Transactions Letters

Improving Security of Real-Time Wireless Networks Through Packet Scheduling

Xiao Qin, Mohamed Alghamdi, Mais Nijim, Ziliang Zong, Kiranmai Bellam, Xiaojun Ruan, and Adam Manzanares

***Abstract*—Modern real-time wireless networks require high security level to assure confidentiality of information stored in packages delivered through wireless links. However, most existing algorithms for scheduling independent packets in real-time wireless networks ignore various security requirements of the packets. Therefore, in this paper we remedy this problem by proposing a novel dynamic security-aware packet-scheduling algorithm, which is capable of achieving high quality of security for real-time packets while making the best effort to guarantee real-time requirements (e.g., deadlines) of those packets. We conduct extensive simulation experiments to evaluate the performance of our algorithm. Experimental results show that compared with two baseline algorithms, the proposed algorithm can substantially improve both quality of security and real-time packet guarantee ratio under a wide range of workload characteristics.**

***Index Terms*—Real-time packets, wireless networks, packet scheduling, deadlines, quality of security.**

I. Introduction

IN the recent years, wireless technology has become one of the hottest buzzwords of the IT industry and academe due to the rapid growth of applications using wireless networks. Nowadays, people can access to the Internet at homes, hotels, airports, and even cars to conduct business, transfer money, play games and the like.

Nobody will doubt the fact that wireless networks bring incredible productivity and new efficiencies to our work. Based on the 2005 survey report of National Telecommunications Cooperative Association (NTCA), 62% of survey respondents are providing wireless services to their customers and 56% offer real time services like mobile voice [1]. The worldwide market value of wireless applications is $433 billion in 2003 and projected to grow to almost $6708 billion by 2008 [2]. Even more excitingly, with the development of wireless technology, an increasing number of innovative applications like GPS, portable printing, signature capture are being used or will be used to improve our lives. As users become increasingly mobile and business applications become more interactive, traditional wireless communication technology is unable to satisfy real-time transmission requirements in mobile electronic commerce applications. To overcome this problem, real-time wireless communication techniques allowing users to collect and transmit data in a timely manner attracts many scholars and researchers.

It should be noted that supporting efficient and reliable data transmission, especially real time data transmission, over wireless networks is extremely difficult and challenging because wireless networks must be facing more complicated environments compared with conventional wired networks. For instance, wireless networks could be disturbed by radio wave and thunderstorms or blocked by physical objects like mountains or skyscrapers. Even worse, high mobility coupled with a variety of explosively increased users makes existing security policies in wireless networks inefficient or even useless, meaning that wireless networks can be easily attacked by computer viruses, worms, spy wares, and similar threats. These security threats cause downtime or continual patching in wireless networks and thus lead to severe disruption in wireless commercial business. Therefore, boosting security of wireless networks has become one of the most important issues in the arena of wireless communications.

An array of security policies such as authentication and confidentiality strategies and 802.11 wireless communication protocol based security schemas have been proposed and applied in real-time wireless networks. However, most of them only consider security issues in a static mode, in which security levels are all configured when wireless network systems are built. In some real-time systems like stock quote updating and trading system [3], users may need flexible quality of security measured as security levels. For example, the data of currently stock price may have higher security level than the data of ten years before. Thus designing flexible security mechanisms for real time applications transmitting packets through wireless networks is highly desirable. In this paper, we present a novel Security-Aware Packet Scheduling Strategy or SPSS for real-time wireless links. The SPSS algorithm aims to dynamically determine security levels of packets according to applications'

Manuscript received February 28, 2007; revised September 4, 2007; accepted October 6, 2007. The associate editor coordinating the review of this letter and approving it for publication was D. Wu.

X. Qin, Z. Zong, K. Bellam, A. Manzanares, and X.-J. Ruan are with the Department of Computer Science and Software Engineering, Auburn University, Auburn, AL 36849 USA (e-mail: xqin@auburn.edu).

M. Nijim is with the Department of Computer Science at the School of Computing, Hattiesburg, MS 39406 USA.

M. Alghamdi is with the Department of Computer Science, New Mexico Tech, Socorro, NM 87801 USA.

Digital Object Identifier 10.1109/TWC.2008.070244.

1536-1276/08$25.00 © 2008 IEEE

security requirements while guaranteeing deadlines for packet. Apart from achieving high quality of security, SPSS described in Section 3.3 can significantly improve guarantee ratio, which is a fraction of total transmitted packets that are found to be delivered before their deadlines (see Section 4.2).

The major contributions of this work include: (1) an analysis of security and real-time requirements for wireless network; (2) a packet model to specify both timeliness and security requirements; (3) a novel security-aware packet scheduling; (4) a new performance metric integrating both security and real-time performance; and (5) a simulator where the SPFF algorithm is implemented and evaluated.

The remainder of this paper is organized as follows. Section 2 discusses previous works in the area of ensuring security for wireless networks. Section 3 describes architecture and the system model. In section 4, we present the performance evaluation of our algorithm. Finally, we will conclude the paper and future work in Section 5.

II. Related Work

Since security concern plays a vital role in the design and development of wireless mobile commercial applications, international wireless organizations, wireless equipment providers and academic researchers made extreme efforts in maximizing the features of existing security mechanisms and finding innovative security policies of wireless networks. IEEE improved the security character of 802.11 by designing 802.1X and 802.11i for WLAN [4]. 802.1X, a port-level access control protocol, provides a security framework for IEEE networks, including Ethernet and wireless networks. The 802.11i standard, also still in draft, was created for wireless-specific security functions that operate with IEEE 802.1X. Cisco provides the solutions for wireless applications by using strong encryption technology and providing unified WLAN [5]. Papers addressing the security problems also provide valuable solutions for wireless business applications [12] [13] [14] [15]. However, most of the efforts were made at the levels of protocols or systems; most existing approaches were focused on non-real time wireless applications.

Packet scheduling plays an important role in achieving high performance in real-time wireless networks. A real time scheduler needs to guarantee both security and real-time constraints of packets even in the presence of hardware and software faults [17] [18]. Real time scheduling algorithms can be classified into static [19] [20] and dynamic [21] [22] strategies.

Our research provides an effective dynamic security mechanism at the packet level for real time wireless links. There are some existing packet-scheduling algorithms designed to improve the performance of wireless networks. Chang and Yu presented packet scheduling algorithms to guarantee the quality of service in wireless applications of ATM and video traffic [6] [16].The fairness issue in packet scheduling has also been addressed in many research papers [7] [8] [9] [10] [11]. To the best of our knowledge, less attention has been paid towards the security issue in the context of packet scheduling for wireless networks. In this paper, we will integrate the proposed packet-scheduling algorithm with dynamic security adjustment strategy. In doing so, we build a new secure packet scheduling scheme for real-time wireless networks.

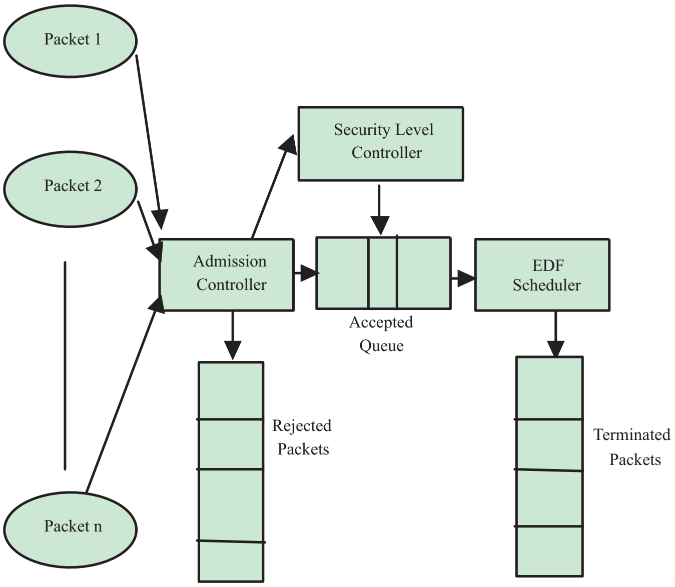

Fig. 1. The architecture of the networked system.

III. Security-Aware Packet Scheduling

A. The system model and assumptions

In this study, we model a wireless channel as an NN switch. Although each wireless node may have a single transmitter and a single receiver, it is common that the transmitter and receiver are combined in a transceiver. As such, a node can not transmit and receive packages simultaneously. In our switch model, there exists a packet scheduler matching transmitters to corresponding receivers. The detailed information regarding the switch model can be found in [23]. In addition to the switch, other three key components in the system include a Security Level Controller (SLC), an Admission Controller (AC), and an EDF (Earliest Deadline First) scheduler as depicted in Fig. 1. This architecture is designed for a link between two nodes in a wireless network. All packets are submitted independently to the wireless link with arrival rates abided by Poisson distribution. The function of the Admission Controller is to determine whether incoming packets can be accepted or not. The Security Level Controller aims at increasing security levels of real-time packets residing in the Accepted queue that can be finished before their deadlines. The EDF scheduler makes use of the Earliest Deadline First policy to schedule admitted packets in which security levels are maximized by the Security Level Controller.

B. The packet model

Our packet model assumes that all packets have soft deadline and each packet is independent of one another. We also assume that packets' arrival times follow the classical Poisson distribution. Packet Pi is represented as a tuple (AT_i, PT_i, SL_i, D_i), where AT_i and PT_i denote the arrival time and the processing time of packet *i*. SL_i and D_i represent the security level and soft deadline of packet *i*. Besides, without loss of generality we assume that each packet is assigned a quality of security measured as a security level SL_i that in the range [1, 2, ..., 10], where 1 and 10 are the lowest and highest levels of security. For example, if packet i has a value of 1 as a security level, this means that the packet has the lowest security level. Although wireless network devices are unable to determine security levels, packets' security levels can be straightforwardly derived from the security requirements of applications.

To calculate the security overhead without loss of generality, we make use of formula (1) to model the security overhead envisioned as the extra processing time experienced by packet i.

$$SO_i = ET_i * (SL_i/R) \qquad (1)$$

where SO_i is the security overhead of packet i, SL_i is the security level provided to packet i, ET_i is the transmission time of the packet. And R is set to 10. Thus, the total processing time WL_i of packet i can be expressed as:

$$WL_i = ET_i + SO_i = ET_i * (1 + SL_i/R) \qquad (2)$$

C. The SPSS Algorithm

The main goal of this study is to maximize the overall system performance, which reflects the guarantee ratio and security level. To achieve this goal, we designed the SPSS scheduling algorithm with security awareness. SPSS aims to maintain high guarantee ratios while maximizing the security levels. We can accomplish high performance and high security level by applying the Security Level Controller to our SPSS algorithm.

Fig. 2 below outlines the flow chart of the security-aware packet-scheduling algorithm (SPSS) for wireless links. The SPSS algorithm strives to maximize the security level of a packet residing in the accepted queue while making the best effort to guarantee its deadline. If the deadline of the packet can be met, the packet will be admitted in the accepted queue. Otherwise, the packet will be dropped and placed in the rejected queue . The following constraint shows whether the packet is equipped to meet its deadline.

$CT_i - ST_i <= d_i$ where ST_i is the start time of transmission of the ith packet, CT_i is the completion time of the transmission, and d_i is the packet's deadline.

The packets stored in the accepted queue are scheduled depending on their specified deadlines, meaning that the packets with earlier deadlines will be processed first. The SPSS algorithm initializes the security levels of all packets to the minimum levels. Then, SPSS gradually enhances the security level of each packet P_i under the condition that (1) the current packet P_i can be transmitted before its deadline; and (2) the deadlines of the packets being processed later than P_i also can be guaranteed. The above criterion is important and reasonable because if a packet is admitted to the real-time wireless link, then the packet's timing constraint has to be guaranteed. In other words, the SPSS algorithm ensures that an admitted packet is not adversely affected by subsequently admitted packets.

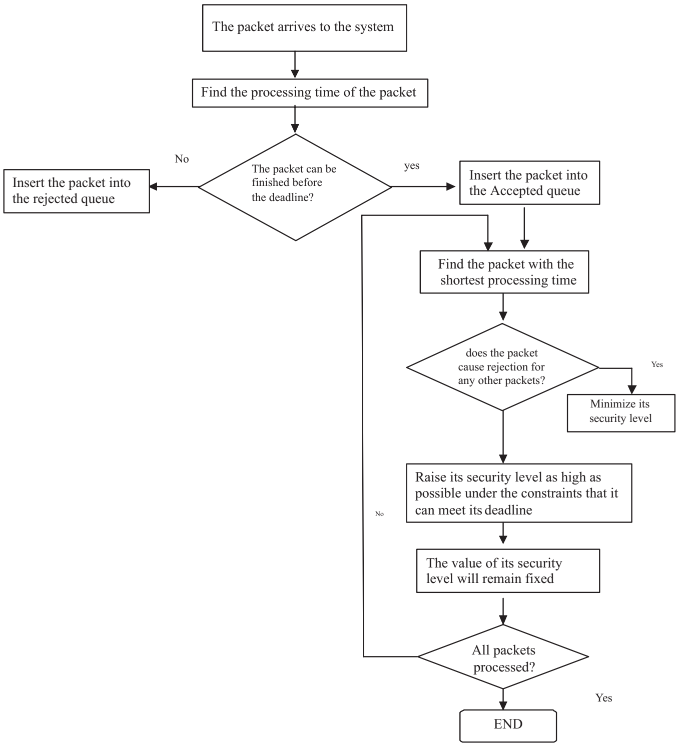
The packet arrives to the system
Find the processing time of the packet
No
The packet can be finished before the deadline?
yes
Insert the packet into the rejected queue
Insert the packet into the Accepted queue
Find the packet with the shortest processing time
does the packet cause rejection for any other packets?
Yes
Minimize its security level
No
Raise its security level as high as possible under the constraints that it can meet its deadline
The value of its security level will remain fixed
All packets processed?
Yes
END

Fig. 2. The SPSS Algorithm.

The following steps delineate the procedure of the SPSS scheduling. Step 1: initialize the scheduler; the security values of incoming packets; and the number of rejected packets is set to zero. Wait for any incoming packets. Step 2: if a packet i arrives and it is the only packet available, process the packet immediately using its highest security level. The starting time *(ST_i)* and the completion time *(CT_i)* of the packet are calculated. Step 3: All the packets arriving in the scheduler during the time period *[ST_i, CT_i]* are temporarily stored into a waiting queue in the non-decreasing order of their deadlines. The starting time of the next packet $ST_i + 1$ is set to CT_i. Step 4: the admission controller is responsible for deciding whether a packet in the waiting queue can be accepted by considering the deadline of this packet. If the packet's deadline and security requirement can both be guaranteed, the packet will be forwarded into the accepted queue (step 3 and step 5). Otherwise, being put into the rejected queue will drop the packet; the number of rejected packets is increased by one. Step 5: the security level controller raises the security levels of all the packets residing in the accepted queue as high as possible. The enhancements of the security levels for real-time packets residing in the accepted queue are subject to the following two constraints: (1) Increasing of an accepted packet's security level should still guarantee the deadline of the packet. (2) The increase of security levels must not lead to any rejection of currently accepted packet. Step 6: At this point, the security level SL_{i+1} of the next starting packet is maximized. The packet's completion time CT_{i+1} is calculated. Steps 3-6 are repeatedly executed until all the arriving packets are processed in one run.

IV. Simulation Results

A. Baseline algorithms

Now we briefly outline the ideas of the following two baseline algorithms, which are used to compare with our proposed packet-scheduling algorithm. *MIN*: The Admission Controller intentionally selects the lowest security level of each coming packet. Therefore, the guarantee ratio is improved at the cost of reducing overall security value of the system. *MAX*: The Admission Controller chooses the highest security level for each accepted packet. As a result, the security values are increased while decreasing the guarantee ratio. On the other hand our proposed algorithm SPSS adaptively select the most appropriate security level of the accepted packet where both the guarantee ratio and the security level are increased which in turns increase the overall performance of the system significantly.

B. Performance metrics

To evaluate the performance of our approach, we compare SPSS algorithm against two baseline algorithms, namely, *MIN*, and *MAX*.

The following three important performance metrics are used to effectively evaluate the proposed algorithm. The Overall Perfomance *(OP)* is measured as the product of security level *(SL)* and the guarantee ratio *(GR)*. The following expression is used to calculate *OP*.

$$OP = SL * GR \tag{3}$$

where the security level is defined as the sum of security level values of all incoming packets issued to the network link. The guarantee ratio is defined as a fraction of total arrived incoming packets that are found to be delivered before their deadlines. Packet arrival rate λ, data sizes of packet, and deadlines are three workload parameters. We investigate performance impacts of these parameters on performance of a real-time wireless link in our simulation experiments.

C. Impacts of arrival rate

This experiment is aimed at comparing the SPSS strategy with the two baseline schemes that make no use of SPSS scheme. The first baseline scheme is called MIN which always assigns the minimum security level to the incoming packets while the other baseline which is called MAX always assigns the maximum security level to the incoming packets issued to the networked system. With different settings to of data size, bandwidth, and deadline, we study the impacts of varying arrival rates on the system performance. To achieve this goal, we increased the arrival rate of the incoming packets from 0.2 to 0.9 No./Sec. while setting the data size to 0.5 KB, the bandwidth to 0.7mbps, and the deadline to 0.7 No./Sec.

Figure 3 plots the security level, Figure 4 plots the guaranteed ratio, and Figure 5 plots the overall performance of the networked system with SPSS, Min, and Max schemes. Figure 3 reveals that the SPSS strategy can significantly increases the security level of the incoming packets. We can attribute this significant improvement to the fact that SPSS strategy increases the security level provided that it can meet the deadline requirement of the incoming packets.

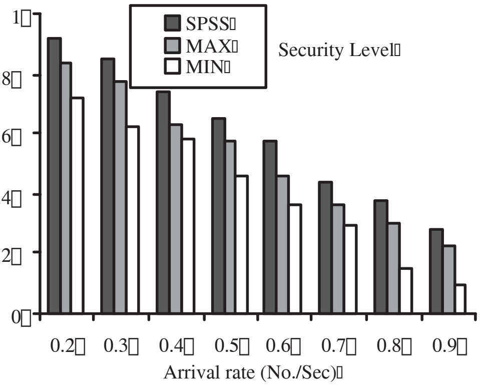

Fig. 3. Impact of arrival rate when data size = 0.5 KB Bandwidth = 0.7 MBPS, and deadline = 0.6 No/Sec.

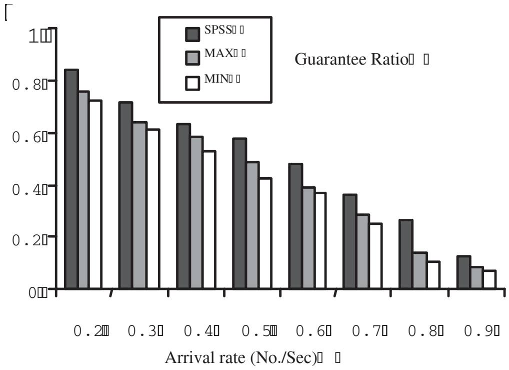

Fig. 4. Impact of arrival rate when data size = 0.5 KB, Bandwidth = 0.7 MBPS, and deadline = 0.6 No/Sec.

Fig. 4 reveals that SPSS outperforms both MIN and MAX algorithms in terms of guaranteed ratio. This result can be explained by the fact that SPSS algorithm keeps increasing the security level of the incoming packets while making the best effort to guarantee their deadlines. Fig. 5 clearly shows that SPSS clearly outperform both MIN and MAX algorithms in terms of overall performance magnificently. Specifically SPSS obtains an improvement in overall performance over MIN and MAX algorithm by an average of 9%. This performance improvement can be attributed by the fact that SPSS adaptively enhance security levels of each incoming packets under the condition that all incoming packets can meet their deadlines.

D. Impacts of deadline

In this section, we varied the deadline from 0.2 to 0.9 No./Sec to examine the performance impact of deadline on the networked system. Figure 6 shows that when the deadline increases from 0.2 to 0.9 No./Sec the security level performance metric increases. The main reason of this result is that when the incoming packet has a loose deadline, it will have more time to be delivered before its deadline which causes the security level to be increased. Clearly, SPSS has higher security level than MIN and MAX strategies. This

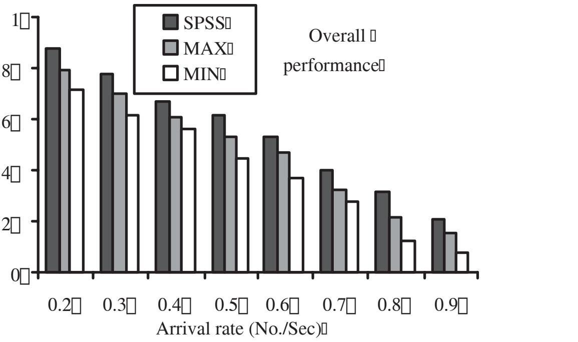

Fig. 5. Impact of arrival rate when data size = 0.5 KB, Bandwidth = 0.7 MBPS, and deadline = 0.6 No/Sec.

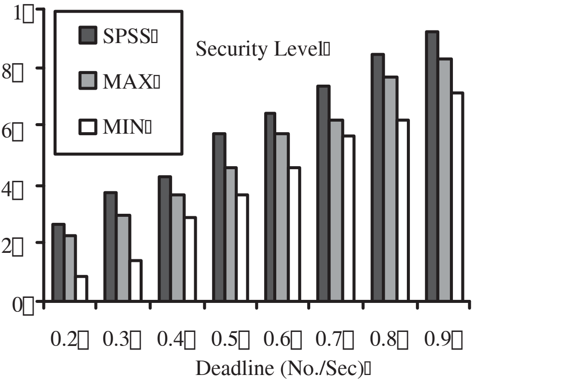

Fig. 6. Impact of deadline when data size = 0.5 KB, bandwidth = 0.7 MBPS, and arrival rate = 0.5 No/Sec.

can be explained by the fact that the looser the deadline for the incoming packets, the more opportunities for SPSS to dynamically increase the security level for each incoming packets.

Figure 7 shows that SPSS outperforms MIN and MAX in terms of guarantee ratio. The rationale behind this result is that when the incoming packet has a loose deadline, SPSS has more opportunities to deliver those packets before their deadlines thereby increasing the guarantee ratio. Because the security level and the guaranteed ratio rapidly increase, the overall performance of SPSS also goes up (see figure 8).

E. Impacts of data size

In this group of experiment, we compared the performance of SPSS against MIN and MAX strategies when we varied the data size from 0.2 to 0.9KB.

Figure 9 demonstrates that SPSS shows a significant improvement in security level over MIN and MAX strategies. Interestingly, it is also observed from figure 9 that the security level gradually drops as the value of data size increasing. This is because the increasing values of the data size results in an overload in the system which also turns in decreasing the values of the security level of the packets residing in the queue in order to finish most of the packets before their deadlines. Figure 10 shows that when the data size varied from 0.2 to 0.9KB the SPSS strategy delivers higher guarantee ratio that both MIN and MAX strategies. This result is consistent with the result in figure 5, which demonstrates that SPSS achieves good performance in guaranteed ratio. Further, we observe from figure 11 that when the data size goes up, the overall performance of SPSS decreases. This is because the security level of the incoming packets is lowered down due to the high load in the system, which results in decreasing the overall performance of SPSS.

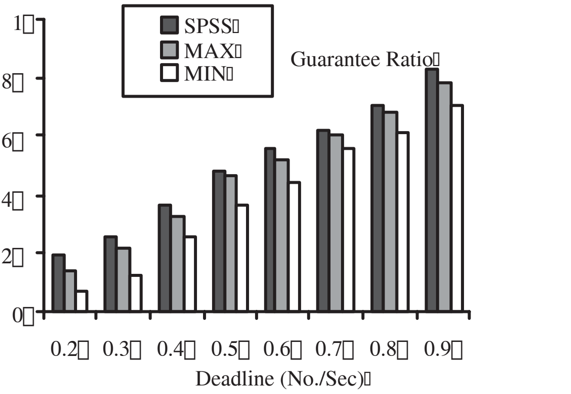

Fig. 7. Impact of deadline when data size = 0.5 KB, bandwidth = 0.7 MBPS, and arrival rate = 0.5 No/Sec.

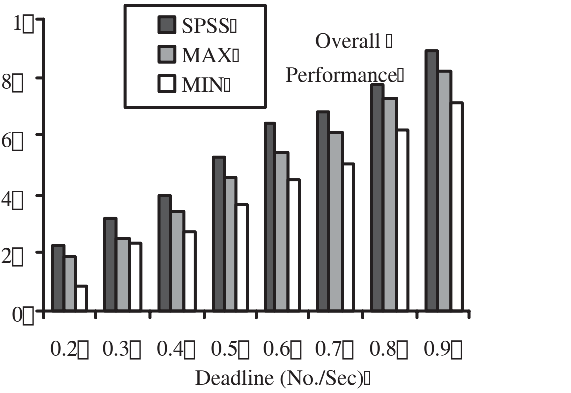

Fig. 8. Impact of deadline when data size =0.5 KB, bandwidth = 0.7 MBPS, and arrival rate = 0.5 No/Sec.

F. Impacts of bandwidth

In this experiment, we investigated the performance of SPSS, MIN, and MAX strategies when the network bandwidth varies from 0.2 to 0.9 MBPS.

Fig. 12 shows that the security level increases as the network bandwidth is increased, because more packets can pass through the network before their deadlines due to the short processing time of the packets over the network.

An important observation drawn from Fig. 13 is that as the network bandwidth increases, the guarantee ratios of the three strategies rise. The result can be explained by the fact that the high network bandwidth leads to short transfer times, which in turn result in short processing times of packets. Consequently, more packets can be passed through the network before their deadlines. Thanks to the increasing in the security level and the guarantee ratio, the overall performance of SPSS is substantially improved.

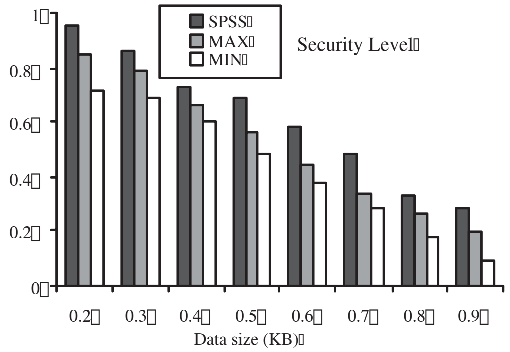

Fig. 9. Impact of data size when bandwidth = 0.7 MBPS, arrival rate = 0.5 No/Sec and deadline= 0.6 No./Sec.

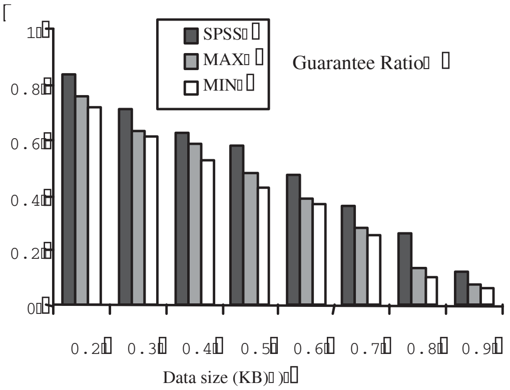

Fig. 10. Impact of data size when bandwidth = 0.7 MBPS, arrival rate = 0.5 No/Sec and deadline = 0.6 No/Sec.

V. Conclusion

In real-time wireless networks not only high guarantee ratio is required for packets, but also high quality of security is needed to protect data stored in the packets transmitted through wireless networks. To develop real-time wireless networks with high quality of security and guarantee ratio, we proposed a novel dynamic Security-Aware Packet Scheduling algorithm (or SPSS for short), which is capable of achieving high quality of security for real-time packets while making the best effort to guarantee real-time requirements of those packets. The SPSS algorithm is designed in a way that makes it possible to achieve a reasonably high guarantee ratio and optimized security level. In particular, our SPSS algorithm leverages an intelligent Security Level Controller to adaptively assign security levels to incoming real-time packets transmitted via a wireless network link. The experimental results show that our approach delivers significant improvements in guarantee ratio, security level, as well as overall system performance under a wide range of workload patterns. Specifically, our approach can provide overall performance improvement by up to 15%.

VI. Acknowledgment

The work reported in this paper was supported by the US National Science Foundation under Grants No. CCF-0742187 and No. CNS-0713895, Auburn University under a start up grant, and the Intel Corporation under Grant No. 2005-04-070.

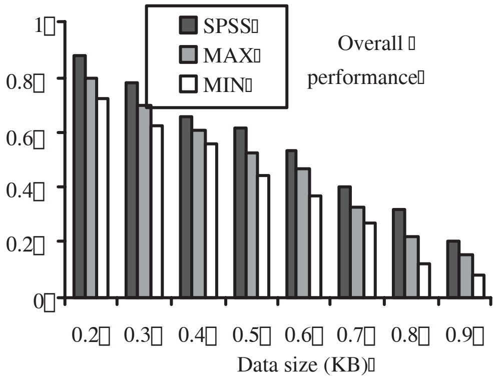

Fig. 11. Impact of data size when bandwidth = 0.7 MBPS, arrival rate = 0.5 No/Sec and deadline = 0.6 No/Sec.

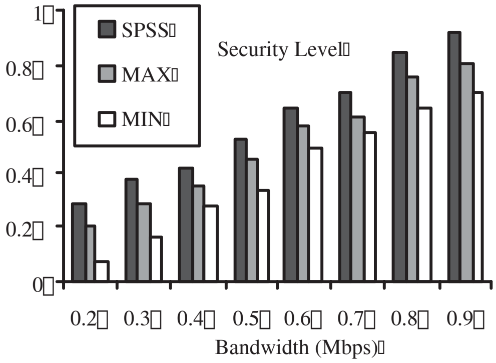

Fig. 12. Impact of bandwidth when data size = 0.6 KB, arrival rate = 0.6 No/Sec, and deadline = 0.6 No/Sec.

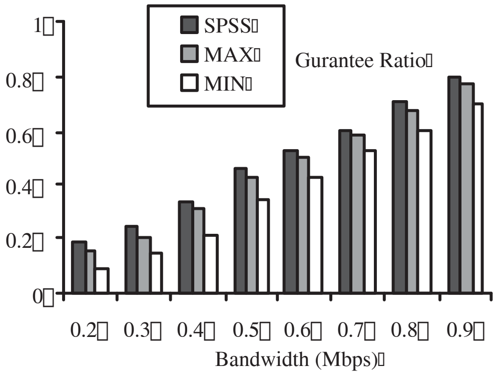

Fig. 13. Impact of bandwidth when data size = 0.6 KB, arrival rate = 0.6 No/Sec, and deadline = 0.6 No/Sec.

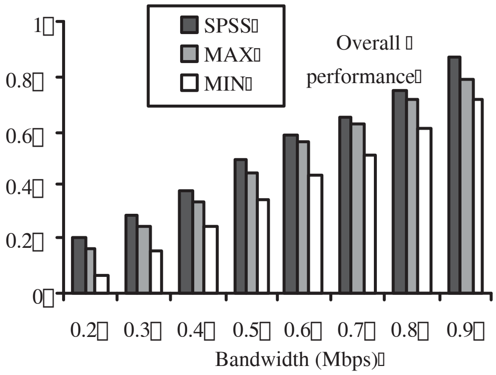

Fig. 14. Impact of bandwidth when data size = 0.6 KB, arrival rate = 0.6 No/Sec, and deadline = 0.6 No/Sec.

REFERENCES

[1] http://www.winbc.org/pdfs/WINBC_2005BC WirelessSurvey.pdf

[2] http://www.ntca.org/content_documents/2005 WirelessSurveyReport.pdf

[3] G. Donoho, "Building a Web service to provide real-time stock quotes," MCAD.Net, Feb. 2004.

[4] T. Karygiannis and L. Owens, "Wireless Network Security 802.11, Bluetooth and Handheld Devices," chapter 2, http://csrc.nist.gov/publications/nistpubs/ 800-48/NIST_SP_800-48.pdf

[5] White paper of Cisco, "Building the mobile business with a unified wireless network http://epsfiles.intermec.com/eps_files/ eps_wp/CISCO_SecuringWirelessLAN_wp_web.pdf

[6] C. Chang, J. Chang, K. Chen, and M. You, "Guaranteed quality-of-service wireless access to ATM," *IEEE J. Select. Areas Commun.*, 1997.

[7] S. Lu, V. Bharghavan, and R. Srikant, "Fair scheduling in wireless packet networks," *IEEE Trans. Networking*, Aug. 1999.

[8] V. Bharghavan, "A new protocol for medium access in wireless packet networks," online document, 1999.

[9] J. C. R. Bennett and H. Zhang, "WF2Q: worst-case fair weighted fair queueing," in *Proc. IEEE INFOCOM,* 1996.

[10] T. S. Ng, I. Stoica, and H. Zhang, "Packet fair queueing algorithms for wireless networks with location dependent errors," in *Proc. IEEE INFOCOM,* Mar. 1998.

[11] T. Nandagopal, T. Kim, X. Gao, and V. Bharghavan, "Achieving MAC layer fairness in wireless packet networks," in *Proc. ACM MOBICOM,* Boston, MA, Aug. 2000.

[12] M. A Badamas, "Mobile computing systems–security considerations," *Information Management and Security 2001*, pp. 134–136.

[13] V. Gupta and S. Gupta, "Securing the wireless Internet," *IEEE Commun. Mag.*, pp. 68–74, Dec. 2001.

[14] O. M. Karygiannis, *Wireless Network Security*, NIST special publication 800–48.

[15] K. Siau, E. P. Lim, and Z. Shen, "Mobile commerce: promises, challenges and research agenda," *J. Database Management*, vol. 12, pp. 4–19, July–Sept. 2001.

[16] X. Yu, D. B. Hoang, and D. Feng; "A QoS control protocol for rate-adaptive video traffic," in *Proc. 9th IEEE Int'l Conf. Networks,* pp. 434–438, Oct. 2001.

[17] X. Qin and H. Jiang, "Dynamic, reliability-driven scheduling of parallel real-time jobs in heterogeneous systems," in *Proc. Int'l Conf. on Parallel Processing,* pp. 113–122, 2001.

[18] X. Qin, H. Jiang, D. R. Swanson, "An efficient fault-tolerant scheduling algorithm for real-time tasks with precedence constraints in heterogeneous systems," in *Proc. Int'l Conf. on Parallel Processing*, British Columbia, Canada, pp. 360–368, Aug. 2002.

[19] J. C. Palencia and H. M. Gonzalez, "Schedulability analysis for tasks with static and dynamic offsets," in *Proc. 19th IEEE Real-Time Systems Symp. 1998*, pp. 26–37.

[20] T. F. Abdelzaher and K. G. Shin, "Combined task and message scheduling in distributed real-time systems," *IEEE Trans. Parallel and Distributed Syst.*, vol. 10, no. 11, Nov. 1999.

[21] M. A. Palis, "Online real-time job scheduling with rate of progress guarantees," in *Proc. 6th Int'l Symp. Parallel Architectures, Algorithms, and Networks 2002*, pp. 65–70.

[22] G. Manimaran and C. S. R. Murthy, "An efficient dynamic scheduling algorithm for multimachine real-time systems," *IEEE Trans. Parallel and Distributed Syst.*, vol. 9, no. 3, pp. 312–319, 1998.

[23] S. Al-Harthi and R. Rao, "A switch model for improving throughput and power fairness in Bluetooth piconets," in *Proc. Globecom 2003*, pp. 1279–1283.