to analyze the network performance. There are several simulation tools that have been developed using in the design and performance modeling of ad hoc network, such as OPNET [OPNET, 2003] and Network Simulator (NS2) [Fall and Varadhan, 1997]. Comparing with the commercial software OPNET, NS2 has an advantage. It offers the open source code, is free and thus can be modified and reconstructed easily.

Network Simulator (NS2)

NS2 [Fall and Varadhan, 1997] is developed at the University of California at Berkeley. Its extension to mobile ad hoc network is developed by the Monarch project at Carnegie Mellon University (CMU). The CMU extension provides the support for accurately simulating a realistic wireless physical channel, data link and MAC layer models in multihop wireless networks. The radio propagation model is a combination of a free space propagation model (attenuation $1/r^2$) at short distance and a two-way ground reflection model at long distance (attenuation $1/r^4$) with omni-directional antenna. The distributed coordination function (DCF) defined in IEEE 802.11b for the wireless LAN is used as the MAC layer model to prevent the hidden terminal problem and capture phenomenon. This is to simulate a commercial radio interface card, i.e., the WaveLAN from Lucent, which is a shared-media radio with a nominal bit-rate of 2 Mbps and radio

range of 250 meters. A detailed description of simulation environment and models can also be found in [McCanne and Floyd, 2003].

Stochastic Petri Nets (SPN)

Petri Nets (PN) [Zhou and Venkatech, 1999] is a graphical and mathematical modeling tool whose nodes are partitioned into two sets, *places* and *transitions*. They are widely used to model and analyze such discrete event systems (DES) as communication, manufacturing, and transportation systems. Petri nets are excellent tools to model and analyze the synchronization, parallel activities, and conflicts among the processes in a system.

The stochastic Petri net model is obtained from the Petri net model by associating a probability distribution function to the firing time of each transition. Additional constructs are often present as well. In the generalized stochastic Petri net model [Ajmone-Marsan et al., 1984], only two distribution types are allowed: exponential, and deterministic with time delay value 0. Transitions with an associated exponential distribution are said to be *timed*; transitions with zero time distribution are said to be *immediate*. In the Extended Stochastic Petri net model [Dugan et al., 1985], the transitions are classified in a similar way, but an arbitrary distribution can be associated to each timed transition. If two or more conflicting transitions should fire at the same moment (this event has a zero probability if the distribution is continuous), a probability function must specify the probability that a subset of transitions actually fire. It allows the quick construction of a simplified abstract model that is numerically solved for different model parameters. SPNP [Ciardo et al., 1989], based on SPN, is used to build an approximate model for a quick numerical analysis of network performance. Besides the

exponential distributions, SPNP allows us to try several distributions, such as Erlang and deterministic distribution, to analyze the performance under different situations. SPN provides a framework not only for modeling and simulation but also verification and formal analysis of a designed system. Such capabilities are very important in ad hoc network design.

Due to the advantages in quick construction and numerical analysis of Petri nets, several related works have been done to investigate the characteristics of networks.

Xiong et al. (2002) modeled and simulated ad hoc routing protocol using Colored Petri Nets (CPN). They proposed a *topology approximation* (TA) mechanism to solve the problem of topology changes, which is an inherent characteristic of ad hoc network and perform simulations of AODV. Their work mainly focus on the routing, not the performance of network.

Ciardo et al. (1995) modeled a scalable high speed interconnect, which is continuous hexagonal mesh like wired network with stochastic Petri Nets. They presented both exact and tractable approximate SPN model and compared it with simulation results based on CPN. Their work can only be applied to a specific network. The result is not persuadable because it is based on the comparison of two Petri nets models.

Chen et al. (2001) developed a Markov Decision Process (MDP) to analyze call admission control policy for wireless communication networks. They used Stochastic Reward Nets (SRN) to model the paradigm, with handoff call dropout time information incorporated into the decision policy. Their approach can be further developed into ad hoc network.

Trivedi et al. (2000) introduced some recent research on SPN including stochastic reward nets, fluid stochastic Petri nets, and so on. They can be used in ATM networks, Ethernet Bridge, and RF recovery in wireless communication. The author gave two examples in ATM networks to illustrate the performance analysis of computer networks.

None of above researches address on the overall performance of ad hoc network, which is our purpose and will be addressed in the following chapters.

2.4 Summary

Due to the increasing interest in ad hoc network, sustainable development issues have been raised in this area. There are still much research work to do. This chapter reviews some of the recent methodology and technology development activities on ad hoc network design. The issues include the concept of ad hoc network, ad hoc network routing protocols, network security and analysis on network performance. This chapter also reviews some basic tools in the research area and makes comparisons between different routing protocols to make decision. This research addresses some critical issues on ad hoc network security, routing protocol, and performance modeling and evaluation. It should facilitate the design and implementation of robust ad hoc network.

CHAPTER 3

PERFORMANCE ENHANCED SECURE AD HOC ON-DEMAND ROUTING PROTOCOL

In QoS routing for wired networks, multiple path routing is popularly used. Due to the advantage of the inherent redundancy in ad hoc network - multiple routes between nodes, many different protocols have been proposed based on different assumptions and intuitions. Marina and Das (2001) proposed "*advertised hopcount*" to guarantee loop-freedom for an on-demand distance vector protocol. The "*advertised hopcount*" is a route table entry. They use multipath as a backup in route discovery to reduce routing overhead and end-to-end delay. Routing On-Demand Acyclic Multipath (ROAM) [Raju and Garcia-Luna-Aceves, 1999] is an on-demand, multipath distance vector algorithm based on diffusing computations. It can detect network partitions like TORA. But it requires close coordination between nodes because state information must be maintained at each node during the route discovery. Thus ROAM is better suited for static ad hoc networks or networks with low node mobility. Split Multipath Routing (SMR) [Lee and Gerla, 2001] uses a modified flooding algorithm and the data traffic is split among the multiple paths. SMR attempts to build maximally disjoint routes to avoid having certain links from being congested. Its multipath is used as a backup in route discovery. Pearlman et al. (2000) analyze the performance impacts of alternative path routing for load balancing. They use *diversity injection* to compute node-disjoint paths. Leung et al. (2001) proposed distributed multi-path dynamic source routing protocol based on the existing Dynamic Source Routing protocol to improve QoS support with respect to end-to-end reliability. It

seeks to compute a set of unicast routes that can satisfy a minimum *end-to-end reliability* requirement.

This chapter proposes the use of multipath in an on-demand distance vector protocol to achieve better performance and security. It is organized as follows: Section 3.1 proposes the "Security Level" concept and gives the probability analysis. Section 3.2 proposes a performance enhanced routing protocol (SOR) embedding a "maximum hopcount" factor to restrict the number of routing packets for a particular purpose. Section 3.3 implements SOR based on AODV. Section 3.4 evaluates the performance of the proposed protocol by simulations. Section 3.5 gives the summery.

3.1 Security Enhancement

So far many protocols have been proposed and their performance has been well researched. But the security issues and concerns have not been addressed in depth. In general, there are two sources of threats: external and internal attacks. Typically, an attacker can eavesdrop routing information and inject erroneous routing information, replay old routing information, or distort routing information. An attacker can successfully partition a network or introduce excessive traffic load into the network by causing retransmission and inefficient routing. More severe attack comes from compromised nodes in the internal network. A compromised node can broadcast incorrect routing information to other nodes. Its detection through routing information is difficult in an ad hoc network because of its dynamically changing topology: an invalid piece of routing information can be generated by a compromised node, or because of the result of topology changes. It is difficult to distinguish between them. On the other hand, false

routing information generated by compromised nodes can be considered the same as the outdated information. As long as there are sufficiently many correct nodes, the routing protocol should be able to find new routes that pass by these compromised nodes. Therefore, on-demand protocols that are capable of finding multiple paths such as TORA, AODV, and DSR, have a potential advantage to enhance the security. Yet such advantage is not fully explored. The following discussions mainly focus on the combination of the multiple paths into AODV for the security enhancement purpose.

3.1.1 Security Level Concept

Suppose there is a battlefield scenario. Two high-rank military officers want to establish a route to communicate important messages with each other. Because these two officers do not lie in a small region that can enable their communications directly, the route has to pass through some low-rank military officers and soldiers inevitably. Apparently they don't want to disclose these messages to them. Their routing information or messages are more important than those of ordinary soldiers. Therefore, they should be well protected. From the security aspect, when a node is compromised, one needs to prevent information from being leaked or decrypted to this node, which passes through it. Consider certain ability of one node to do the cryptanalysis, an officer's routing information should be harder to be decrypted or acquired than soldiers'. Such situation and multipath routing possibility in ad hoc network area motivate us to classify the mobile nodes into different levels and develop a revised protocol to meet different security requirements.

A "*Security Level*" concept is proposed and embedded into the routing protocols that allow multipath routing. The difference between traditional hierarchy of privileges

and security levels lies in a different communication manner. Ad hoc network communication requires the participation of a lower rank node while a traditional one does not. In a structure embedded with security levels, high priority level nodes keep their message or routing information in a high level security such that the low priority level nodes can only forward this message without the ability to look through the encrypted message. Low priority level nodes are required to encrypt or transmit information with only low-level security. While traditional multipath routing is used in route recovery when a route is broken, one wants to employ it for more secure packet transmission. That is, after a node finds multipaths to another node, it separates its protocol data unit into several pieces and transmit each piece via one path.

This structure integrated with threshold cryptography [Desmedt, 1994] can significantly improve network security. Threshold cryptography could distribute the trust in key management. In an $(n, t+1)$ threshold cryptography, suppose that n nodes share the ability to perform a cryptographic operation such as digital signature so that any $t+1$ nodes can perform this decryption operation jointly. It is infeasible to do such operation by only t nodes, i.e., it is impractical if t nodes want to attack this system corporately. Each node participating in the encryption is called a "neighbor node" of the sender because in the framework, each node does such operation with its neighbor node while finding the route. In order to do the encryption, the private key k of the sender is divided into n shares $(s_1, s_2, \ldots, s_n)$, assigning one share to each of its neighbor nodes. For the node that wants to sign a certificate, each of its neighbors generates a partial signature $(PS(m, s_1), PS(m, s_2), \ldots, PS(m, s_n))$ for the certificate using its own private key share and sends it to the receiver. Using $t+1$ correct partial signatures, a receiver can figure out the

signature for the certificate. If at most t neighbor nodes are compromised, these compromised nodes cannot generate correctly signed certificates with only their partial signatures.

Consider RSA signature generation and decryption. When signing or decrypting, one computes

$$g_b(d) = b^d \ \mathbf{mod}\ n \tag{3.1}$$

where b is the text to sign or the ciphertext, d is the secret key and n is the public modulus. This function g satisfies the property that,

$$g_b(k_1 + k_2) = g_b(k_1) * g_b(k_2) \tag{3.2}$$

Using Shamir's secret sharing scheme [Shamir, 1979], we have:

$$key = \sum_{i \in B} (constant_{i,B}) \cdot (share_i) \tag{3.3}$$

where B is a subset of the set of all neighbor nodes for encryption. Combining (3.2) and (3.3), we have

$$g_{input}(key) = g_{input}(\sum_{i \in B} (constant_{i,B}) \cdot (share_i)) \tag{3.4}$$

$$= \prod_{i \in B} g_{input}(constant_{i,B} \cdot share_i) \tag{3.5}$$

$$= \prod_{i \in B} (g_{input}(share_i))^{constant_{i,B}} \tag{3.6}$$

$g_{input}(share_i)$ can be computed by neighbor nodes and sent together with every node's identification to the receiver. With enough neighbor nodes responding, the receiver can compute $constant_{i,B}$ for all $i \in B$ and then evaluate (3.6).

In a protocol embedded with security levels, each level has a corresponding number: t. The higher t, the more security provided. In the battlefiled situation, t can be

related to the rank. This number means when a node wants to send a message, the message is encrypted with an *(n, t+1)* threshold scheme. Because an on-demand routing protocol can find multipath routes, the sender needs to compare route reply messages in order to choose *n* shortest routes to send message, which ensures security when at most *t* paths can be compromised at a time. That means, when the first RREP message returns to the sender, it can assign one share of its private key *k*, which has been divided into *n* shares, to its neighbor node of the first route, and use the first route to forward the partially encrypted messages to the receiver. This process continues until the n_{th} route is found. After that, if a new route is found, the sender compares all the paths to find the shortest one among them to do the encryption. Even if one cannot find *n* paths to the destination and *t* is less than *n-1*, *t+1* paths are still effective in some cases because the receiver can get the correct message using *t+1* partial signatures. If the receiver only receives less than or equal to *t* partial signatures or the sender doesn't find enough routes, this transmission fails and the retransmission begins. The difference between multipath routing and the proposed method is that multipath routing doesn't classify the mobile nodes and give every node the same bandwidth to transmit message. Ad hoc network resource is saved by giving different security-level nodes different ability to transmit, especially in a situation where only few high security level nodes exist among all the participating nodes.

Consider internal attacks. When one node is compromised, it could generate an incorrect partial signature that would yield an invalid signature. This is prohibited by verifying the validity of a computed signature using a node's public key. If the verification fails, the receiver tries another set of *t+1* partial signatures. This process

continues until the receiver constructs the correct signature from $t+1$ correct partial signatures.

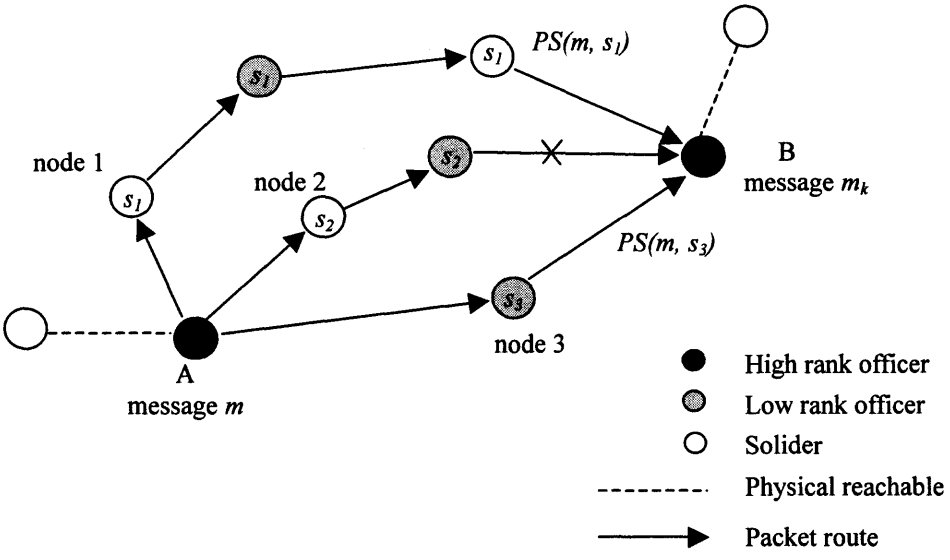

Figure 3.1 An illustration of security level.

In a case illustrated in Figure 3.1, which includes three security levels, a high rank officer A wants to communicate with another high rank officer B in an ad hoc network. Decided by A's level, a *(3, 2)* threshold cryptography scheme is chosen to sign certificates. First, node A finds a path to node B based on an on-demand routing protocol. After receiving route reply messages, A needs to choose three shortest paths to distribute private key k to neighbor nodes 1, 2, and 3, named s_1, s_2, and s_3. Then these neighbor nodes all generate partial signatures and forward the signatures to node B. When route two is broken or one node in this route is compromised which lead to the failure to submit s_2, B still has the ability to generate the signature m_k of m signed by A's private key k finally. However, if the link from s_3 to B is also broken, then the transmission

cannot succeed. Node A needs to find additional shortest paths. In case such paths do not exit, A cannot send any message to B successfully.

3.1.2 Probability Analysis

Yang and Papavassiliou (2001) demonstrated that redundant traffic dispersion had a smaller connection intrusion probability than that of single path routing. The proposed scheme is evaluated by calculating the probability that a transmission from the source results in successful packet reception at the destination via *(n, t+1)* threshold cryptography.

There are three assumptions to simplify the analysis without losing the generality:

1) The mean time of packet transmission is much smaller than the mean time between variations in network topology. This means that the topology of the network does not change significantly while a packet is being transmitted.
2) The transmission is node-disjoint routing, which means there are no shared intermediate nodes or links from the source to the destination.
3) Every node has the same probability for it to receive successful attack, which is defined as p.

Suppose that one has a scheme that includes N nodes and n paths. A path being compromised means that at least one node on that path is compromised. If a *T(n, t)* scheme, which means an *(n, t)* threshold cryptography scheme, is compromised, then at least t paths in this scheme are compromised. Each path i has q_i nodes and k_i is the number of nodes being likely to be compromised in this path. Suppose that one has total

K nodes being compromised. Then the probability that a *T(n, t)* scheme is compromised is given by

$$P_{n,t} = \sum_{k_1=0}^{\min(q_1,K)} \sum_{k_2=0}^{\min(q_2,K)} \cdots \sum_{k_n=0}^{\min(q_n,K-\sum_{j=1}^{n-1}k_j)} \prod_{i=1}^{n} \binom{q_x}{k_x} p^K (1-p)^{N-K} u_1 u_2 \qquad (3.7)$$

where u_1 and u_2 are defined as

$$u_1 = \begin{cases} 1 & if \sum_{j-1}^{n} k_j = K \\ 0 & otherwise \end{cases} \qquad u_2 = \begin{cases} 1 & if \sum_{j-1}^{n} v_j \geq t \\ 0 & otherwise \end{cases}$$

where

$$v_j = \begin{cases} 1 & k_j \neq 0 \\ 0 & k_j = 0 \end{cases}$$

u_1 is used to guarantee that there are total K nodes being compromised and use u_2 to make sure that at least t paths are compromised.

For instance, suppose that one has a four path routing scheme similar to Figure 3.1. Each path has two, three, four, and five nodes, respectively. Each node has the same probability of 0.2 being compromised. Give the condition that total K nodes have been compromised among all four paths, Figure 3.2 compares the probability that a *T(4, t)* scheme is compromised with different t. It is observed that all three curves approach zero after four nodes the in the total scheme are compromised. This happens because four or more nodes being successfully attacked is an event with small possibility in the scheme by giving the small probability p that each node is compromised. *T(4,3)* scheme has a much larger probability to keep the information safer than the other two. Especially when two nodes are attacked, the probability that a *T(4,3)* scheme leaks out the information is zero. Thus, *T(4,3)* scheme can be assigned to a high security level compared to other two schemes. *T(4,1)*, which means the system is considered unsafe if one of its paths is

compromised, is a reference that will not be adopted usually because it has no advantage than single path routing. Even worse, because *T(4, 1)* scheme needs 3 additional paths to transmit the same information comparing with single path routing, it wastes bandwidth that is an important factor in ad hoc networks.

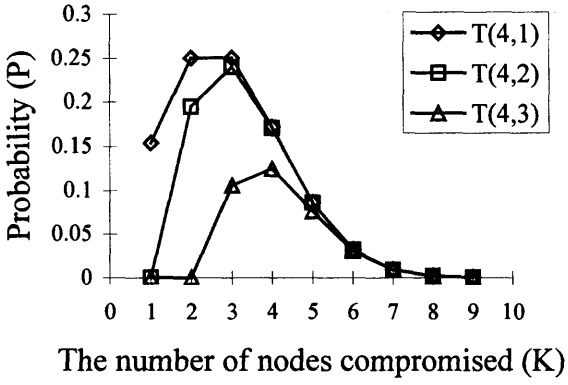

Figure 3.2 Comparison of probabilities being compromised with different *t*.

3.2 Hopcount Restriction

An ad hoc network has been designed for use by mobile nodes. In some cases, a collection of mobile nodes may operate in a fixed relationship to each other and move together within an area where an ad hoc network has formed. Unlike position-based routing protocol such as LAR and DSR using Global Positioning System [Basagni et al., 1999], one assumes that the moving range of nodes is pre-designated while position-based routing protocol use geographic position as an inherent property. Given a simulation field with dimensions 670m x 670m used in ns2, this research simulates the routing under a random situation, i.e., each node starts its movement from a random location to a random destination with a random speed. For 20 mobile nodes, all of the routing procedure's hopcount is less than or equal to three while the hopcount used

needed for 30 nodes is less than or equal to four, as shown in Figure 3.3. This figure implies that the routing procedures end in limited hops. In other words, any route request packet beyond that hopcount is not useful. Furthermore, the hopcount depends on the dimension of the region that holds the mobile nodes. The larger the area, the more hops needed to find a path. This is reasonable because of the limited propagation range of a node.

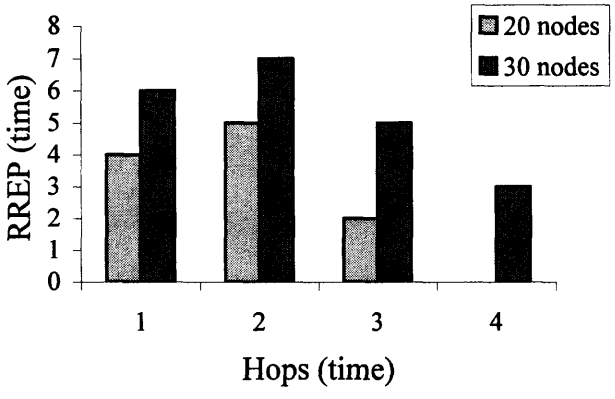

Figure 3.3 Hopcount used in routing.

Suppose that the nominal radio range for a wireless LAN is d (unit is meter). Given an x × y rectangular region, assuming that the nodes are uniformly distributed, it is predicted that most connections use at most h hops to reach the destination where:

$$h = \left\lfloor \frac{\sqrt{x^2 + y^2}}{d} \right\rfloor \tag{3.8}$$

where $\lfloor a \rfloor$ means the largest integer less than or equal to a.

Equation (3.8) can be used only when the number of nodes reaches certain density. One problem in the communication between arbitrary nodes is the connectivity of wireless links when mobile nodes are likely to be moving independently of one

another. Consequently, node mobility causes the frequent failure and activation of links, leading to increased network congestion while the network's routing algorithm reacts to the topology changes. Unlike fixed infrastructure network where link failure is comparatively rare events, the rate of link failure due to node mobility is the primary obstacle to routing in ad hoc network.

If the nodes are distributed sparsely, hopcount becomes a useless metric because most of the nodes are unreachable from others. Hence, the relationship between node density and the connectivity has to be explored.

Consider a given square area with dimensions 670m × 670m. Nodes are generated using a random number generator [Park and Miller, 1988] provided by ns2 and uniformly distributed. Consider only the topology without considering network parameters such as bandwidth, congestion window size, energy, and so on. When n nodes in this area, all possible connections are given by:

$$The\ number\ of\ all\ possible\ connections = C_n^2 = \frac{n \times (n-1)}{2}$$

Note that there are multiple paths between two nodes. The wireless propagation range d used in simulation is set to 250 meters, which is similar to a Lucent's WaveLAN [Tuch, 1993] that is a shared-media radio with a nominal bit-rate of 2 Mb/sec. Under the wireless situation, one unreachable connection means that two nodes cannot communicate directly or via any intermediate nodes. Counting all unreachable connections, the unreachable rate is defined as:

$$Unreachable\ rate = \frac{all\ unreachable\ connections}{all\ possible\ connections}$$

The unreachable rate value varies for every simulation because of the random case. After a large-scale test, a statistical result of the unreachable rate versus the required number of nodes in a given region is obtained as shown in Figure 3.4. Each result is based on 5000 simulations. When there are only 2 nodes in this region, the unreachable rate is as high as 70%. Considering an acceptable unreachable rate of 0.1, which means that 90% connection attempts can be reached, at least 16 nodes are needed in this area.

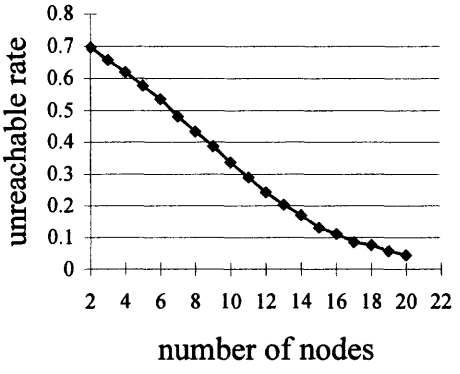

Figure 3.4 Unreachable rate for a given area with 670m× 670m.

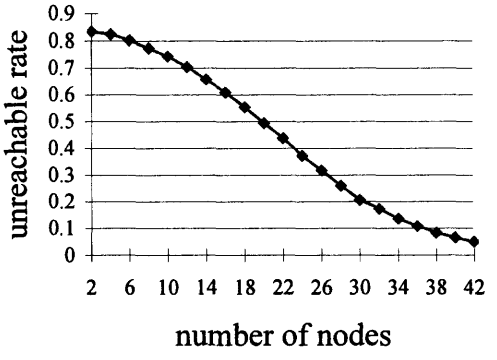

Figure 3.5 Unreachable rate for a given area with 1000m× 1000m.

Figure 3.5 shows another simulation based on the dimensions 1000m× 1000m. The generated curve is similar to the previous one. Furthermore, the number of nodes that represent an acceptable unreachable rate of 0.1 increases proportional to the area, which means n can be acquired by Figure 3.5 or the calculation below:

$$n = 16 \times \frac{1000 \times 1000}{670 \times 670} \approx 36$$

One can derive from (3.8) as follows: in a square $(x \times x)$ region and n nodes moving in this region, most connections, which means that the unreachable rate is below 0.1, use at most h hops to reach the destination where:

$$h = \left\lfloor \frac{\sqrt{2}x}{d} \right\rfloor \qquad n \geq N \qquad (3.9)$$

where:

$$N \approx \frac{x^2}{670 \times 670/16}$$

Given (3.9), the hopcount of a routing packet is restricted by discarding it if it is beyond *h*. The research uses this method to decrease the number of routing packets, reduce the flooding phenomena, and save network resources. Note that (3.8) is suitable for rectangular region while (3.9) is suitable for square region. Using these two formulas with the security level scheme, we expect to reach a good tradeoff between resource consumption and security requirement.

Based on the ad hoc on-demand routing protocol and "Security Level" concept, a performance enhanced routing protocol (SOR) embedding a "maximum hopcount" factor is proposed to restrict the number of routing packets.

3.3 Implementation of SOR

The security advantage due to multipaths has been explained before using probability analysis. Even if one piece is captured by a compromised node, the whole information remains safe. The "*Security Level*" concept can be used to enhance the security aspect of a routing protocol bound with the identity of a user. Without this binding, any user can impersonate anybody else and obtain the privileges associated with higher security levels. To prevent this, messages must be protected by using authentication, confidentiality and integrity services, such as those involving the generation of unforgeable and cryptographically strong message digests or digital signatures.

Each security level in SOR has a corresponding number: l. This number means that a message is encrypted and separated into l pieces when a node wants to send the message. The larger l, the higher level and more security provided. For instance, l can be related to the rank in the army. The smallest value of l is one, corresponding to the lowest security level. That means the single path routing as it is in an ordinary routing protocol. Because an on-demand routing protocol can find multi-path routes, a sender needs to compare route reply messages in order to choose l shortest path routes to send a message. That is, after confirming that there are l possible routes, the sender can use these routes to forward the encrypted messages to the destination. This process continues until some routes are broken. If n paths are disconnected, the sender needs to find n instead of l new paths. The research economizes ad hoc network resources by giving different-security-level nodes different ability to transmit, especially in a scheme where only a few high-security-level nodes exist among all the nodes. Such scheme is common when one constructs an ad hoc network. Suppose that 80% mobile nodes belong to the lowest security level. The routings between them acts as single path routing. Therefore, network performance keeps unchanged if one considers these nodes only. Even if the throughput and delay varies after considering high security level nodes, it is predicted that the network overload may not be significantly larger because only 20% nodes need multipath routing. The performance between single path and proposed multipath scheme will be compared later.

3.3.1 Security Level Implementation

Because of the advantage of finding multipaths in ad hoc on-demand routing protocol, SOR is built as an augmentation to the AODV. The researcher retains the most of AODV's original behavior, such as on-demand route discovery using flooding, and reverse path maintenance in intermediate nodes. The route table entry is modified to carry additional multipath information. Table 3.1 gives a comparison of the structure of routing table entries of AODV and SOR.

Table 3.1 Difference of Routing Table Entries of AODV and SOR

AODV	SOR
Destination address	Destination address
Destination sequence number	Destination sequence number
Hopcount	Hopcount
Nexthop	{nexthop[0], nexthop[1], nexthop[2],...}
Expiration time	Expiration time
Not applicable	Security level
...	...

In SOR, the notion of nexthop is extended to a nexthop array that defines multiple next hops with respective routes. For instance, considering a source node numbered 0, nexthop[0] equals 2 means in the 0^{th} route to a certain destination, the nexthop that node 0 should forward the packet is node 2; nexthop[1] equals 4 means that the nexthop is

node 4 in the 1st route. However, all next hops still have the same destination address and destination sequence number.

A new notion of "*security level*" is added to each node's routing table entry. Security level is set only once during the initialization and does not change during the process.

When node i wants to communicate with node j, it needs to decide first how many paths it should use, abiding the following rule:

$$l_{i,j} = min(l_i, l_j) \tag{3.10}$$

where

$l_{i,j}$ is the actual number of paths that node i uses

$l_i, and\ l_j$ are the security levels of node i and node j, respectively

Equation (3.10) shows that when two nodes with different security levels want to communicate with each other, the number of paths that they use is determined by the nodes with lower security level. This situation is determined by the cryptological analysis ability of one node. Usually lower security level nodes are assigned low-end devices with lower computational ability. Hence these nodes need more time to do the encryption and decryption than higher-level nodes do. It is assumed that the information exchanged between higher and lower level nodes is less important than the information exchanged between two nodes with the same higher level. Hence, the former's protection can be relatively lower than the latters.

A node-disjoint route is used to secure a message when finding the multipath for one connection. A disjoint route includes node-disjoint and link-disjoint routes. Node-disjoint route means that common nodes are strictly prohibited during routing, except the

source and destination. In contrast, link-disjoint routes do not have any common link. Note that link-disjoint may have common nodes. Node-disjoint restriction presents a smaller number of disjoint routes. This makes node-disjoint less effective. On the other hand, from the security point of view, a link-disjoint route is considered more risky than node-disjoint route since common nodes may be compromised.

In the original AODV, any node discards the RREQ packet if it has already received it, including the destination that a message will finally reach. In SOR, this rule is still effective for intermediate nodes. But for the destination, it sends RREP regarding each RREQ packet it has received and updates its route table entry for different routes. One additional field named "*pathno_no*" is added to the RREP. This item indicates the i^{th} path that the destination has received. *pathno_no* is started counting from zero. Similarly, any nodes in the RREP back route updates their route table using the *pathno_no* including in the RREP packet. For instance, if source node 0 receives a RREP packet including *pathno_no* 1 from node 4, it updates its route table entry with nexthop[1] equaling 4. For intermediate node that has the latest known route to the destination, it discards the same RREQ packet because of the node-disjoint restriction.

3.3.2 Implementation for Hopcount Restriction

The AODV's RREQ parameter is modified with a new notion of "*maximum hopcount*". The maximum hopcount of a source node to a destination represents the "maximum" hopcount that a single path can traverse. For instance, maximum hopcount four means that a source node needs to find a path within four hopcounts, that is, this path's maximum number of intermediate nodes is three. When a node receives RREQ, it follows

the normal procedure that AODV has defined. Only when it decides to forward this packet, it checks the RREQ's hopcount to make sure that the hopcount is smaller than the maximum hopcount. If the hopcount has exceeded the threshold, this RREQ packet is discarded. Note that this process is performed after the increase of the hopcount and before a forward operation.

The maximum hopcount is specified when a node initializes its RREQ packet, and remains unchanged until time out. If it cannot find a path to the destination, which means either it needs higher hopcount or the packet has been congested because of the limited bandwidth or buffer of wireless resources, it automatically increases its maximum hopcount by one and sends a new RREQ as described in AODV. This step increases the probability to find a new path and guarantees the same final result as original AODV can obtain. There is no upper limit to this maximum hopcount increase. Note that the increase of the maximum hopcount does not directly mean the increase of routing packets. Suppose that an isolated node wants to communicate with others. The distance between it and the nearest node is greater than d. No matter what the maximum hopcount it has, the total number of routing packets between it and others remains unchanged. In some cases, one node's maximum hopcount increase consumes additional resources and energy considering this node only. But after evaluating the whole ad hoc network, network resources are still saved by restricting the hopcount.

3.3.3 Implementation of SOR

Figure 3.6 is the flow chart of SOR for a specific node i. After node i chooses initial security level, it sets its routing table entry. Routing table entry is for one node only. For

each destination, node *i* creates a routing table entry. RREQ packet is set after node *i* chooses initial maximum hopcount. Note when node *i* needs to send RREQ again because of the link broken, it has to use initial maximum hopcount instead of currently used maximum hopcount to set RREQ.

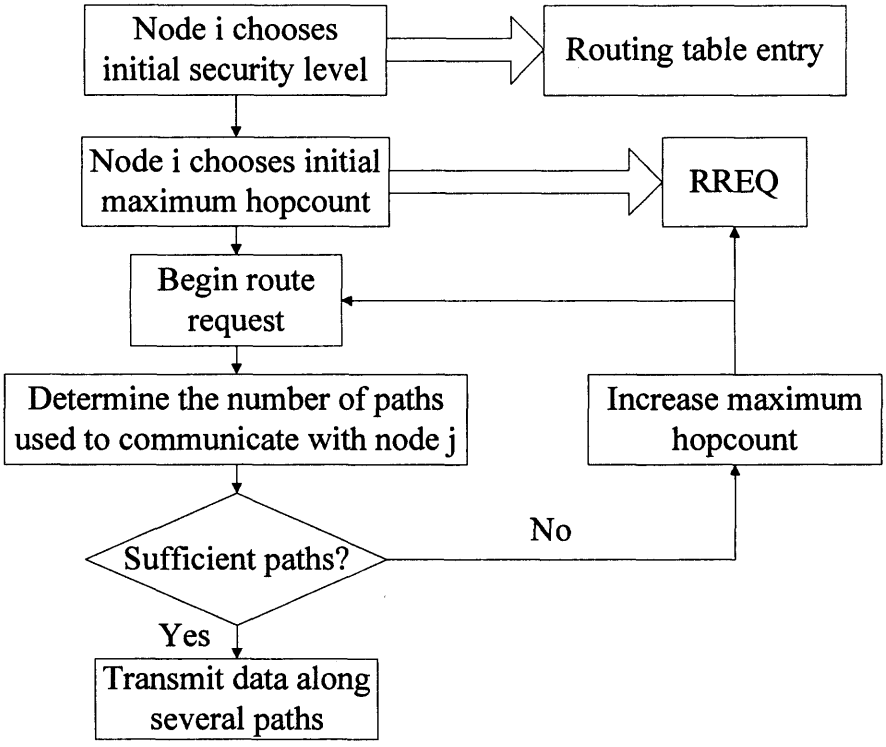

Figure 3.6 Implementation of SOR.

3.4 Simulation

A detailed simulation model is used based on the latest ns2. In the experiment, random waypoint model is used to simulate random node mobility model. Maximum speed of a node varies from 0 m/s to 20m/s to reflect different mobility. An ad hoc network in a field with dimensions 670m × 670m and 1500m × 300m is used. The number of nodes varies from 10 to 60. Traffic sources are CBR, i.e., continuous bit-rate. The source-destination pairs are spread randomly throughout the network. Only 512 byte data

packets are used. The packet rate at the source nodes is 4 packets/sec. The number of sessions increases along with the increase of the number of nodes. All traffic sessions are established at random times and stay active until the end. Simulations are run for 300 simulated seconds. Each data point in the following figures is the average results. Similar simulation environment has been used before in several recent performance studies on ad hoc networks [Marina and Das, 2001; Broch et al., 1998; Perkins et al., 2001].

The security level and hopcount restriction are combined together into SOR. Security level is set to two for simplicity. Level-l_h and level-l_l represent the level of high security level nodes and the level of low security level nodes, respectively. Level-l_h equals 2 and 10% nodes belong to this level, which means they need to find two separate paths and transmit split CBR using two paths separately when they talk with each other. Level-l_l equals 1 and 90% nodes belong to this level. Their sessions between level-l_h and level-l_l act as in a single path routing. The following parameters are considered that has been defined in Chapter 2: (1) routing load — the number of routing packets for each session; (2) packet delivery fraction — ratio of the data packets delivered successfully to the destination to those generated by the CBR sources; and (3) average end-to-end delay of data packets — this includes all possible delays caused by buffering during route discovery, queuing delay at the interface, propagation and transfer times, and retransmission delays at the MAC layer.

Routing load can be acquired by the total number of routing packets divided by total number of sessions. Note that much more routing packets are needed when nodes move. When two paths are used to send one packet, this packet is delivered successfully after the destination receives two pieces together. The result of SOR is an integrated

result of the multipath and hopcount restriction. This work uses packet delivery fraction and average delay to evaluate the multipath performance of SOR.

3.4.1 Normal Security Level Simulation

Figure 3.7 shows the routing load as a function of nodes in a 670m × 670m area. Maximum hopcount is set to three based on (3.9). A source node needs approximately 100 routing packets to find a route in a 10 node network; while in a 50 node network, it needs 900 routing packets to accomplish the same connection. There is no doubt that routing load increases along with the number of nodes due to the flooding property of on-demand routing protocol. Performance of SOR and AODV are similar in the sparse node density case. Their performance becomes apparently different at more node cases. As expected, SOR generates fewer routing load than AODV (10%-30%) when most connections are successful. Figure 3.8 shows the routing load as a function of nodes in a 1500m × 300m area. Maximum hopcount is set to six. The work does not simulate 10 node case in Figure 3.8 because most sources cannot find destinations in a wide area owing to this small number of nodes. For the large area, the results similar to the small area's are obtained as shown in Figure 3.8.

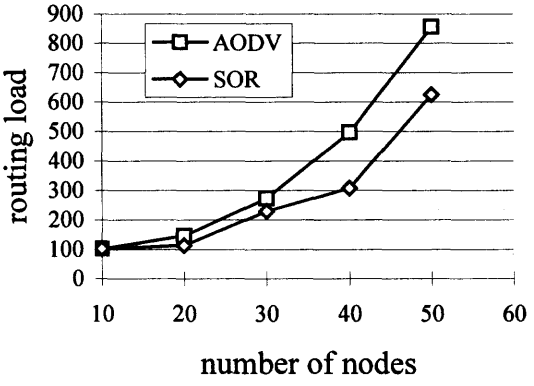

Figure 3.7 Routing load with 670m× 670m.

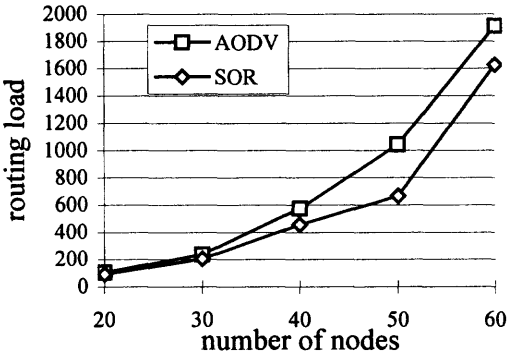

Figure 3.8 Routing load with 1500m× 300m.

Figure 3.9 studies the packet delivery fraction as a function of nodes in a 670m× 670m area with the maximum speed 10m/s. A desirable result is obtained when 20 nodes move. It has a sharp drop when the number of nodes varies from forty to fifty. This is because excess crowd increases the routing load and the network is already under saturation (over 40% packets are dropped by both protocols). At each point, SOR has a slightly higher packet delivery fraction than AODV has although some of SOR routes need more paths. This is mainly because SOR reduces more routing load than AODV.

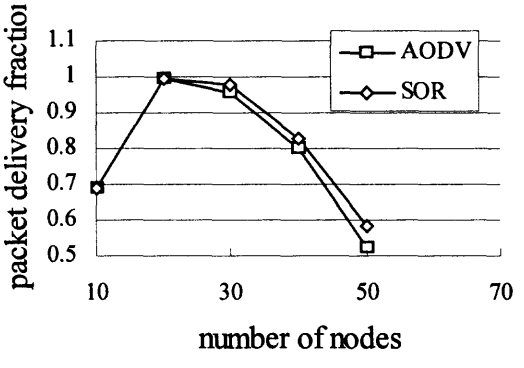

Figure 3.9 Packet delivery ratio with 670m× 670m.

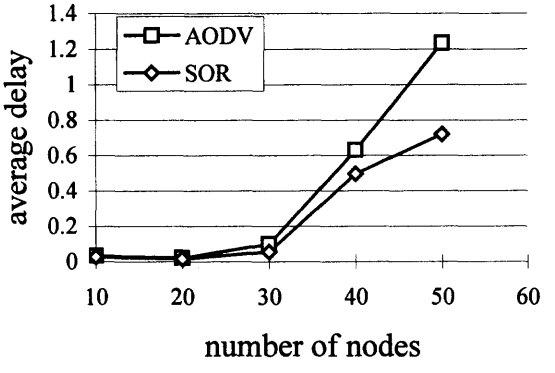

Figure 3.10 Average delay with 670m× 670m.

Figure 3.10 shows the average end-to-end delay in a 670m× 670m area measured in seconds. The maximum speed remains the same as 10m/s. Average delay reaches an unacceptable high value while 40 or 50 nodes are in this area. This is because useless routing packets waste a large portion of network resources. SOR has a tremendous reduction comparing to AODV (around 50%) in high density nodes network. Besides the contribution of hopcount restriction, multipath in SOR can also reduce the average delay because it requires less bandwidth in a link between two nodes.

3.4.2 Internal Enemy Attack Simulation

In order to check the impact of multipath to ad hoc network, a "hypothetical" enemy is simulated inside the network. It compromises an individual node randomly and eavesdrops or blocks any packets, including routing packets and data packets, which go through this node. The difference between eavesdropping and blocking is that eavesdropping does not impact the normal data transmission of the whole network. Eavesdropping only analyzes the packets that pass through the compromised node. Blocking packets means that no packets can pass this node, which results in error message sending from its neighborhood. The action of blocking can significantly reduce the performance, but the routing protocol can detect a compromised node and find new routes that pass by it. Thus, information will not be leaked out after a short time.

A simplified threshold cryptography [Desmedt, 1994] is used to keep the information safe. In an (n, $t+1$) threshold cryptography scheme, a data packet is divided into n pieces. In correspondence with such a scheme, SOR needs to find n paths for multipath transmission. After any $t+1$ pieces arrive at the destination node, one data

packet is assumed to transmit successfully. It is infeasible to recover the whole data packet for up to t pieces, i.e., it is impractical if only t nodes attack this scheme corporately. There is a slight modification of previous SOR at the time when a source node begins to find a new path. As long as $t+1$ paths are valid, the source node does not need to find a new path. In other words, the source node discards the error message sent from an intermediate node. Only if a destination node sends an error message to the source node, a new-path-find process begins. The difference between multipath routing and the proposed method is 1) multipath routing does not classify the mobile nodes and 2) it gives every node the same bandwidth to transmit messages. Ad hoc network resources are saved by giving different security-level nodes different ability to transmit, especially in a scheme where only a few high security level nodes exist among all the nodes.

A simplified (3, 2) threshold cryptography scheme is simulated in SOR. Level-l_h equals three and one assigns 20% nodes to this level. If a compromised node can eavesdrop or block any two pieces of a level-l_h packet, this route is considered being compromised. Level-l_l equals one. All packets going through a compromised node are considered unprotected. The effect of blocking or eavesdropping is discussed below separately.

Routing load with 670m × 670m under the effect of blocking is shown in Figure 3.11. Comparing to Figure 3.7, the routing load has a distinct increase at each point. A source node needs 2-3 times more routing packets to find a route than Figure 3.7's. This is mainly because a compromised node makes route unavailable frequently. SOR significantly reduces routing packets by multipath routing compared with AODV. Most

of the level-l_h paths are kept available when one path is compromised. The more level-l_h nodes participate in the network, the more routing load SOR reduces.

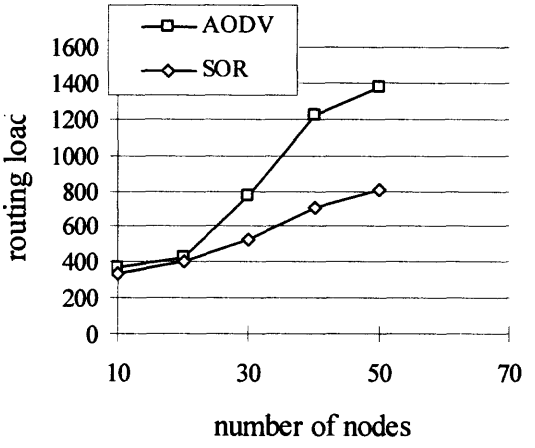

Figure 3.11 Routing load with a compromised node.

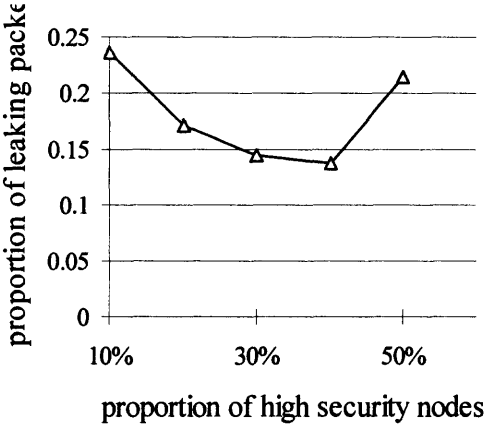

Figure 3.12 Proportion of leaking packets.

Figure 3.12 shows the ratio of leaking packets to those generated by the CBR sources with 670m × 670m under the effect of eavesdropping. Note that one compromised path does not imply any leaking packets because of the implementation of the security level. Leakage happens when two paths are compromised for level-l_h. For level-l_l destination nodes, one path is compromised is defined as a leakage. The work does not allow level-l_h nodes to be the majority of all nodes because it violates the assumption in Section 2. When the proportion of level-l_h nodes varies from 20% to 40%, one can keep over 80% information safe. Comparing to over 30% of leaking packets in the original AODV, the proposed scheme is effective against the attack of eavesdropping. If there are too many level-l_h nodes inside the network, the proportion of leaking packets begins to rise. This is because more level-l_h nodes need more intermediate nodes to participate in

the transmission, especially when level-l_h nodes need to find three paths. Thus, the probability that their messages go through the compromised nodes rises as well. Figure 3.12 illustrates that maximum security is achieved when the proportion of high security nodes is around 40%.

3.5 Summary

Dividing the traffic into multiple routes helps distribute the load to the network hosts. Multipath routing can also be used in on-demand protocols to achieve security. This chapter analyzes the security of ad hoc routing algorithm with regard to the protection associated with the transmission of routing messages, proposes a performance enhanced secure ad hoc on-demand routing protocol (SOR) embedded with "Security Level" concept, and uses "maximum hopcount" to restrict the number of routing packets in a given area.

The proposed scheme provides customizable security to the flow of routing protocol messages themselves. The work has studied the performance of SOR relative to AODV under a wide range of traffic scenarios. SOR offers a significant reduction in average packet delay and provides up to about 30% reduction in routing load. A "hypothetical" enemy is constructed that can compromise nodes inside the network. Through these compromised nodes the advantage of SOR over AODV is investigated in the aspect of security. In general, SOR offers an alternative way to implement better security in on-demand routing protocol.

CHAPTER 4

MODELING AND ANALYSIS OF AD HOC NETWORK WITH STOCHASTIC PETRI NETS

4.1 Introduction

In order to analyze the performance of ad hoc networks such as average delay, data throughput, and packet delivery fraction, a large quantity of research work has been done. New ad hoc routing protocols are proposed and the existing ones are compared against each other. However, only few papers address the network architecture as a whole traffic model. In particular, more investigations are needed in forming the actual environment in which the protocols are expected to operate. Ad hoc network is too complex to allow analytical study for explicit performance expressions. One of the widely used approaches is using simulation methods. However, there are two main drawbacks in using such methods: First, they may be time consuming to execute the necessary simulations. Imaging in a highly variable scenario, with the number of nodes ranging from tens to thousands, node mobility varying from zero to tens of m/s, the simulation time of most current systems will increase dramatically to an unacceptable level. Second, it may be difficult to achieve results that are precise enough. In order to get a reliable value, one has to run simulation tens of iterations with different seed values of a random generator.

Petri Nets (PN) [Zhou and Venkatech, 1999] is a graphical and mathematical modeling tool. They are widely used to model and analyze such discrete event systems as communication, manufacturing, and transportation systems. Petri nets are excellent tools

to model and analyze a system that exhibits such features as parallelism, synchronism, and conflicting events.

An PN can be defined as a 5-tuples:

$$(P,T,I,O,M)$$

where

1) $P = \{p_1, p_2, ..., p_n\}$ is a finite set of places.

2) $T = \{t_1, t_2, ..., t_s\}$ is a finite set of transitions.

3) $I: P \times T \rightarrow \{0, 1\}$ is an input function/matrix that defines the set of directed arcs from P to T.

4) $O: P \times T \rightarrow \{0, 1\}$ is an output function/matrix that defines the set of directed arcs from T to P.

5) $M: P \rightarrow N$ is the marking vector whose i^{th} component represents the number of tokens in the i^{th} place.

The advantages of applying a PN formalism to ad hoc network system performance analysis are summarized as follows:

1) PN's graphical nature allows one to visualize the structure of an ad hoc network system and make the models relatively simple and legible, and

2) PN's mathematical foundation allows one to express the dynamic behavior of a system in algebraic forms.

This work intends to present an approach to the modeling and analysis of large-scale ad hoc network systems using Petri nets. There are two requirements to effectively represent network features. First, a model should be detailed enough to describe some important network characteristics that have a significant impact on performance. Second,

it should be simple enough to be scalable and analyzable. Our goal is to build up a generalized model dealing with different environments. If a probability distribution function is associated to the transition, and random distribution function time delay to the firing time of each transition, the resulting net class is called Stochastic Petri net (SPN) [Ajmone-Marsan et al., 1984]. SPNP is used to build an approximate model for a quick numerical analysis of network performance. SPN consists of places and transitions as well as a number of functions. Enabled transitions fire according to their exponentially distributed time delay. The exponential distribution is required in order to derive a Markov process from SPN. It allows the quick construction of a simplified abstract model that is numerically solved for different model parameters. Besides exponential distribution, SPN allows us to try several other distributions, such as Erlang and deterministic distributions, to analyze the performance under different situations. SPN provides a framework for modeling, simulation, verification, and formal analysis of designed systems. This is very important in ad hoc network design.

Due to the advantage in the quick construction and numerical analysis of SPN, several related works have been done to investigate the characteristics of communication networks. Xiong et al. (2002) modeled and simulated ad hoc routing protocol using colored Petri Nets (CPN). They used a topology approximation mechanism to solve the problem of topology changes, which is an inherent feature of ad hoc networks. Ciardo et al. (1995) modeled a scalable high speed interconnect, which is continuous hexagonal mesh like wired network, with stochastic Petri Nets. They presented both exact and tractable approximate SPN model and compared it with simulation results based on CPN. Their work can only be applied to a specific network, however, Chen et al. (2001)

developed a stochastic reward nets to analyze call admission control. They incorporated handoff call dropout time information into the decision policy. Trivedi et al. (2000) introduced some recent research on SPN including stochastic reward nets, fluid stochastic Petri nets, and so on. They can be widely used in ATM networks, Ethernet Bridge, and RF recovery in wireless communication. However, none of above researches addresses on the overall performance of ad hoc networks, which is addressed in the following sections.

The remaining chapter is organized as follows. Section 4.2 presents a stochastic Petri net model to represent an ad hoc network. Section 4.3 compares the numerical results of the proposed SPN model with simulation results. Section 4.4 gives the summary.

4.2 System Model

4.2.1 Basic Concept

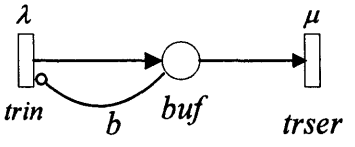

Figure 4.1 SPN model for M/M/m/b queue.

Using SPN to model network is not novel. Figure 4.1 represents a finite buffer M/M/m/b queue in SPN, where m is the number of servers and b is the capacity of the queue. Transition *trin* represents the arrival of a customer with firing rate λ. An inhibitor arc with multiplicity b from place *buf* to *trin* represents the capacity of the queuing

system. Transition *trin* is disabled when the number of tokens in the *buf* equals or is greater than *b*. The transition rate of *trserv* is defined as:

$$rate = \begin{cases} \#(buf)\mu & if(\#(buf) < m) \\ m\mu & otherwise \end{cases}$$

where *#(buf)* represents the number of tokens in place *buf*.

From Figure 4.1, one can easily calculate the network performance parameter as:

$$Throughput = E[rate(trser)]$$

$$Average\ queue\ length = E[\#(buf)]$$

where *E[]* means the steady-state average throughput.

However, this model is far from enough to describe a real ad hoc network. It cannot describe the operation of routing protocols; and a complicated system with different transition rates cannot be illustrated.

Figure 4.2 illustrates a simple ad hoc network's packet flow with three nodes. Each node has a finite buffer that can hold *w* packets. When packets arrive or are generated, it can either enter the waiting buffer or be discarded immediately because of the full buffer. Dashed lines between two nodes mean wireless connections. A packet is being transmitted successfully between nodes 1 and 2. Because node 3 is moving away from node 1, their connection is broken. Therefore, any packets being transmitting are lost. Considering the dynamic topology of an ad hoc network, packet loss may happen frequently.

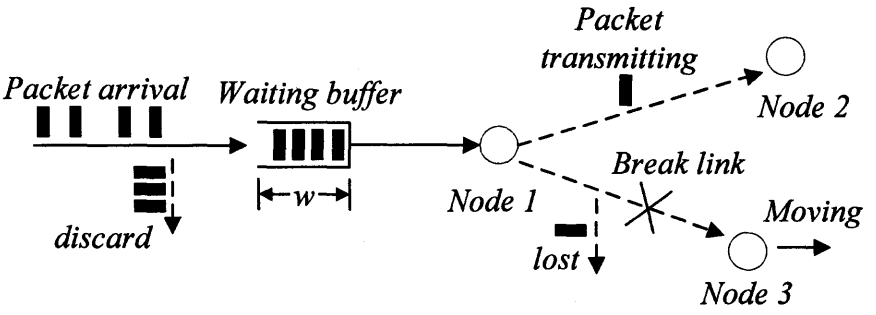

Figure 4.2 Packet flow in ad hoc network.

When an ad hoc network is modeled, one cannot construct such structure by placing nodes into it one by one. Its size will expand too large for an exact numerical solution. This research would rather describe an approximate model based on the idea of SPN decomposition [Ciardo and Trivedi, 1993]. This approximate model exploits a large number of nodes and essentially describes the behavior of one node under a workload that is generated by the whole ad hoc network. Thus the basic idea is to approximate and generate a proper amount of traffic going through one node in a network of a particular size. A fixed-point iteration scheme is used to derive results.

An approximate SPN model is constructed from outgoing and incoming subnets representing different node activities from the perspective of a single node. The former models the packets that are transmitted from the current node to another node. The latter models the current node that is dealing with packets from outside.

4.2.2 Outgoing Subnet Model

The outgoing subnet is shown in Figure 4.3. Subscript *o* and *i* are used to represent outgoing and incoming, respectively. Transition *Ao* generates the packets at a given rate λ and puts them into place *WBo*. An inhibitor arc with cardinality *Co* from *WBo* to *Ao* is

needed to ensure that the number of packets waiting to enter the current node is finite and bounded by *Co*.

Place *Buffer* contains tokens corresponding to free buffer spaces inside the current node. *Buffer* is shared by incoming and outgoing packets. The initial number of tokens is the total number of buffer spaces in a node. The immediate transition *GBo* reserves a buffer space for outgoing packets and puts it into place *RBo*.

After a token arrives in place *RBo*, two possibilities are

1. Because of the shortage of buffer spaces, physical failure during transmission, rapid movement of mobile nodes, or predefined timeout during waiting, the token (message) is dropped out from the network and discarded. Since the work is using one node to represent the whole network, this process can happen at any time during transmission, for instance, when packets are in the source node, destination node or intermediate node's buffer. When immediate transition *DPYo* fires, the token is moved to place *RDo*; and
2. Nothing happens to the token. It still remains in the buffer. When transition *DPNo* fires, the token is forwarded to place *RTo*.

The probability, denotes by α, that a token remains in the buffer without being dropped depends upon the size of an ad hoc network, buffer capacity, node density distribution, transmission rate of packets, and node mobility. α is a variable in the SPN model. Once α is assigned to *DPNo*, The probability for *DPYo* is $1-\alpha$.

Timed transition *DFo* represents the completion of the dropping, after which one buffer in the current node is released by returning a token to place *Buffer*. Timed

transition *TFo* means that packets are successfully transmitted to another node and a token is released to *Buffer*.

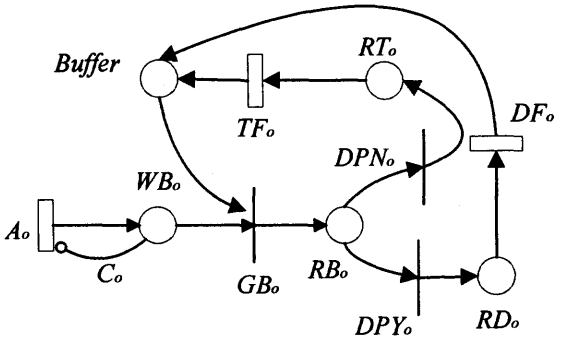

Figure 4.3 SPN outgoing subnet: packet transfer from current node to another node.

4.2.3 Incoming Subnet Model

Incoming subnet is shown in Figure 4.4. Transition *Ai* generates the packets ready to be sent by neighbor nodes to the current node and put the packets into place *WBi*. An inhibitor arc with cardinality *Ci* from *WBi* to *Ai* is still needed. After getting a buffer from shared place *Buffer,* transition *GBi* fires, and the token is transferred to place *IBi*. A token in place *IBi*, representing a packet received by the current node from its neighbors, is either destined to the current node, or has to be forwarded to other nodes.

1. If the packet has to be forwarded, immediate transition *Te* moves the token to place *RBo*, which means that the incoming packet becomes an outgoing one for the current node. It must be pointed out that a packet generated from *Ai* also has the possibility to be dropped out throughout the transmission. Thus, the incoming packet going through transition *Te* consequently goes through *DPNo* or *DPYo*.
2. If the packet's destination is the current node, immediate transition *Tl* moves the token to place *RRi*. Transition *RFi* represents the completion of receiving packets.

After its firing, one buffer in the current node is released by returning a token to place *Buffer*.

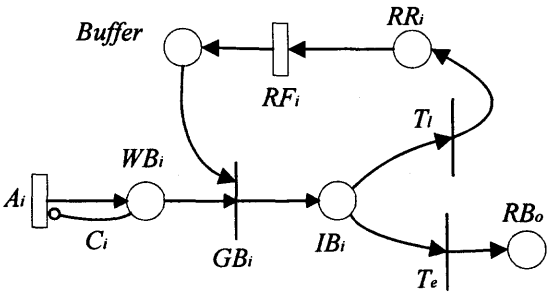

Figure 4.4 SPN incoming subnet: packet transfer from neighbor nodes to current node.

Once the SPN models are built up, the next question is to determine the parameters needed in the SPN model. The probability that a packet is forwarded or received by current node involves an approximation of transmission length. According to [Li et al., 2001] and [Gupta and Kumar, 2000], when the node density is constant, the probability density function (pdf), which means the probability of one node communicating with another node at distance x, is given by

$$p(x) = \frac{x}{\int_0^{\sqrt{A}} t\,dt} = \frac{2x}{A}$$

where A is a square network area and $\sqrt{A}$ is the maximum distance of A. Thus, the expected path length for a random traffic pattern is

$$\overline{L} = \int_0^{\sqrt{A}} x\,p(x)\,dx = \frac{2\sqrt{A}}{3}$$

Suppose that the nominal radio range for a wireless LAN is d. One has the average number of hops n required to send a packet from source to destination.

$$n = \bar{L}/d \tag{4.1}$$

Hence, a fraction *1/n* of the incoming packets is directed to the current node. Note that *n* is a variable depending on the variation of the network area.

After the average number of hops is determined, the firing rate of transition *Ai* can be easily derived by the product of the outgoing rate λ and the average hop number *n*, since the packet is delivered into neighbor nodes for each hop it takes.

4.2.4 Overall SPN Model

The composite SPN model is shown in Figure 4.5 resulting from a combination of outgoing and incoming subnets by merging shared places, *RBo* and *Buffer*. The meaning of the places and transitions in SPN is summarized in Table 4.1. The firing rate and probabilities of the transitions are given in Table 4.2. *#(p)* is defined as the number of tokens in place *p*. The priority of transitions depends on the definition. Higher priority is represented by a larger number. It is illustrated that *GBi*'s priority is higher than *GBo*'s to ensure that the delivery of packets in transmission takes priority over the injection of new packets into the network.

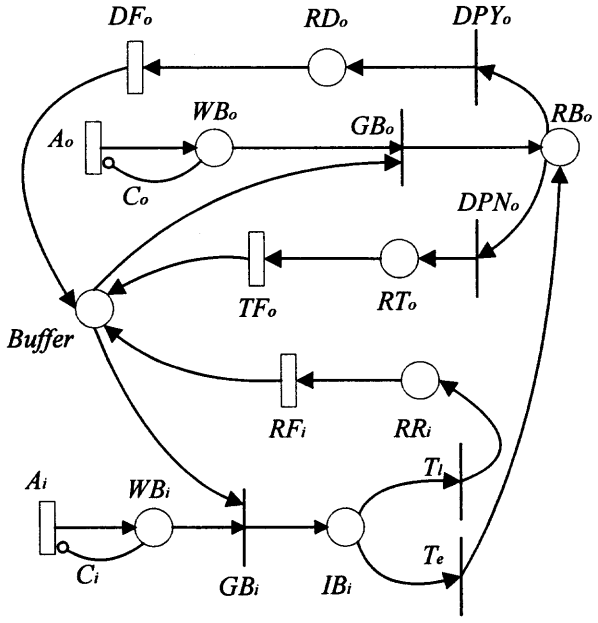

Figure 4.5 Overall SPN model.

The undefined parameter in Table 4.2 is x, the average time required by an outgoing packet to obtain and fill a buffer in the next node on its path. This process refers to one hop only. By summery, this work follows the rule that x has the same average as the time that a packet in place *WBi* must wait before it can obtain a local buffer slot and enter place *IBi*.

Hence, the following fixed-point iteration scheme is set up:

(1) Choose an initial guess $x(0)$ for x;

(2) Compute the successive values of x as:

$$x(i) = w$$

where w is the average waiting time and is obtained by using Little's Law:

$$w = \frac{E[\text{number of packets waiting }]}{E[\text{throughput of packets}]}$$

Table 4.1 Meaning of Places and Transitions in the SPN Model

Place	Meaning
Buffer	Buffer spaces available
WBo	Outgoing packets waiting for a buffer space
WBi	Incoming packets waiting for a buffer space
IBi	Incoming packet in the buffer
RBo	Outgoing packet that remains in the buffer
RTo	Ready to transmit an outgoing packet
RDo	Ready to drop an outgoing packet
RRi	Ready to receive an incoming packet
Transition	**Meaning**
Ao	Generate a packet that is to be transmitted
Ai	Externally generate a packet that enters the local node
GBo	Outgoing packet acquires a buffer space
GBi	Incoming packet acquires a buffer space
DPYo	Outgoing packet is going to be dropped
DPNo	Outgoing packet is going to be transmitted
Tl	The current node receives an incoming packet
Te	Forward an incoming packet to another node
TFo	Transmit an outgoing packet
DFo	Drop an outgoing packet
RFi	Receive an incoming packet

$$= \frac{E[\#(WBi)]}{E[rate(Ai)]}$$

(3) Stop the iterations when $x(i+1)$ and $x(i)$ are sufficiently close.

Table 4.2 Firing Rates and Probabilities of the Transitions in SPN Model

Transition	Firing rate
Ao	λ
Ai	$n\lambda$
TFo	$\#(RTo)/x$
DFo	$\#(RDo)\cdot x/n$
RFi	$\#(RRi)\cdot x/n$

Transition	Priority	Firing probability
GBo	4	1
GBi	5	1
DPYo	2	$1-\alpha$
DPNo	2	α
Tl	3	$\beta = 1/n$
Te	3	$1-\beta$

4.3 Numerical Results and Comparison

4.3.1 Network Parameter Setting

The average per flow throughput ζ can be defined as the number of packets received by the destinations divided by the number of connections, and average packet delay τ.

Because network overall throughput is affected by many factors, there's no uniform metrics for ad hoc networks. In the proposed model, ζ can be calculated as:

$$\zeta = E[rate(Ao)] \cdot \alpha$$

Rate(Ao) is not a constant despite that its firing rate is defined. Its value changes within each iteration. τ is defined as the average time elapsing from the instant a packet is generated by its source node (firing of transition *Ao*), to the instant it is read by its destination node (firing of transition *RFi*). In the model of Figure 4.5, this is obtained as the sum of three components:

$$\tau = t_1 + t_2 + t_3 \tag{4.2}$$

(1) The average time a packet waits before it is put into a buffer in the current node. It is computed using Little's law:

$$t_1 = \frac{E[\#(WBo)+\#(RBo)]}{E[rate(Ao)]} \tag{4.3}$$

(2) The average time a packet waits before it is removed from the buffer in the destination node:

$$t_2 = \frac{E[\#(RRi)]}{E[rate(RFi)]} \tag{4.4}$$

(3) The average time a packet passes through intermediate nodes. It equals the product of transmission time x and average number of hop n:

$$t_3 = x \cdot n \tag{4.5}$$

A parameter named delivery ratio is defined, which is α in SPN model. α refers to the packet delivery fraction in performance metrics of an ad hoc network.

A detailed simulation model based on the latest ns2 version is also used to establish the believability of the proposed SPN mehtod. The research adopts AODV and

DSR protocols to deal with a routing problem. Both are on-demand protocols. The detailed description of two protocols is in Chapter 2.

In the experiment, a random waypoint model is used to generate a node mobility model. 802.11 for wireless LANs is used as a shared-media radio with a nominal bit-rate of 2 Mb/sec and a nominal radio range of 250 meters. An ad hoc network in two fields with dimensions 670m × 670m and 1000m × 1000m is used. Thus, from Eq. (4.1), n is 2.5 and 3.77, respectively in the experiment. β can be derived from n. One cannot use a fraction in simulation because none of the packets can be divided into several parts. Fraction can be used in an analytical model to describe the proposed SPN model more accurately.

The proposed fixed-point scheme converged only in a few iterations. Conditions for the existence of a fixed point may be found in [Mainkar and Trivedi, 1996]. The proof of a unique solution is considered as a harder problem that will be addressed in the future research. Comparing to the time consuming simulation, the fixed-point scheme's cost is negligible.

The traffic sources are CBR, i.e., constant bit rate. CBR/UDP (User Datagram Protocol) traffic flows are used as data packets during the performance analysis process instead of TCP traffic because of the TCP's poor performance in ad hoc networks. The performance of TCP in ad hoc networks can be seen in [Günes and Vlahovic, 2002; Holland and Vaidya, 1999; Sun and Man, 2001]. Figure 4.6 shows the TCP connection in simulation. The delivery ratio of typical TCP connection is either very high around 0.9, or very low, nearly zero. This is because TCP intends to provide a reliable connection and has been optimized for fixed networks. Due to the high node mobility, an ad hoc

network topology changes continuously. Current TCP implementations cannot guarantee a stable performance because they lack the ability to distinguish packet losses caused by network congestions or route failures. Because the intercurrent status of TCP are unable to be well studied, CBR traffic is used.

Figure 4.6 TCP delivery ratio.

There are 30 nodes roaming in this area with zero pause time (constant mobility). The speed of a node varies from 0 m/s to 20m/s to change mobility. The source-destination pairs are spread randomly throughout the network. Only 512 byte data packets are used. The number of sessions increases along with the number of nodes. All traffic sessions are established at random times and stay active until the end. Similar simulation environment was used before in several recent performance studies on ad hoc networks [Broch et al., 1998; Perkins et al., 2001]. However, most of the simulations before this work keep a relative low traffic to maintain a high successful packet delivery ratio. For instance, the packet rate varies between 2 and 4 packets/s [Perkins et al., 2001], and up to 8 packets/s [Broch et al., 1998]. In order to explore the throughput and latency in low delivery ratio situation, more than 10 packets are generated and transmitted in one

second in this simulation. Broch (1998) and Perkins (2001) evaluated routing protocols under different pause time, that is, nodes stay motionless until pause time. In SPN, the node is keep moving constantly from very beginning because continuous moving reflects more accurately the high mobility. Note that the high mobility is a basic characteristic of an ad hoc network.

The delivery ratio in SPN can be specified. Note that this value can be acquired only from statistics and calculation in simulation. Hence the delivery ratio value from SPN has a small difference compared with the data from ns2.

4.3.2 Result Comparison

Figures 4.7 and 4.8 show the result of average per flow throughput as a function of delivery ratio α under two different fields, respectively. It is shown that a close match exists between the obtained results and those from a simulation model in ns2. The maximum error between SPN and two routing protocols is below 9% for both fields. The SPN model well represents the per flow throughput varying with delivery ratio. These two figures show that no matter what routing protocol is using, their per flow throughput keeps a linear relationship with delivery ratio. There is no noted difference between two different protocols. The experiment shows that even packet drop probability is low, which means most generated packets can arrive at their destinations successfully, the system throughput, usually under 0.05Mb/s, is surprisingly low compared to the channel capacity 2Mb/sec. This result is concurrent with the research results in [Li et al., 2001]. If one wants to improve network performance by increasing the sending rate of per flow, delivery ratio drops quickly.

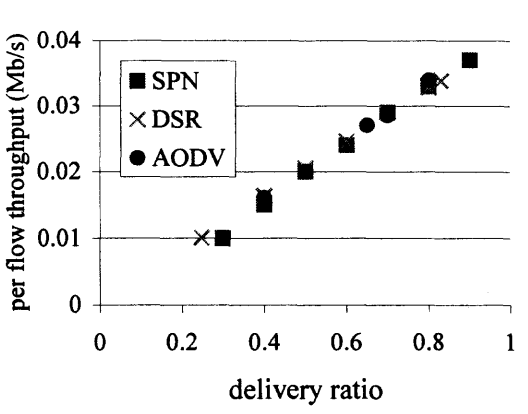

Figure 4.7 Delivery ratio vs. throughput with 670m × 670m.

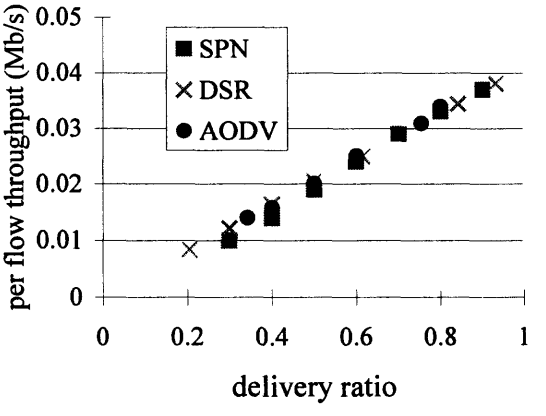

Figure 4.8 Delivery ratio vs. throughput with 1000m × 1000m.

Figures 4.9 and 4.10 show the latency varying with α. Delay time becomes longer as the number of drop packets increases. Latency becomes quite high if α is less than 0.6 that is unacceptable. There's a difference between two protocols because they use different schemes to deal with latency. The overhead of DSR is potentially larger than that of AODV since each DSR packet have to carry full routing information, whereas in AODV packets need only contain the destination address. Similar settings apply to route reply and memory overhead. Furthermore, DSR is not scalable to large networks. Combining these features, the AODV's overall performance is better than DSR's. When delivery ratio is larger than 0.5, SPN shows a good match to the simulation results. The maximum error is below 6%. When delivery ratio is smaller than 0.5, the proposed model's maximum error with actual protocols becomes high. At few points, like 0.4 in abscissa, error can be 20%. This is because the latency is too high to be acquired accurate and believable. It is a reference value since no application allows such high drop packet rate. However, the model shows the basic varying trend of the relationship between

latency and delivery ratio that exists in both protocols. When delivery ratio decreases, Latency's increase behaves like a parabola rather than linear increase. The proposed model can be improved to obtain a more accurate solution to represent this parameter.

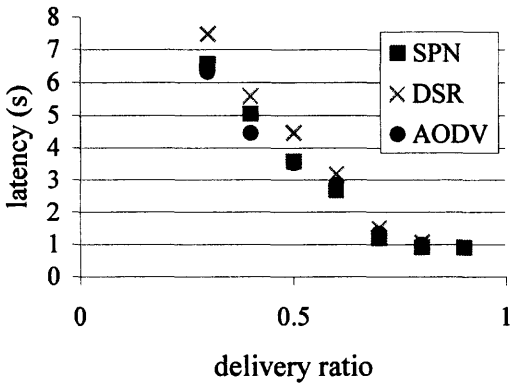

Figure 4.9 Delivery ratio vs. latency with 670m × 670m.

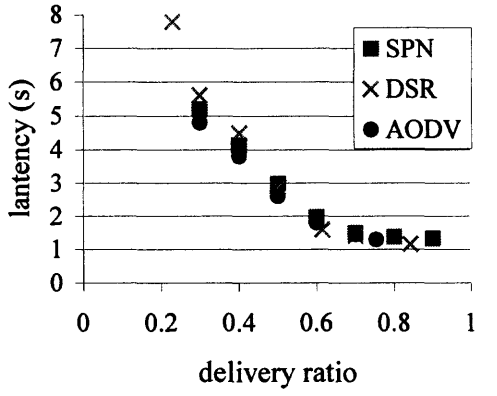

Figure 4.10 Delivery ratio vs. latency with 1000m × 1000m.

4.4 Summary

In this chapter, a stochastic Petri net model is developed to represent an ad hoc network. The proposed scheme provides a customizable approach to analyze the characteristics and performance of the system. It is shown that a close match exists between the obtained results and those from a simulation model in ns2. The proposed scheme costs negligible computational effort compared with that of a simulation method. While SPN model can give a theoretical solution for ad hoc network, ns2 is only used as a detailed model. Because ns2 has become a standard simulation tool in network research, the comparison gives us a link between Petri nets and detailed discrete event systems.

Some characteristics can be obtained by SPN with slight modification of the proposed model. All of the time delays attached with transitions in the proposed model are approximated with exponential distributions. This is not always true in a real system.

For instance, sometimes delays are constants. One can apply Erlang distributions with a given mean in the SPN model to approximate the constant distribution. That increases the computation complexity but can improve the model with better practicability.

Currently, the proposed SPN model cannot reflect a specific routing protocol's characteristics. Thus, it is a generalized model. In order to better analyze the routing protocols under different situation, additional places and transitions must be considered to represent those parameters. Such changes shall make the model more applicable.

Network security is an important issue in the current ad hoc network research. Several aspects have been stressed including routing protocols, authentication, access control, quality of service (QoS), etc. The next chapter models the "Security Level" concept in Chapter 3 into Petri nets to analyze corresponding ad hoc network's changes and performance.

CHAPTER 5

STOCHASTIC PETRI NET MODELING OF SECURITY

In Chapters 3 and 4, the performance enhanced secure ad hoc on-demand routing protocol (SOR) and stochastic Petri nets model of ad hoc network are proposed to analyze the network performance. SPN model has been proved to well represent the characteristic of an ad hoc network. This chapter wants to analyze the SOR's performance, especially the security performance, by analytical SPN model.

In order to embed "Security Level" model in SPN, multiplicity of an arc of PN is used to represent multipath selection. An arc with multiplicity *i* from place to transition means that, to fire the transition, at least *i* tokens are needed in the place. An arc with multiplicity *i* from transition to place means that, after firing the transition, there are *i* tokens going into the place. The use of arc multiplicity in PN can well represent the "Security Level" concept. In Figure 3.1, the process of node A dividing and transferring messages to neighbor nodes 1, 2, and 3 can be illustrated by a arc with multiplicity three from a place to a transition. After firing the transition, three tokens are moved to next place in the SPN model. The process of node B integrating messages $PS(m, s_1)$ and $PS(m, s_3)$ can be illustrated by a arc with multiplicity two from transition to place. Two tokens are needed to fire the transition.

Consider measuring security criteria in SPN model. One cannot assign random number in SPN model as we did in Chapter 3, which means the attack frequency of a compromised node must be determined before SPN simulation begins. At the same time, attack frequency, proportion and communication frequency of different security levels

can be viewed as a variant of SPN model, as delivery ratio α in Chapter 4, to explore the relationship between attack effects and security levels.

The remaining of this chapter is organized as follows. Section 5.1 adds multipath parameter into the SPN model. Section 5.2 integrates security measurement into the scheme. Section 5.3 compares the numerical results of the modified SPN model with simulation results. Section 5.4 gives the summary.

5.1 Multipath Parameter

In order to incorporate the multipath characteristic into the previous proposed stochastic Petri net model and analyze the SOR's effect on ad hoc network, Chapter 4's SPN model is modified as follows.

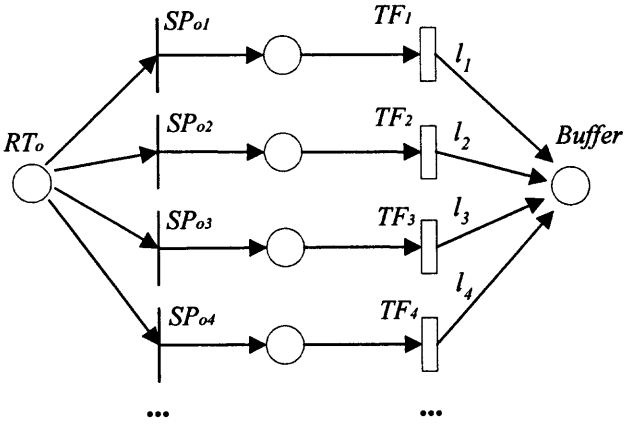

Figure 5.1 SPN multipath outgoing part model.

Multipath parameter adding means the "Security Level" concept's implementation in SPN. Figure 5.1 represents a multipath outgoing part model derived from Figure 4.3. Suppose that there are n security levels in an ad hoc network. Each security level i in SOR has a corresponding number l_i, which means there are l_i paths for

each level. After firing the timed transition *TFi*, l_i tokens are moved from place *RTo* to place *Buffer*. Such process can be defined as:

$$O(Buffer, TF_i) = l_i \qquad i = 1, \ldots, n$$

where *O(p, t)* is defined as the multiplicity of a directed arc from transition *t* to place *p* in Chapter 4.

After defining the proportion of each security-level node in ad hoc network and packet rates of different level nodes, the probability that fires the transition SP_{ok} is defined as:

$$p(SP_{ok}) = p_k \qquad k = 1, \ldots, n \tag{5.1}$$

where p_k can be calculated as:

$$\begin{cases} p_k = \dfrac{r_k \cdot \lambda_k}{\sum_{i=1}^{n} r_i \cdot \lambda_i} & k = 1, \ldots, n \\ \sum_{k=1}^{n} p_k = 1 \end{cases}$$

where r_i is the proportion of each security level and λ_i is the packet arrival rates of different levels. The summary of all p_k should equal one.

A two-level multipath outgoing subnet sample is shown in Figure 5.2. l_1 is equal to one and l_2 is equal to three. A new place *MPo* is defined to represent the beginning of outgoing packet multipath selection. After firing the timed transition TF_1 or TF_2, one or three tokens are released to *Buffer*, respectively. l_i is assigned during the initialization of the model and cannot been changed in the simulation.

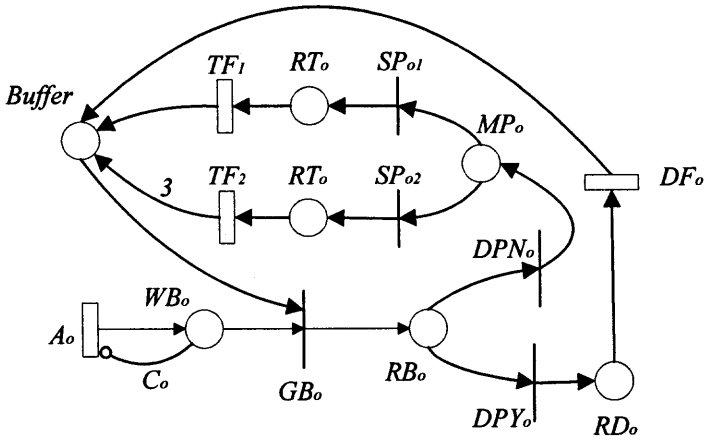

Figure 5.2 SPN 2-level multipath outgoing subnet.

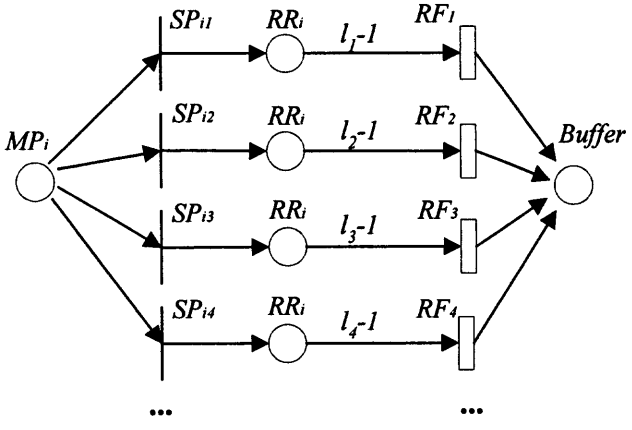

Figure 5.3 SPN multipath incoming part model.

In correspondence with Section 3.4.2, the simplified (3, 2) threshold cryptography scheme is used to construct the model as an example. Figure 5.3 represents a multipath incoming part model derived from Figure 4.4. The settings are similar to the outgoing subnet. Multiplicity arc is defined as:

$$I(RR_i, RF_i) = \begin{cases} 1 & l_i = 1 \\ l_i - 1 & l_i > 1 \end{cases} \qquad i = 1, \ldots, n$$

where *I(p, t)* is defined as the multiplicity of a directed arc from place p to transition t in Chapter 4.

l_i-1 means that, to fire timed transition *RFi*, at least l_i-1 tokens are required to be in place *RRi* if l_i is greater than one. The physical meaning of the process from *RRi* to *Buffer* is that, one data packet is assumed to transmit successfully after any l_i-1 pieces arrive at the destination node. A new place *MPi* is defined to represent the beginning of incoming packet multipath selection. The probability that fires the transition P_{ik} is obtained as:

$$p(SP_{ik}) = p_k \qquad k = 1, ..., n \tag{5.2}$$

where p_k is equal to p_k in Equation (5.1).

A two-level multipath incoming subnet sample, corresponding to Figure 5.2, is shown in Figure 5.4. One or two tokens are needed to fire timed transition RF_1 or RF_2, respectively.

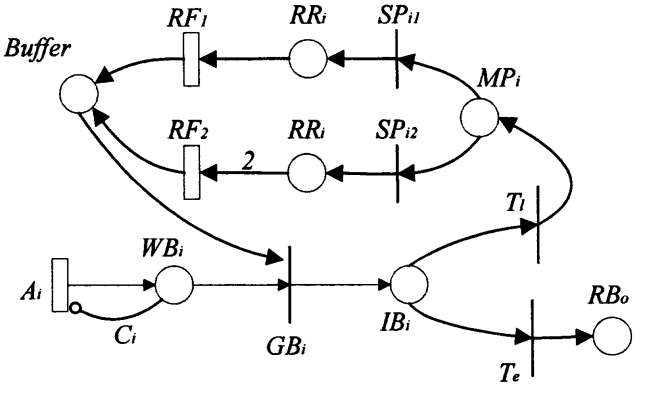

Figure 5.4 SPN 2-level multipath incoming subnet.

The modified SPN model incorporated with outgoing and incoming subnets is shown in Figure 5.5. The meaning of places and transitions is summarized in the next

section. In order to maintain the balance of tokens, some additional places, which are irrelevant to the underlying network, need to be added during the simulation. However, this current model does not simulate the detailed situations including several broken paths. It rather considers the ad hoc network under an ideal running condition, i.e., no broken path. This is one limitation of this model.

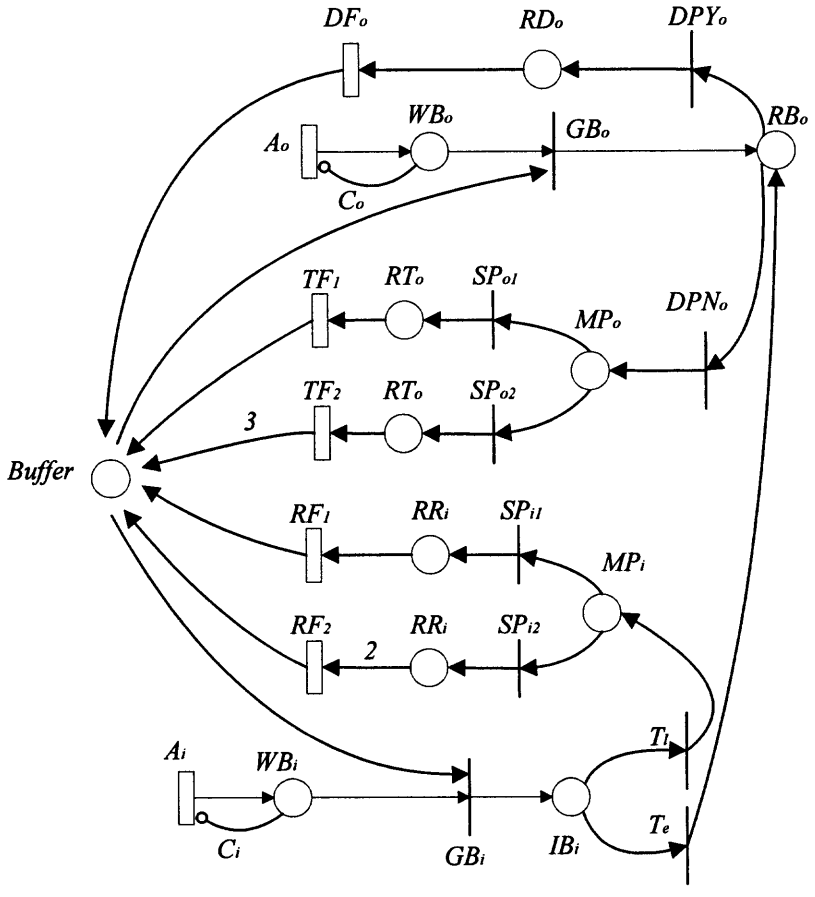

Figure 5.5 SPN 2-level multipath model.

5.2 Security Measurement

As done in Chapter 3, a "hypothetical" enemy is constructed. The enemy node can be inserted either in the outgoing subnet or incoming subnet. For simplicity, the compromised node is placed in the incoming subnet.

Figure 5.6 shows the SPN incoming part model with a compromised node. Only eavesdropping is considered because blocking has a less potential effect than eavesdropping on leaking packets, which was analyzed in Chapter 3. Suppose that the hypothetical enemy can eavesdrop a certain portion packets of ad hoc network through various nodes during the simulation, no matter the security level. When an incoming packet arrives at place *MPi*, it either enters the regular multipath selection, or is eavesdropped by a compromised node. Transition *CEi* moves a token to a compromised node place *CN*. The probability, denoted by ω, that a token enters transition *CEi* is a variable in SPN model. Corresponding to *CEi*, Equation (5.2) has to be modified as:

$$p(SP_{ik}) = p_k \cdot (1-\omega) \qquad k = 1, ..., n \qquad (5.3)$$

where p_k is equal to p_k in Equation (5.1).

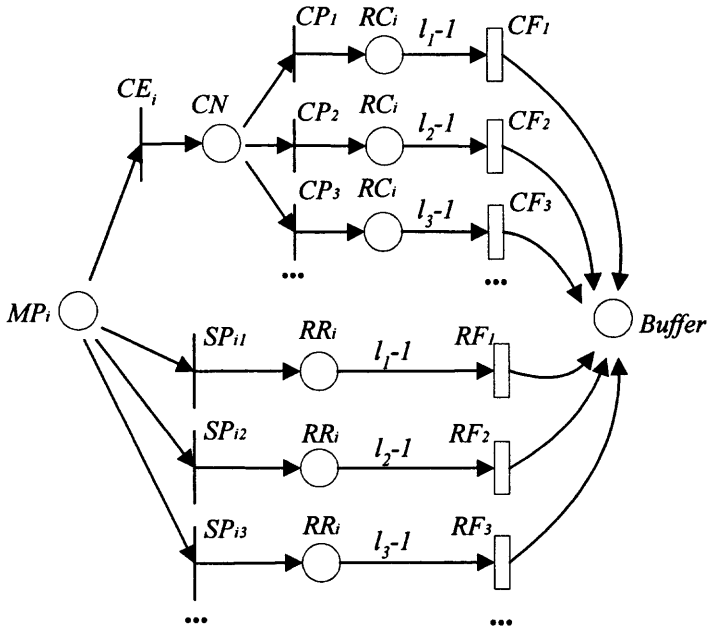

Figure 5.6 SPN incoming part model with compromised node.

Tokens in place *CN* do not mean that the packet information has been leaked out. For the i^{th} level security node, information is leaked out only if there are l_i-1 places in *RCi*

to fire transition *CFi*. Thus, the rule of token flow after place *CN* is similar to that of Figure 5.3, where

$$I(RC_i, CF_i) = \begin{cases} 1 & l_i = 1 \\ l_i - 1 & l_i > 1 \end{cases} \qquad i = 1, \ldots, n \qquad \text{and}$$

$$p(CP_k) = p_k \qquad k = 1, \ldots, n \qquad (5.4)$$

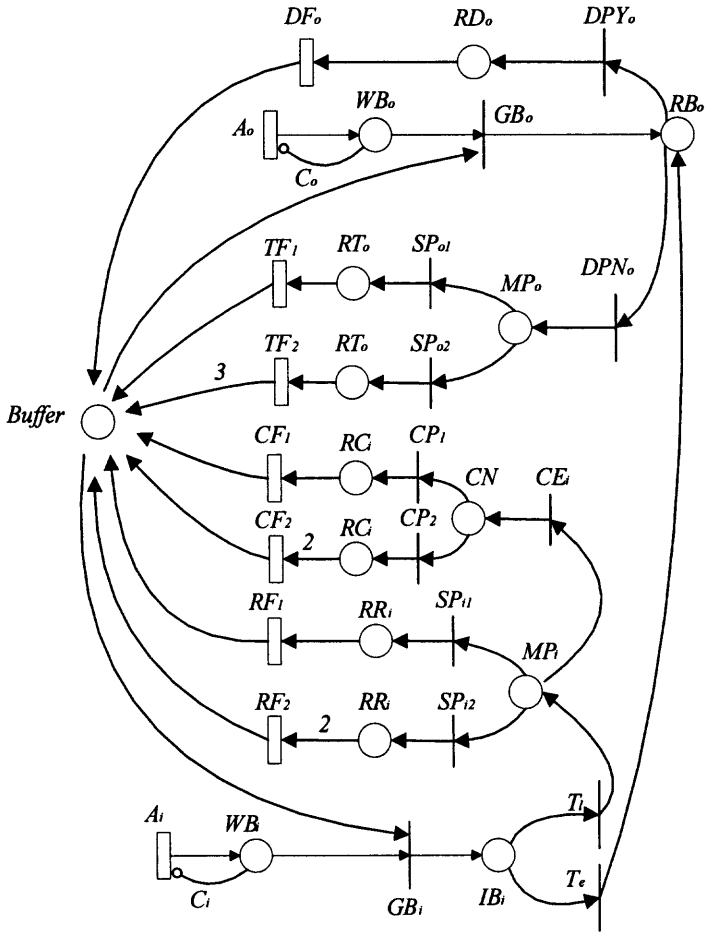

Figure 5.7 Overall SPN model.

The overall SPN model incorporated with multipath selection and security measurement is shown in Figure 5.7. The firing rate and probabilities of the transitions are given in Table 5.1. The meaning of the places and transitions in SPN is summarized

in Table 5.2. Some places and transitions can be merged together. For the purpose of the physical meaning, they keep unchanged.

The fixed-point iteration is the same as in Chapter 4.2.4 because the entrance of SPN model is not changed.

Table 5.1 Firing Rates and Probabilities of the Transitions in Figure 5.7

Transition	**Firing rate**	
Ao	λ	
Ai	$n\lambda$	
TFi	$\#(RTo)/x$	
DFo	$\#(RDo)\cdot x/n$	
RFi	$\#(RRi)\cdot x/n$	
CFi	$\#(RCi)\cdot x/n$	
Transition	**Priority**	**Firing probability**
GBo	4	1
GBi	5	1
DPYo	2	$1-\alpha$
DPNo	2	α
T_l	3	$\beta = 1/n$
T_e	3	$1-\beta$
SPik	4	Equation (5.3)
SPok	4	Equation (5.1)
CEi	5	ω
CPi	5	Equation (5.4)

Table 5.2 Meaning of Places and Transitions in Figure 5.7

Place	Meaning
Buffer	Buffer spaces available
WBo	Outgoing packets waiting for a buffer space
WBi	Incoming packets waiting for a buffer space
IBi	Incoming packet in the buffer
RBo	Outgoing packet that remains in the buffer
RTo	Ready to transmit an outgoing packet
RDo	Ready to drop an outgoing packet
RRi	Ready to receive an incoming packet by normal node
MPo	Ready to select multipath for an outgoing packet
MPi	Ready to select multipath for an incoming packet
RCi	Ready to receive an incoming packet by compromised node
CN	Incoming packet compromised by an enemy
Transition	**Meaning**
Ao	Generate a packet that is to be transmitted
Ai	Externally generate a packet that enters the local node
GBo	Outgoing packet acquires a buffer space
GBi	Incoming packet acquires a buffer space
DPYo	Outgoing packet is going to be dropped
DPNo	Outgoing packet is going to be transmitted
T_l	The current node receives an incoming packet
T_e	Forward an incoming packet to another node
TFi	Transmit an outgoing packet
DFo	Drop an outgoing packet
RFi	Receive an incoming packet
CEi	Eavesdrop an incoming packet
CPk	Compromised packet going into k^{th} security level
SPok	Outgoing packet going into k^{th} security level
SPik	Incoming packet going into k^{th} security level
CFi	Receive an compromised packet

5.3 Simulation and Comparison

There are five parameters in the SPN model, α, β, ω, r, and λ. First, multipath performance is studied. ω is set to zero to eliminate the effect of any compromised node. The setting of ns2 for SOR is similar to those in Chapter 3. An ad hoc network with dimension 670m × 670m is used. β is 0.4 from Table 5.2. There are 30 nodes roaming in this area with zero pause time (constant mobility). The speed of a node varies from 0 m/s to 20m/s to change mobility. The source-destination pairs are spread randomly throughout the network. The traffic sources are CBR. Only 512 byte data packets are used. The security level setting is the same as that in Figure 5.7. λ is a constant for different security level for simplicity. The average packet delay τ is calculated almost the same as Equation (4.2), in which (4.4) should be modified as:

$$t_2 = \sum_{i=1}^{n} \left(\frac{E[\#(RRi)]}{E[rate(RFi)]} \cdot p_k \cdot (1-\omega) \right)$$

In Figure 5.8, average packet delay τ varies as a function of proportion of the 1st security level r_1. Since we only have two security levels, r_2 is equal to $1\text{-}r_1$. r_1 equals 1.0 means that all of the nodes in ad hoc network belong to the first security level.

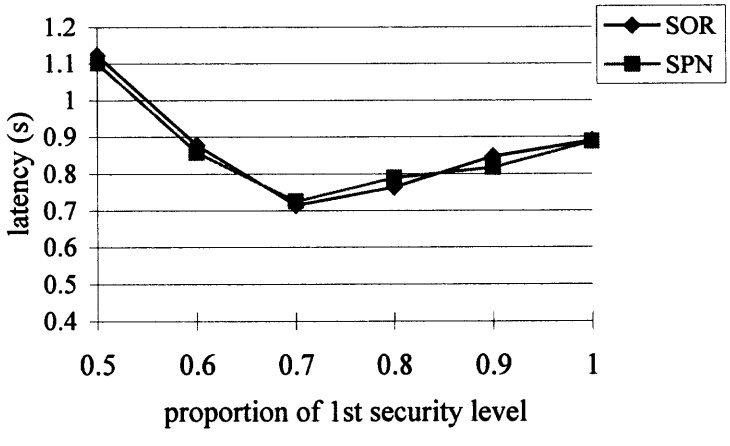

Figure 5.8 Proportion of 1st security level vs. latency.

Delay time does not increase along with the increase of the proportion of the 1st security level nodes. SPN model shows the consistency with the simulation result of SOR. The difference between SPN and SOR is below 8%. Figure 5.8 proves from another aspect that, the proposed multipath routing model can significantly reduce the latency, comparing to that in Figure 3.10.

Consider the internal enemy attack condition. Since firing transition *CFi* means that a packet has been leaked out by a compromised node, the proportion of leaking packets shown in the SPN model can be calculated as:

$$\frac{\sum_{i=1}^{n} E[rate(CFi)]}{\sum_{i=1}^{n} E[rate(RFi)] + \sum_{i=1}^{n} E[rate(CFi)]}$$

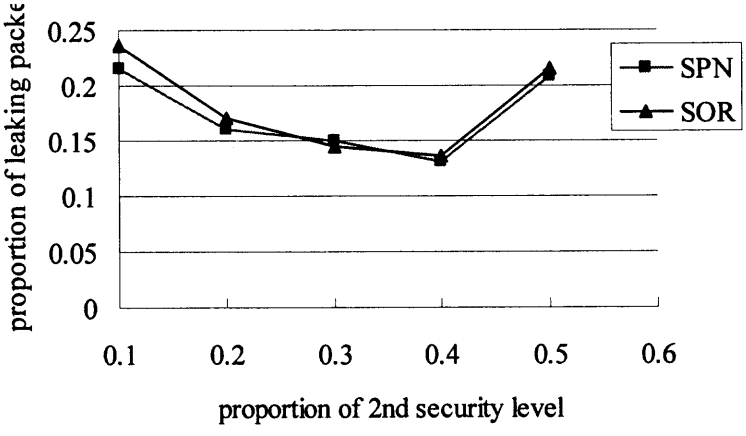

Figure 5.9 Proportion of leaking packets.

Figure 5.9 shows the ratio of leaking packets to those generated by the CBR. The proportion of the 2nd security level is adopted corresponding to Figure 3.12. The data of SOR is acquired from Chapter 3. It is shown that a close match exists between the SPN and SOR. The difference between SPN and SOR is below 10%. When the proportion of

2^{nd} security level nodes varies from 20% to 40%, one can keep over 80% information safe. Hence, the SPN model proves that SOR is effective against the attack of eavesdropping.

5.4 Summary

In this chapter, the proposed stochastic Petri net model is improved to represent performance enhanced secure ad hoc on-demand routing protocol (SOR). Security level and detailed security measurement are embedded into the proposed SPN model as a parameter. Multipath parameter can be represented by arc multiplicity in Petri nets. This chapter gives a quantificational measurement to analyze the performance of modified SPN model under the effect of multipath and attack of a hypothetical compromised node. Simulation results show that the proposed SPN model well represents the characteristic of SOR.

The proposed SPN model is an analytical model. It approximates and generates a proper amount of traffic going through one node in a network of a particular size. It is difficult to simulate many mobile nodes in a SPN model. Colored Petri Nets is a feasible alternate tool for future research of ad hoc network.

CHAPTER 6

CONCLUSIONS AND FUTURE RESEARCH

6.1 Conclusions

This doctoral dissertation work proposes a performance enhanced Secure ad hoc On-demand Routing protocol (SOR). It provides a customized stochastic Petri net-based approach to modeling and analyzing ad hoc networks and their related performance and security issues.

6.1.1 Summary of Contributions

The contributions of this dissertation are summarized into three aspects:

1) Performance enhanced secure ad hoc on demand routing protocol (SOR):

   A novel ad hoc on demand routing protocol is proposed. It is embedded with "*Security Level*" concept and "*maximum hopcount*" to restrict the number of routing packets in a given area. The proposed scheme provides customizable security to the flow of routing protocol messages themselves. The performance of SOR relative to AODV is studied under a wide range of traffic scenarios. SOR offers a significant reduction in average packet delay and provides up to about 30% reduction in routing load. In general, SOR offers an alternative way to implement excellent security in an on-demand routing protocol.

2) Stochastic Petri net Model:

   A stochastic Petri net model is proposed to represent an ad hoc network. The proposed scheme provides a customizable approach to analyze the characteristics and performance of the system. It is shown that a close match exists between the obtained

results and those from a simulation model in ns2. The proposed scheme costs negligible computational effort compared with that of a simulation method. While SPN model can give a theoretical solution for ad hoc network, ns2 is used to model a detailed model. Because ns2 has become a standard simulation tool in network research, the comparison gives us a link between Petri nets and detailed discrete event systems.

3) Analysis of security improvement:

   Based on the previous stochastic Petri nets model, this research improves it to accommodate SOR characteristics. Security level and detailed security measurement are embedded into the SPN model as a parameter. Multipath parameter can be represented by arc multiplicity in Petri nets. This research gives a quantificational measurement to analyze the performance of the modified SPN model under the effect of multipath and attack of hypothetical compromised nodes. Simulation results show that SPN model well represent the characteristic of SOR.

6.1.2 Limitations

This research has the following limitations:

1) In the performance enhanced secure ad hoc on demand routing protocol, the moving area of mobile nodes is given in advance, which is uncertain in a real environment.
2) In stochastic Petri nets model, all of the time delays attached with transitions in the proposed model are approximated with exponential distributions. This is not always true in a real system. For instance, sometimes delays are constants.

3) In security modeling and analysis, the proportion and communication frequency of different security level nodes are assumed known and unchanged during the simulation. This research does not consider the changes between different security level nodes. Moreover, current model does not simulate the detailed situations including several broken paths. It rather considers the ad hoc network under an ideal running condition.

6.2 Future Research

There are several ways in which this work could be extended in the future. Some important and promising directions are listed as follows:

1) The practicability and security of ad hoc networks is still under research. No routing protocol can outperform all the others. Ad hoc network is most likely to be applied to some specific fields, for example, battlefield. Under such situation, the security issue is more important than wired network and should be stressed. Some results in SOR are based on the result of simulation. The theoretical solution of the relationship between maximum security and proportion of different security level nodes requires further research. Quality of service in multipath routing is also worthy for more investigation.
2) The proposed SPN model is an analytical model. It approximates and generates a proper amount of traffic going through one node in a network of a particular size. Colored Petri Nets should be developed to represent a more detailed model. However, because of the limitation of Petri nets, it is difficult to simulate a large number of mobile nodes in Petri nets model, like what was did in ns2. Hence, using a analytical

model to analyze the network performance is a feasible research issue based on the advantage of Petri nets.

3) Erlang distributions with a given mean can be applied in the Petri net model to approximate the constant distribution. That will increase the computational complexity but can improve the proposed model with better practicability.

4) The manner of attack by enemy has been described in Chapter 2. How to quantify these attacks needs future development. For example, how to distinguish between a compromised node and a link-lost node is still undefined. If this node moves into a linkable area, should we establish a new link to it or treat it as a compromised node again? More security characteristics need to be quantified and clarified.

REFERENCES

1. Ahuja A., Agarwal S., Singh J. P., and Shorey R., "Performance of TCP over Different Routing Protocols in Mobile Ad-Hoc Networks," in *IEEE 51st Vehicular Technology Conf. Proc.*, vol. 3, pp. 2315-2319, May 2000.

2. Ajmone-Marsan M., Conte G., and Balbo G., "A Class of Generalized Stochastic Petri Nets for the Performance Evaluation of Multiprocessor Systems," *ACM Trans. on Computer Systems*, 2(2), pp. 93-122, May 1984.

3. Algesheimer J., Cachin C., Camenisch J., and Karjoth G., "Cryptographic Security for Mobile Code," in *IEEE Symp. on Security and Privacy* pp. 2-11, 2001.

4. Ayanoglu E., Chih-Lin I., Gitlin R. D., and Mazo J. E., "Diversity Coding for Transparent Self-Healing and Fault-Tolerant Communication Networks," *IEEE Trans. on Comm.*, vol. 41, no. 11, pp. 1677-1686, Nov. 1993.

5. Aydos M., Yantk T., and Koc C. K., "A High-speed ECC-based Wireless Authentication on an ARM Microprocessor," in *16th Annual Conf. on Computer Security Applications (ACSAC '00),* pp. 401-409, Dec. 2000.

6. Basagni S., Chlamtac I., and Syrotiuk V., "Dynamic Source Routing for Ad Hoc Networks Using the Global Positioning System", in *Proc. IEEE WCNC,* pp. 301-305, 1999.

7. Basagni S., Chlamtac I., Syrotiuk V., and Woodward B. A., "A Distance Routing Effect Algorithm for Mobility (DREAM)," *Proc. MobiCom'98*, pp. 76-84, Oct. 1998.

8. Beritelli F., Cola E. Di, Fortuna L., and Italia F., "Multilayer Chaotic Encryption for Secure Communications in Packet Switching Networks," in *WCC-ICCT Intl. Conf. on Comm. Technology Proc.*, pp. 1575-1582, Aug. 2000.

9. Bertsekas D. and Gallager R., *Data Network*, Second Ed., Prentice Hall, Inc., pp. 404-410, 1992.

10. Beznosov K., Deng Y., et al., "A Resource Access Decision Service for CORBA-based Distributed Systems," in *Proc. Annual Computer Security Applications Conf.*, pp. 310-319, 1999.

11. Binkley J., "Authenticated Ad Hoc Routing at the Link Layer for Mobile Systems," *Wireless Networks*, vol. 7, issue 2, pp. 139-145, 2001.

12. Boukerche A., "Performance Comparison and Analysis of Ad Hoc Routing Algorithms," in *20th IEEE Intl. Performance, Computing, and Comm. Conf.*, pp. 171-178, April 2001.

13. Broch J., Maltz D., Johnson D., Hu Y. C., and Jetcheva J., "A Performance Comparison of Multi-Hop Wireless Ad Hoc Network Routing Protocols," in *The 4th annual ACM/IEEE Intl. Conf. on Mobile computing and networking*, pp. 85-97, 1998.

14. Cano J. C. and Manzoni P., "A Performance Comparison of Energy Consumption for Mobile Ad Hoc Network Routing Protocols," in *8th Intl. Symp. on Modeling, Analysis and Simulation of Computer and Telecommunication Systems*, pp. 57-64, Aug. 2000.

15. Celebi E., "Performance Evaluation of Wireless Mobile Ad Hoc Network Routing Protocols," *http://www.cmpe.boun.edu.tr/~emre/research/msthesis/*, 2001.

16. Chen D., Soong B. and Trivedi K., "Optimal Call Admission Control Policy for Wireless Communication Networks," in *Proc. Intl. Conf. on Information, Communication and Signal Processing (ICICS)*, Singapore, Oct., 2001.

17. Chen W., Jain N., and Singh S., "ANMP: Ad Hoc Network Management Protocol," *IEEE Journal on Selected Areas in Comm.*, vol. 17, no. 8, pp. 1506-1531, Aug. 1999.

18. Chiasserini C. F. and Rao R. R., "A Distributed Power Management Policy for Wireless Ad Hoc Networks," in *IEEE Wireless Comm. and Networking Conf.*, vol. 3, pp. 1209-1213, 2000.

19. Chiou G. H. and Chen W. T., "Secure Broadcasting Using Secure Lock," *IEEE Trans. Software Eng.*, vol. 15, pp. 929-933, Aug. 1989.

20. Ciardo G. and Trivedi K. S., "A Decomposition Approach for Stochastic Petri Net Models," *Performance Evaluation*, 18(1), pp. 37–59, July 1993.

21. Ciardo G., Cherkasova L., Kotov V., and Rokicki T., "Modeling a Scalable High-Speed Interconnect with Stochastic Petri Nets," in *Proc. Intl. Workshop on Petri Nets and Performance Models (PNPM'95)*, pp. 83-92, Durham, NC, Oct. 1995.

22. Ciardo G., Trivedi K., and Muppala J., SPNP: Stochastic Petri Net Package, in *Proc. 3rd Intl. Workshop on Petri Nets and Performance Models (PNPM'89)*, pp. 142-151, Kyoto, Japan, Dec. 1989, IEEE Computer Society Press.

23. Corson M., Macker J., and Cirincione G., "Internet-based Mobile Ad Hoc Networking," *IEEE Internet Computing,* vol. 3, no. 4, pp. 63-70, July-Aug. 1999.

24. Deng Y., Wang J., and Tsai J. J. P., "Formal Analysis of Software Security System Architectures," in *5th Intl. Symp. on Autonomous Decentralized Systems*, pp. 426-434, March 2001.

25. Desmedt Y. G., "Threshold Cryptography," *European Trans. on Telocommunications* , vol. 5, no. 4, pp. 449-457, 1994.

26. Desmedt Y. G., "Threshold Cryptosystems," *Euro. Trans. Telecommunication*, vol. 5, no. 4, pp. 307-315, July-Aug. 1994.

27. Dugan J. B., Trivedi K. S., Geist R. M., and Nicola V. F., "Extended Stochastic Petri Nets: Applications and Analysis," in E. Gelenbe, editor, *Performance '84*, pp. 507-519. Elsevier Science Publishers B. V. (North-Holland), Amsterdam, Netherlands, 1985.

28. Encapsulating Security Payload, Internet draft, *http://www.faqs.org/rfcs/rfc1827-.html*, 1995.

29. Fall K. and Varadhan K. (Editors). Ns Notes and Documentation. The VINT Project, UC Berkeley, LBL, USC/ISI, and Xerox PARC, Nov. 1997, available from *http://www-mash.cs.berkeley.edu/ns/*

30. Freudenthal M., Heiberg S., and Willemson J., "Personal Security Environment on Palm PDA," in *16th Annual Computer Security Applications Conf.*, pp. 159-167, Dec. 2000.

31. Fumy W. and Landrock P., "Principles of Key Management," *IEEE Journal on Selected Areas in Comm.*, vol. 11, issue 5, pp. 785-793, June 1993.

32. Golle P. and Modaduge N., "Authenticating Streamed Data in the Presence of Random Packet Loss," in *ISOC Network and Distributed System Security Symp.*, pp. 13-22, 2001.

33. Günes M. and Vlahovic D., "The Performance of the TCP/RCWE Enhancement for Ad Hoc Networks," in *Proc. of the 7th IEEE Symposium on Computers and Comm., ISCC 2002*, pp. 43-48, Italy, July 2002.

34. Gupta V. and Montenegro G., "Secure and Mobile Networking," *Mobile Networks and Applications*, vol. 3, pp. 381-390, 1998.

35. Gupta P. and Kumar P. R., "The Capacity of Wireless Networks", *IEEE Trans. on Information Theory*, vol. 46, issue 2, pp. 388-404, March 2000.

36. Hayes J., "Policy-based Authentication and Authorization: Secure Access to the Network Infrastructure," in *16th Annual Computer Security Applications Conf.*, pp. 328-333, Dec. 2000.

37. Heckman M. R. and Levitt K. N., "Applying the Composition Principle to Verify a Hierarchy of Security Servers," in *Proc. of the 31st Hawaii Intl. Conf. on System Sciences*, vol. 3, pp. 338-347, Jan. 1998.

38. Holland G. and Vaidya N., "Analysis of TCP Performance Over Mobile Ad Hoc Networks," in *Proc. of the 5th annual ACM/IEEE Intl. conf. on Mobile computing and networking*, pp. 219-230, Aug. 1999.

39. Hou T. C. and Li V., "Position Updates and Sensitivity Analysis for Routing Protocols in Multihop Mobile Packet Radio Networks," in *Proc. IEEE GLOBECOM*, pp. 243-249, 1985.

40. IEEE Standard Specifications for Public-Key Cryptography (IEEE Std 1363-2000), *http://www.ieee.org*, 2000.

41. IPSEC, internet-draft, *http://www.tml.hut.fi/Tutkimus/IPSEC/*, 2001.

42. IPv6 Organization, *http://www.ipv6.org*, 2001.

43. Jiang M., Li J., and Tay Y. C., "Cluster Based Routing Protocol (CBRP)," *IETF MANET Working Group*, internet-draft, Aug. 1999.

44. Johnson D. B. and Maltz D. A., "Dynamic Source Routing in Ad Hoc Wireless Networks, Mobile Computing," *Kluwer Academic Publishers,* pp. 153-181, 1996.

45. Jubin J. and Tornow J. D., "The DARPA Packet Radio Network Protocols," in *Proc. IEEE*, 75(1), 1987.

46. Kärpijoki V., "Security in Ad-hoc Networks," available online at *http://www.hut.fi/~vkarpijo/ netsec00/netsec00_manet_sec.html*

47. Ko Y. B. and Vaidya N. H., "Location-aided Routing (LAR) in Mobile Ad Hoc Networks," *ACM Wireless Networks*, vol. 6, issue 4, pp. 307-321, Jul. 2000.

48. Lee S. J. and Gerla M., "Split Multipath Routing with Maximally Disjoint Paths in Ad Hoc Networks," in *Proc. of IEEE Intl. Conf. on Comm.*, pp. 3201-3205, 2001.

49. Lee S. and Kim C., "Neighbor Supporting Ad Hoc Multicast Routing Protocol," in *1st Annual Workshop on Mobile and Ad Hoc Networking and Computing*, pp. 37-44, 2000.

50. Leinwand A. and Fang K., *Network Management: A Practical Perspective*, Reading, MA: Addison-Wesley, 1993.

51. Leung R., Liu J. L., Poon E., Chan A. L., and Li B. C., "MP-DSR: A Qos-aware Multi-path Dynamic Source Routing Protocol for Wireless Ad-Hoc Networks," in *Proc. of the 26th IEEE Annual Conf. on Local Computer Networks (LCN 2001)*, Tampa, Florida, pp. 132-141, Nov. 2001.

52. Li J., Blake C., Couto D. De, Lee H. I., and Morris R., "Capacity of Ad Hoc Wireless Networks", in *Proc. of the 7th annual Intl. conf. on Mobile Computing and Networking*, pp. 61-69, Rome, Italy, July, 2001.

53. Mainkar V. and Trivedi K. S., "Sufficient Conditions for the Existence of a Fixed Point in Stochastic Reward Net-based Iterative Models," *IEEE Trans. on Soft. Eng.*, vol. 22, issue 9, pp. 640-653, Sept. 1996.

54. MANET working group, [Online]. Available: *http://www.ieft.org/html.charters-/manet-charter.html.*

55. Marina M. K. and Das S. R., "On-demand Multipath Distance Vector Routing for Ad Hoc Networks," in *Proc. of the Intl. Conf. for Network Protocols (ICNP)*, Riverside, Nov. 2001.

56. Mäki S., "Security Fundamentals in Ad-hoc Networking," in *Proc. of the Helsinki University of Technology, Seminar on Internetworking - Ad Hoc Networks*, Apr. 2000.

57. McCanne S. and Floyd S., NS Network Simulator. [Online]. Available: *http://www.isi.edu/nsnam/ns/.*

58. Menezes A. J., "Elliptic Curve Public Key Cryptosystems", *Kluwer Academic Publishes*, Boston, MA, 1993.

59. Miner S. and Staddon J., "Graph-based Authentication of Digital Streams," in *Proc. of 2001 IEEE Symp. On Security and Privacy*, pp. 232-246, May 2001.

60. Murthy S. and Garcia-Luna-Aceves J. J., "An Efficient Routing Protocol for Wireless Networks," *ACM Mobile Networks and Applications Journal, Special issue on routing in mobile comm. network*, vol. 1, issue 2, pp. 183-197, 1996.

61. Netscape Inc., available online at *http://developer.netscape.com,* 2001.

62. OPNET Technologies Inc., available online at *http://www.opnet.com.*

63. Palm, Inc., available online at *http://www.palm.com*, 2001.

64. Pandiarajan V., Martin T. L. and Joiner L.L., "Recommendations on a New Cellular Encryption Standard Using Elliptic Curve Cryptography," in *IEEE Proc. on Southeast Conf.*, pp. 136-142, March 2001.

65. Papavassiliou S., Tekinay S., Malick K. and Walker K., "Performance Evaluation Framework and Quality of Service Issues for Mobile Ad Hoc Networks in the MOSAIC ATD", in *21st Century Military Comm. Conf. Proc.*, vol. 1, pp. 297-303, 2000.

66. Park V. and Corson M., "A Highly Adaptive Distributed Routing Algorithm for Mobile Wireless Networks," in *Proc. of IEEE INFOCOM '97,* vol. 3, pp. 1405-1413, April 1997.

67. Park S. K. and Miller K. W., "Random Number Generators: Good Ones are Hard to Find," *Comm. of the ACM (CACM)* 31:10, pp. 1192-1201, Oct. 1988.

68. Pearlman M., Haas Z., Sholander P., and Tabrizi S., "On the Impact of Alternate Path Routing for Load Balancing in Mobile Ad Hoc Networks," in *Proc. of the ACM MobiHoc*, pp. 3-10, 2000.

69. Pei G., Gerla M. and Chen T.W., "Fisheye State Routing: a Routing Scheme for Ad Hoc Wireless Networks", in *Proc. ICC 2000*, vol. 1, pp. 70-74, June 2000.

70. Perkins C. and Bhagwat P., "Highly Dynamic Destination-Sequenced Distance-Vector Routing (DSDV) for Mobile Computers," in *ACM SIGCOMM*, pp. 234-244, Oct. 1994.

71. Perkins C. and Royer E., "Ad Hoc On-Demand Distance Vector Routing," in *Proc. of the 2nd IEEE Workshop on Mobile Computing Systems and Applications*, pp. 90-100, Feb. 1999.

72. Perkins C., Royer E. M., Das S. R., Marina M. K., "Performance Comparison of Two On-Demand Routing Protocols for Ad Hoc Networks," *IEEE Personal Comm.,* vol. 8, issue 1, pp. 16-28, Feb. 2001.

73. Pfleeger C. P., *Security in Computing*, 2nd ed. Englewood Cliffs, NJ: Prentice-Hall, 1997.

74. PKCS–Public-Key Cryptography Standards, Available online at *http://www.rsasecurity.com /rsalabs/pkcs/pkcs-11/index.html.*

75. Raju J. and Garcia-Luna-Aceves J. J., "A New Approach to On-Demand Loop-free Multipath Routing," in *Proc. of the Intl. Conf. on Computer Comm. and Networks*, pp. 522-527, 1999.

76. Ritchey R.W. and Ammann P., "Using Model Checking to Analyze Network Vulnerabilities," in *Proc. of IEEE Symp. on Security and Privacy*, pp. 156-165, May 2000.

77. Rivest R., Shamir A. and Adleman L., "A Method for Obtaining Signatures and Public-Key Cryptosystems," *Comm. of the ACM*, vol. 21, no. 2, pp. 120-126, 1978.

78. Royer E. M. and Toh C. K., "A Review of Current Routing Protocols for Ad Hoc Mobile Wireless Networks," *IEEE Personal Comm.*, vol. 6, issue 2, pp. 46-55, 1999.

79. Ruppe R., Griswald S., Walsh P., and Martin R., "Near Term Digital Radio (NTDR) System," in *Proc. MILCOM '97*, pp. 1282--1287, Nov. 1997.

80. Sander T. and Tschudin C. F., "Protecting Mobile Agents Against Malicious Hosts," *Mobile Agents and Security*, vol. 1419 of Lecture Notes in Computer Science, pp. 44-60, 1998.

81. Sander T., Young A., and Yung M., "No-Interactive Crypto-Computing for NC," in *Proc. 40th IEEE Symp. on Foundations of Computer Science*, pp. 554-566, Oct. 1999.

82. Shamir A., "How to Share a Secret", *Comm. ACM*, vol. 22, pp. 612-613, Nov. 1979.

83. Smith B. R., Murthy S. and Garcia-Luna-Aceves J.J., "Securing Distance-Vector Routing Protocols," in *Proc. of the Symp. on Network and Distributed System Security*, pp. 85-92, 1997.

84. Stallings W., "SNMP and SNMPv2: the Infrastructure for Network Management," *IEEE Comm. Magazine,* vol. 36, issue 3, pp. 37-43, 1998

85. Sun D. and Man H., "TCP Flow-based Performance Analysis of Two On-demand Routing Protocols for Mobile Ad Hoc Networks," in *Proc. of IEEE VTC 2001,* Atlantic City, NJ, Oct. 2001.

86. Trivedi K. S., Sun H.R., Cao Y. and Ma Y., "Stochastic Petri Nets and Their Applications," in *Intl. Conf. on the Performance and QoS of Next Generation Networking (P&QNet2000)*, Nagoya, Japan, Nov. 2000.

87. Tsirigos A. and Haas Z., "Multipath Routing in the Presence of Frequent Topological Changes," *IEEE Comm. Magazine*, pp. 132–138, Nov. 2001.

88. Tuch B., "Development of WaveLAN, an ISM Band Wireless LAN," *AT&T Technical Journal*, 72(4): 27-33, 1993.

89. Varadharajan V., "A Multilevel Security Policy Model for Network," in *9th Annual Joint Conf. of the IEEE Computer and Comm. Societies,* vol. 2, pp. 710-718, 1990.

90. Venkatraman L. and Agrawal D. P., "A Novel Authentication Scheme for Ad Hoc Networks," in *Proc. of IEEE Conf. on Wireless Comm. and Networking*, vol. 3, pp. 1268-1293, Sept. 2000.

91. Williams J., "Internet/Network Security," available online at *http://netsecurity.about.com/library/weekly/aa080299.htm*

92. Xiong C., Murata T, and Tsai J., "Modeling and Simulation of Routing Protocol for Mobile Ad Hoc networks Using Colored Petri Nets," *Research and Practice in Information Technology*, vol. 12, pp. 145-153, Australian Computer Society, 2002.

93. Xu C., Gong F., Baldine I., Sargor C., Jou F., Wu S., Fu Z., and Huang H., "Celestial Security Management System," in *DARPA Information Survivability Conf. and Exposition*, vol. 1, pp. 162-172, Jan. 2000.

94. Xu S., Papavassiliou S. and Amouris K., "On the Performance of a Scalable Single-Tier Routing Protocol for Mobile Ad-hoc Wireless Networks", in *Proc. IEEE ISCC2000,* pp. 575-580, July 2000.

95. Xu S., Papavassiliou S. andAmouris K., "On the Optimal Multi-Zone Configuration for the Position-Guided Sliding-Window Routing (PSR) Protocol for Mobile Ad-hoc Networks", in *Proc. IEEE MILCOM2000,* pp. 534-538, Oct. 2000.

96. Yao A. C., "How to Generate and Exchange Secrets," in *Proc. 27th IEEE Symp. on Foundations of Computer Science*, pp. 162-167, 1986.

97. Yang J. and Papavassiliou S., "Improving Network Security Performance by Multipath Traffic Dispersion" in *Proc. IEEE MILCOM2001,* Oct. 2001.

98. Yi S., Naldurg P., and Kravets R., "Security-Aware Ad Hoc Routing for Wireless Networks," in *ACM Symposium on Mobile Ad Hoc Networking & Computing (Mobihoc 2001)*, Long Beach, CA, Oct., 2001.

99. Zhang Y. and Lee W., "Intrusion Detection in Wireless Ad Hoc Networks," in *Proc. of the 6th Annual Intl. Conf. on Mobile Computing and Networking,* pp. 275-283, Aug. 2000.

100. Zhang C. Z. and Zhou M. C., "Security Enhanced Ad Hoc On-Demand Routing Protocol", in *3rd Annual IEEE Information Assurance Workshop*, West Point, NY, June 2002.

101. Zhou L. and Haas Z., "Securing Ad Hoc Networks," *IEEE Network*, vol. 13, no. 6, Nov.-Dec., pp. 24-30, 1999.

102. Zhou M. C. and Venkatech K., *Modeling, Simulation and Control of Flexible Manufacturing Systems - A Petri Net Approach*, World Scientific Publisher. 1999.