Science Arena Publications
Specialty journal of Engineering and Applied Science
Available online at www.sciarena.com
2016, Vol, 1 (2): 56-63

Design of Optimal a Wireless Mesh Network

Mahdi Mozaffari Legha[1], Mohammad Eftekharikenzerki[2], Marjan Karimi[3]
[1]*Department of Power Engineering, Institute of Higher Education Javid, Jiroft, Iran.*
Email: Mozaffarilegha.m@gmail.com
[2]*Department of Power Engineering, Institute of Higher Education Javid, Jiroft, Iran.*
Email: Mohammadeftekharikenzerki@gmail.com
[3]*Department of Power Engineering, Institute of Higher Education Javid, Jiroft, Iran.*
Email: Atefe.karimi405@gmail.com

Abstract: *In these days, IEEE802.16 protocol is being developed and in the next years, 802.16 modems will be used in every house to connect to the Internet through base-stations in both urban and rural areas. Since user bandwidth for connection to the Internet is limited, a large fraction of each modem's bandwidth is not used. PPMN is a public peer-to-peer mesh network using the non-utilized bandwidth of independent 802.16 users. In this paper, we improve security and resiliency in the design of PPMN. Design performance is evaluated using simulation.*
Key words: *Peer-to-Peer - Wireless Mesh Network – Resiliency - Security*

INTRODUCTION

Every city will soon be full of IEEE802.16 users covered by BSs (Base-Stations) belonging to different service providers. Each BS covers a limited area and gives service to a number of users. Users are mostly fixed and do not encounter energy limitations. Bandwidth to BS is a critical resource for them, but they have a significant amount of bandwidth to their neighbors in a way that all 802.16 users in a city can sustain a huge volume of intra-city data traffic. To utilize this bandwidth, users under the same authority can compose a wireless mesh but such a network is not scalable.

In this paper, we improve security and resiliency in the design of PPMN [1] which is the Public Peer-to-peer Mesh Network of all users in a city. Users can be wired or wireless under any authority. They have to forward intra-city traffic themselves. As a result, it is not required to transfer intra-city traffic through ISPs except for critical information.

Since PPMN consists of nodes under different authorities, it needs a registration scheme to keep track of nodes. Then, security issues can be handled. [2] Categorizes security solutions in peer-to-peer networks into three categories: identity, trust & reputation and incentives. The research that is done in the identity category focuses on researching solutions that achieve anonymity and access control. In the trust & reputation category, the research focuses on trying to achieve availability and authenticity using trust systems. The research that is categorized in the incentives category deals with trying to achieve fair trading and availability of peers by researching various ways to incite peers to contribute to the system.

Since nodes are not trusted, there must be a resiliency mechanism to guarantee a level of reliability for a data transfer session. There is extensive research aiming at finding two link-disjoint paths for data transfer, one as the active path and the other as the backup path which is used when the active path fails [3]. The authors in [4] argue that using disjoint paths limits route reliability in mobile ad hoc networks compared to using multiple loop-free paths that need not be disjoint. [5] proposes a generic resilient multi-path routing scheme for mobile ad hoc networks with the aim of ensuring network throughput in both adversarial and node failure scenarios. [6] investigates resilience to security attacks in ad hoc networks. SEMAP [7] is a secure enhancement mechanism based on Attacking Point which converts the possibility of security threat to a concrete metric. It can exclude the nodes that will be the objects of adversaries from the network before actual routing process.

The rest of this paper is organized as follows. Section II presents the network model of PPMN. Section III discusses security issues in PPMN. Section IV proposes a resiliency scheme. Section V contains our simulation results. Section VI finally concludes the paper.

NETWORK MODEL

Each ISP contains an LRC (Local Routing Controller) and there is one GRC (Global Routing Controller) in the city (Fig. 1). There is a hierarchy. An LRC composes a domain and controls and performs local routing operations in its domain whereas the GRC performs inter-domain routings. An LRC knows the topology in its domain whereas the GRC only knows the top-level interconnections between the domains. A directed link from node *a* to node *b* in a topology graph means that node *b* is willing to forward node *a*'s packets. LRCs and

the GRC are connected to the city backbone and only update topology changes. The routing scheme is described in [1].

A user needs the following items to be able to use PPMN.

1. An IEEE802.16 wireless modem.
2. A connection to the city backbone. This connection can be wired or wireless provided by an ISP or another PPMN node.

Each LRC is responsible for routing in the domain containing the following nodes.

1. Subscriber nodes of the ISP.
2. Nodes which are not subscribed to any ISPs and send *RouteRequest* to a subscriber node of the ISP.

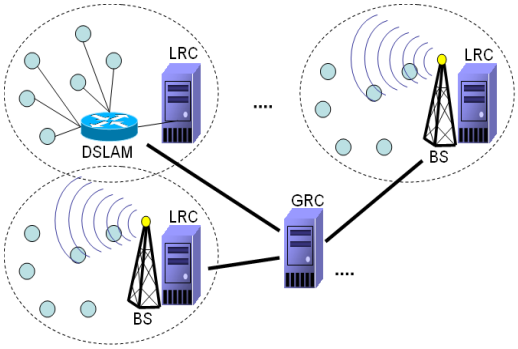

Fig. 1: An illustration of routing domains in PPMN; two domains are logically distinct but may have physical overlap;

Increasing number of domains generates a bigger top-level topology graph and thus increases the load on the GRC. Therefore, we should not assign a separate domain to small ISPs in a large city. Instead, we aggregate small neighbor ISPs in a single domain with one LRC. Furthermore, we assume physical overlap of domains are not too much in a manner that number of intra-domain links is at least 10 times more than number of inter-domain links in the PPMN.

A session is defined as the duration of data transfer between a source node and a destination node in PPMN. A full-duplex route is established for a session such that both the source and the destination can send packets to each other. The source embeds the route into packets such that intermediate nodes are able to forward the packet without doing routing.

Bandwidth is reserved along the route. The amounts of bandwidth reserved along the forward path and backward path depends on traffic symmetry of the application [8]. For example, IP-Telephony is symmetric but FTP is asymmetric. Even if the application is highly asymmetric, ACK packets (Section 5.3) are at least transferred along the non-loaded path.

Security Scheme

There must be a scheme in PPMN to ensure security and reliability of established routes and prevent/resist misbehaviors/attacks. Because of the virtual banking, every node must register with the GRC using its MAC

address as a unique ID along with the name and the address of the owner just like telephone registration. In other words, each node owner has to go to the GRC office to register its MAC address. Then that node can join the PPMN.

Key Management

A key management scheme [9] suitable for non-trusted environments should be used to enable cryptography in PPMN. The scheme must guarantee that keys are confidentially established and updated. The following kinds of keys are essential in PPMN.

- *Session Key*. A new key should be established between the source and the destination for each data transfer session. Every packet generated between the source and the destination of session is encrypted using session key.
- *Initial Session Key*. (Described below)
- *Pairwise Key*. A node has to separately establish a pairwise key with each neighbor so that routing information exchanged between two neighbors is not exposed to other neighbors.

- *Routing Key*. Each node has to establish a routing key with the corresponding LRC to ensure secure routing operations. This key has to be updated periodically.
- *Bank Key*. Each node has to establish a bank key with the bank server to ensure secure money exchange. This key has to be updated periodically.

There are a number of key-exchange mechanisms (for example, Diffie-Hellman[10] and SPEKE[11,12]) that provide key establishment between two nodes without transferring the real key across the network. Such a mechanism transfers a number of messages between the two nodes to establish a common key. To protect the key-establishment messages from malicious intermediate nodes, these messages are encrypted. The two nodes initially have no common key. They request a trusted key server for an initial key to encrypt key-establishment messages. Once the session key is established, they release the initial key and use the session key afterwards. Although extracting the session key from key-establishment messages is too hard, it is preferred that no node should be able to have both encrypted key-establishment messages and the initial key such that it extracts plain key-establishment messages. Since key-establishment messages are transferred along a PPMN path, the key server which is the only node who knows the initial key is not able to receive them. This mechanism ensures that the session key is not exposed to anyone other than the source and the destination.

Misbehaviors and Attacks

A single node or a group of cooperative nodes can harm the system in one of the following ways.

- *Sybil Attack*. This is an easy attack in most peer-to-peer systems in which a node joins the system with different identities [9,13]. Since nodes are identified by their MAC addresses, this is not possible in PPMN.
- *Frequent Departures*. This action imposes heavy load on the system since it requires frequent topology updates and path recoveries. To avoid this
- misbehavior, there should be a minimum time interval in which a node can join PPMN. Neighbors do not send *TopologyReport* messages until a node reach a stable state and an LRC does not accept frequent *TopologyReport* or *Hello* messages from a single node.
- *False Misbehavior Report*. A report to a controller, that indicates a node is misbehaving, lowers the reputation of that node. To prevent false reports, the system should only accept reports that contain evidence.
- *False Money-Exchange Report*. A number of mechanisms [14,15] have been proposed to guarantee proper money exchange after a data transmission session.
- *False Failure Report*. We discuss how to reject this kind of false reports in the next subsection.
- *Misforwarding*. A malicious node can do selective forwarding [9], disordering packets, or degrading QoS of a flow. Detecting such kind of attacks is so hard since congestion can cause the same consequences as well and misbehavior reports are mostly not trusted.
- *False Routing*. This happens when a node injects false routing information into the system. If a node lies about a neighbor to an LRC, then that neighbor may tell the truth and then the false information will be rejected. When both the neighbor nodes lie and establish a fake link, they are not able to make extra payoff and the fake link will be detected since it does not satisfy the QoS requirements of passing flows. This kind of attacks is categorized in [9] and is prevented in our PPMN design since routing messages are fully controlled.

Resiliency Scheme

Since not all kinds of failure/misbehavior can be prevented, an established session should be resilient to them. In this section, we propose a multipath session resiliency scheme in PPMN based on [5] to cope with 1) Node Departure/Failure, 2) Misforwarding, 3) Misreporting, and 3) Congestion (QoS Degradation).

Two paths of similar length and price between a given source node and a destination node are discovered on-demand (Fig. 2). Then the source and the destination initially transmit data along these paths in a round-robin manner. Then they adapt data transmission rate along the two paths according to path price and bandwidth. When the source discovers a path performs poorly in comparison with the other one, it releases the path and requests the LRC for another path. This mechanism hinges on the fact that misforwarding and congestion in a network have the common symptoms of disruption of data communication which manifests itself in the form of reduced throughput. Both the source and the destination continuously monitor the throughput of each path. The destination reports the throughput to the source along the other path. In this way, no intermediate node is able to alter the reports or to generate fake reports.

To acknowledge arrived packets, the receiver (the source or the destination) sends ACK packets (as specified in TCP) along the two paths in a round-robin manner to the sender. We assume that packet content is encrypted so

that intermediate nodes are not able to understand the content or determine which packets are being acknowledged.

If *packet(i)* is lost, then *packet(j)* where $j>i$ which is received successfully will become useless. This is the base of the selective forwarding attack in which a malicious intermediate node drops a few number of selected packets in a session to effectively disturb the whole session. In this case, the throughputs of both the paths are not far apart and the losses affect data transmission along both the paths. A solution to this is that the receiver gains knowledge about the sequence of packets that arrive from each one of the paths and immediately requests the sender for retransmission if an out of order packet arrives since it

indicates that a packet is lost. Once a path is established, the source and the destination secretly agree on a series of sequence numbers with which packets are sent on that path. This series is updated when the sender decides to

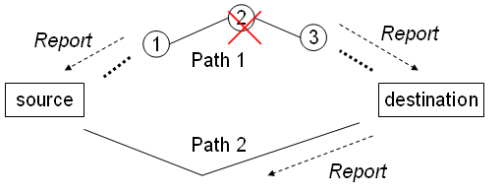

Fig. 2: Reports when node(2) fails or departs

change data transmission rate on that path. Another solution is to create content redundancy between the two paths. If a packet is lost along a path, it can be recovered from the contents received along the other path.

This mechanism can be incorporated in our routing algorithm. The two paths should be maximally disjoint. There should be a time interval in which no more than one route request is accepted from one node.

This level of reliability works perfect for applications such as web, file transfer, and telnet. However, there are real time applications such as telephony that require not much bandwidth but more reliability. This can be achieved using a low bandwidth reliable path through ISPs and one high bandwidth path inside PPMN. This strategy is also useful when at most one path is found in PPMN between the source and the destination. To increase reliability and decrease cost, control messages such as acknowledgements and reports are transferred along the low speed ISP path and data messages are transferred along the high speed PPMN path. In general, the following paths can be selected for a session.

- Two paths in PPMN.
- A low-speed path through ISPs and a high-speed path through PPMN.
- One path through ISPs.

Reporting Mechanism

Reporting to a controller is required in one of the following cases. A report of any kind may be falsely generated.

1. *Failure*.
2. *Misbehavior and Reputation*.
3. *Money Exchange*. A number of mechanisms are proposed in the literature [14,15] to guarantee proper money exchange and avoid fraud.

When *node*(*i*) along one of the established paths between a source and a destination fails or departs, three reports are generated (Fig. 2): *node*(*i*-1) sends a failure report to the source along the same path and *node*(*i*+1) sends a failure report to the destination along the same path. Then the destination sends a failure report to the source along the other path. This mechanism ensures the source that both the reports which it receives are not fakes. Then the source informs the LRC of this node failure and asks for another path. The LRC directly checks *node*(*i*)'s status and discover if the source, the destination, or an intermediate node is telling the truth. This mechanism provides resiliency to failure/departure.

Since misbehaviors can not be proved, this kind of reports is not trusted. Reputation systems solve this problem by using a voting mechanism in which a number of nodes are asked about a particular node. In PPMN, only two neighbors along a path are able to monitor an intermediate node and then voting can not guarantee that a node is misbehaving to get a bad reputation. Therefore, a bad/negative reputation does not work. For this reason, most reputation systems only consider positive reputation where the reputation of a node will never become less than zero [13]. Thus, we suggest the reputation system in PPMN in a way that an LRC sets the reputation of a node according to the way that the node handles the sessions to which it is assigned. In response, the more reputation a node has, the cheaper and the shorter paths it gets for data transfer.

OHOF

Opportunistic routing (OR) and network coding (INC) are two state-of-the-art techniques to improve the performance of wireless mesh networks. In contrast to traditional routing which forwards packets along a fixed path from a source to a destination, OR opportunistically exploits multiple paths between the source and the destination. OR broadcasts the packet first and then decides the next hop among all neighbors that hear the packet successfully, thus providing more chances for a packet to make some progress towards the destination. In INC, a forwarding node encodes a number of packets belonging to different flows into a single packet and then broadcasts the packet to all its neighbors. This reduces the number of transmissions needed for forwarding packets belonging to different flows, and hence increases the effective capacity of the network.

A malicious node can easily make encoded packets non-decodable to receivers [16]. To prevent this, there must be a mechanism that may increase overhead. Thus, this is an open problem whether INC is useful in a non-cooperative network. In contrast, we use the idea of OR and propose a mechanism called One-Hop Opportunistic Forwarding (OHOF) that does not contain routing. It can be incorporated in a unicast routing algorithm to improve resiliency and load balance.

We assume that the session path is embedded in packets. When a node is going to forward a packet to the next hop, it broadcasts the packet (Fig. 3). In addition to the next hop, every node that is neighbor of all the three nodes (Forwarding node, Next-Hop, and Second Next-Hop) and has free bandwidth receives the packet. Each neighbor (whether it is the next hop or not) waits a random time interval until it hears another node sending the packet to the second next hop. Otherwise, it takes the -

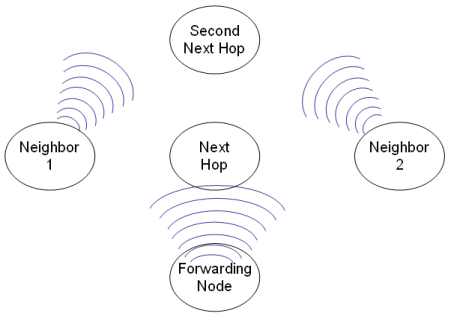

Fig. 3: OHOF mechanism with one next hop and two neighbors

wireless channel and forwards the packet to the second next hop. There are a number of channel-scheduling schemes [17] to avoid collision in taking the channel even in non-cooperative environments [18]. When a neighbor receives a packet that is not a neighbor of the packet's second next hop, it simply drops the packet. This avoids packets from getting away from their default path and prevents redundant transmissions.

If two neighbors are too far or if directional antennas are used, then they are not able to monitor each other and they may independently forward the packet. To cope with this problem, each neighbor has to check with the second next hop before forwarding if it is receiving the packet now or has received the packet yet. When the second next hop successfully receives the packet, it broadcasts a *Confirmation* message to all its neighbors to stop trying to forward the packet.

When an intermediate node is congested, it makes delay in forwarding packets and thus its non-congested neighbors do the forwarding. Since 802.16 modems have high transmission range, it is expected that a single node has several neighbors and then OHOF will significantly improve tolerance to failure and congestion. Since a path is not so thin, the two disjoint paths are preferred to be zone-disjoint to being link-disjoint.

Since packets may be forwarded by different nodes during a session, OHOF requires a per-packet money exchange scheme to ensure intermediate nodes receive appropriate payoffs. A neighbor node that has forwarded a single packet and is not in the path should be able to receive its payoff. When a packet is forwarded at a hop in the path, a number of nodes receive it. The scheme must guarantee that the node who delivers the packet to the next hop earns the money. Among the existing money exchange schemes, Sprite [14] and ASR [15] suit to PPMN that requires a trusted virtual bank.

Simulation Results

SWANS [19] is a Scalable Wireless Ad hoc Network Simulator built atop the JiST platform, a general-purpose discrete event simulation engine. Simulation environment is prepared by developing a new MAC layer called MacPPMN to SWANS. This layer acts as both the MAC sub-layer and the PPMN sub-layer. We developed the PPMN routing functionality in a file called RoutePPMN.java. Other layers in SWANS are not

changed and we used them in file "sim.java". The following packet types are added to SWANS.

- RouteRequestMessage
- RouteReplyMessage
- TopologyReportMessage
- HelloMessage

Figure 4 illustrates number of route requests the GRC has to reply versus number of nodes. We have designed our routing algorithm in a hierarchical way that the GRC only maintain a top-level topology and does not need to be aware of internal topology of domains. As the figure shows, doubling number of nodes does not double load on the GRC. When number of nodes increases, we have more traffic flows and more departures/joins. These two elements require more route requests to establish new path.

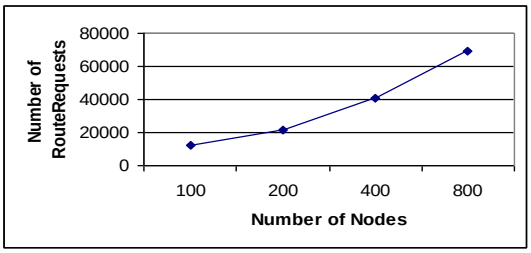

Fig. 4: Load on the GRC vs. number of nodes

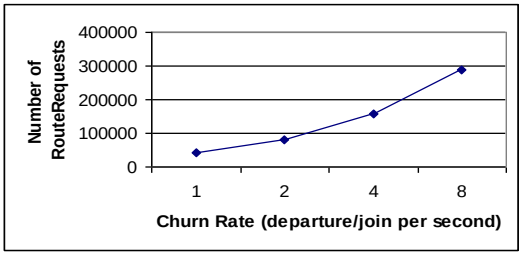

Fig. 5: Load on the GRC vs. churn rate

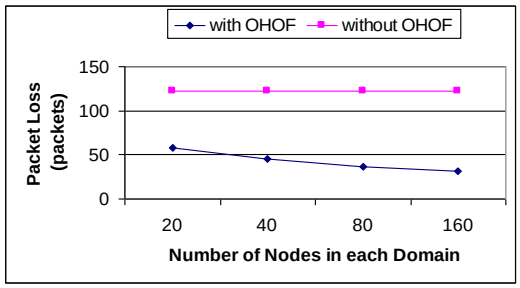

Fig. 6: Packet loss vs. node density; nodes and traffic do not change; multipath routing is disabled

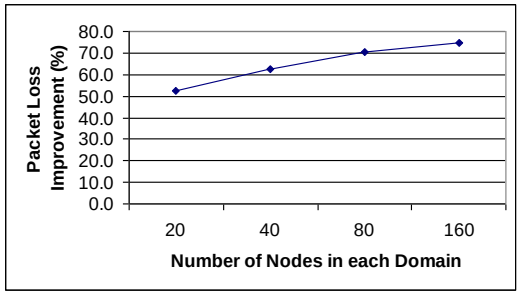

Fig. 7: Packet loss improvement when using OHOF in Figure 6

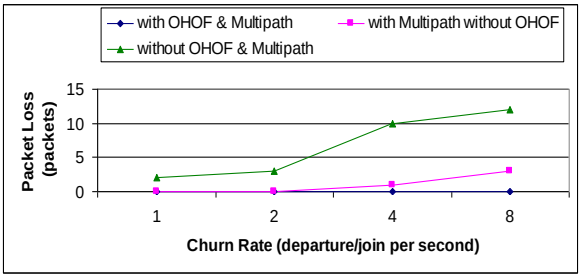

Fig. 8: Packet loss vs. churn rate; nodes and traffic do not change; each flow uses two paths when Multipath is enabled

Figure 5 illustrates number of route requests the GRC has to reply versus churn rate. A churn happens when a node joins or departs the network or even when it fails. When this happens, a traffic flow may lose a path and requires sending request to the GRC to discover another path. When churn rate increases, we have more departures/joins. This leads to more route requests to establish new paths. Doubling churn rate does not double load on the GRC.

Figure 6 illustrates number of lost packets in the network versus node density, while the number of nodes and the traffic do not change and there is a uniform loss probability of 0.1 in the network. Without using the opportunistic forwarding (OHOF) technique, we expect that 10 percent of packets are lost which equals 122 packets. With OHOF enabled, neighbors of a forwarding node help if a packet is lost. The more neighbors a node have, the more effective OHOF is in avoiding end-to-end packet loss. These facts are satisfied in the figure. The results show that doubling node density decreases packet loss by averagely 6 percent (Fig. 7).

Figure 8 illustrates number of lost packets in the network versus churn rate, while the number of nodes and the traffic do not change and there is a loss probability of 0 in the network. Churn may break an established route. Without using OHOF and Multipath techniques, packets are lost until a new path is found. Without using OHOF and with Multipath technique, packets are sent on one path when one of the two paths fails. If we use both the techniques, when a node departs, the path does not fail and packets are forwarded through it since neighbors of the departed node forward the incoming packets. This has led to zero packet loss in this simulation experiment.

References

A. Kavianfar, and J. Habibi, “PPMN: A Resilient Public Peer-to-Peer Mesh Network of IEEE802.16 Users”, in proc. of International Conference on Computer Design and Applications (ICCDA’09), Singapore, 2009.

J. Risson, and T. Moors, “Survey of Research towards Robust Peer-to-Peer Networks: Search Methods”, Techincal Report UNSW-EE-P2P-1-1, University of New South Wales, Sydney, 2004.

R. Bhandari, “Survivable networks: algorithms for diverse routing”, in Kluwer, 1999.

M. Mosko and J. Garcia-Luna-Aceves, "Multipath routing in wireless mesh networks", in Proc. of the First IEEE Workshop on Wireless Mesh Networks (WiMesh), September 2005.

M. Kesavan, D. Swarup, K. Andiappan, “A Generic Resilient Multipath Routing Mechanism for Failure Prone Ad Hoc Networks”, in the poster session at The International Conference on High Performance Computing (HiPC) 2005, Goa India.

I. Aad, J.P. Hubaux, and E. Knightly, “Denial of Service Resilience in Ad Hoc Networks”, in Proceedings of ACM MobiCom 2004, Philadelphia, PA, September 2004.

Z. Lu, C. Huang, F. Wang, and C. Rong, “SEMAP: Improving Multipath Security Based on Attacking Point in Ad Hoc Networks”, in proc. of ATC08, 2008.

Y. Chen, T. Farley, and N. Ye, “QoS requirements of network applications on the Internet”, Information, Knowledge, Systems Management, Vol. 4, No.1, pp. 55-76, 2004.

C. Karlof, D. Wagner, "Secure Routing in Wireless Sensor Networks: Attacks and Countermeasures", in IEEE SPNA, 2002.

W. Diffie, P. C. Oorschot, and M. J. Wiener, "Authentication and Authenticated Key Exchanges", Designs, Codes and Cryptography, vol.2 n.2, pp. 107-125, June 1992.

P. MacKenzie, "On the Security of the SPEKE Password-Authenticated Key Exchange Protocol", Technical. Report 2001/057, Lucent Technologies, 2001.

D. K. Nilsson, "Bluetooth Pairing and Authentication Vulnerabilities, Protocol Design, Implementation", Edited by Erland Jonsson. Goteborg, Sweden, Department of Computer Engineering Chalmers University of Technology, 2006.

J. Vroonhoven, "Peer to Peer Security", 4th Twente Student Conference on IT, Enschede 30 January, 2006.

S. Zhong, J. Chen, and Y. R. Yang, "Sprite: a simple, cheatproof, credit-based system for mobile ad-hoc networks", in proc. of the 22nd Annual Joint Conference on the IEEE Computer and Communications Societies (INFOCOM '03), vol. 3, pp. 1987-1997, San Francisco, Calif, USA, March-April 2003.

H. Choi, W. Enck, J. Shin, P. McDaniel, T. Porta, "ASR: Anonymous and Secure Reporting of Traffic Forwarding Activity in Mobile Ad Hoc Networks", Technical Report NAS-TR-0034-2006, Dept. Computer Science and Eng., Pennsylvania State Univ., 2006.

S. Jaggi, M. Langberg, S. Katti, T. Ho, D. Katabi, M. Medard, and M. Effros, “Resilient network coding in the presence of byzantine adversaries”, IEEE Transactions on Information Theory, 2006.

L. Bui, A. Eryılmaz, R. Srikant, X. Wu, "Joint Asynchronous Congestion Control and Distributed Scheduling for Multi-Hop Wireless Networks", in proc. of IEEE INFOCOM'06, 2006.

M. Felegyhazi, M. Cagaljy, S. Saeedi, and J. Hubaux, "Non-cooperative Multi-radio Channel Allocation in Wireless Networks", in Proceedings of Infocom'07, Anchorage, USA, May 6-12, 2007.

R. Barr, "SWANS– Scalable Wireless Ad hoc Network Simulator - User Guide", http://jist.ece.cornell.edu/swans-user/index.html,March 2004.