2006

QoS Issues of Using Probabilistic Non-Repudiation Protocol in Mobile Ad Hoc Network Environment

Yi-Chi Lin
University of South Australia

Jill Slay
University of South Australia

Originally published in the Proceedings of 4th Australian Information Security Management Conference, Edith Cowan University, Perth, Western Australia, 5th December, 2006

QoS Issues of Using Probabilistic Non-Repudiation Protocol in Mobile Ad Hoc Network Environment

Yi-Chi Lin
Jill Slay
School of Computer & Information Science
University of South Australia
Adelaide, Australia
Linyy021@students.unisa.edu.au
Jill.Slay@unisa.edu.au

Abstract

So as to guarantee the fairness of electronic transactions, users may require a Non-Repudiation (NR) service in any type of network. However, most existing NR protocols cannot work properly in a Mobile Ad hoc Network (MANET) due to their characteristics (e.g. limited resources and lack of central authority). The design of the Probabilistic Non-Repudiation Protocol (PNRP) is comparatively suitable for the nature of a MANET, but it still poses some QoS issues. This article points out the QoS issues which are caused by using PNRP in a MANET environment. These issues explain the difficult of performing PNRP in such an infrastructure-less environment.

Keywords

Quality of Services, MANET, Non-Repudiation

INTRODUCTION

Many researchers are investigating security related issues in the area of Mobile Ad hoc Networks (MANETs) (Buttyan & Hubaux, 2003; Zhou & Haas, 1999) due to their characteristics (e.g. limited resources and lack of central authority). Several practical MANET applications (Guarnera et al. 2002) may need Non-Repudiation (NR) services. For example, military battlefield communications, emergency medical systems for accidents, conference communications, and network game applications. Battlefield communications lack a central authority (e.g. Desert Storm in 1991). Therefore, exchanged data packets can only be sent between nodes (soldiers). Furthermore, accidents, such as bushfires, may happen anywhere and all sorts of rescue mission are necessitated by natural disasters. Rescue teams cannot build infrastructure before an incident and their primary role is to help injured people. They need wireless communication ability as soon as possible. Each of these applications involves the transfer of critical data which has a paramount need for security. Based on this, the development of a system that protects and validates correct information without allowing false data through a network is vital.

The definition of NR is given by Internet Engineering Task Force (IETF) in (Shirey, 2000) as "a security service that provides protection against false denial of involvement in a communication." Two different types of NR protocols are developed nowadays, and they are: 1. NR protocols with a Trusted Third Party (TTP) (Zhou & Gollmann, 1997a; Onieva et al. 2003), and 2. NR protocols without a TTP (Even, Goldreich & Lempel, 1985; Syverson, 1998). The function of a TTP is to manage any possible disagreement between an originator and a recipient. Furthermore, A NR protocol uses two components (Zhou & Gollmann, 1997b), Non-Repudiation of Origin (NRO) and Non-Repudiation of Receipt (NRR), to ensure the fairness. An originator and/or a recipient is unable to take advantage of the other when he/she obtains an NRO or NRR which also can prevent malicious users from falsely denying previous actions. The concept of NRO/NRR evidence is the same as the idea of invoice in reality.

Based on the previous descriptions, it is obvious that a TTP acts as a central management authority in a NR protocol. The usage of a TTP hence violates the characteristic of a MANET. On the other hand, it seems that the

NR protocols without a TTP can easily work in a MANET. However, they usually require participating nodes to have equal computational ability or only keep the fairness of the protocols for a short period of time. These limitations bound the suitability for NR protocols without a TTP in a MANET.

Probabilistic Non-Repudiation Protocol (PNRP) proposed by Markowitch and Roggeman (Markowitch & Roggeman, 1999) is adopted due to no above limitations. Unfortunately, performing PNRP in a MANET still poses Quality of Service (QoS) issues. In this article, these QoS issues are determined, and examination of these QoS issues is made to test the feasibility of PNRP in a MANET.

RELATED WORK

The definition of QoS can be found in different sources. According to Consultative Committee for International Telephony and Telegraphy (CCITT) Recommendation E.800, QoS means "the collective effect of service performance which determines the degree of satisfaction of a user of a service".

Two basic ways to achieve QoS are resource reservation and traffic prioritization. Many QoS works (Sethi & Barua, 2003; Garg, Kappes & Mani, 2002; Hejmo et al. 2006; Ali, Inayatullah & Rotenstreich, 2004) have been done in the area of MANETs. They mainly focus on QoS routing issues, new resources management way, and security enhancement by QoS. However, this article puts emphasis on discussing QoS issues caused by using PNRP in a MANET environment rather than talking about QoS issues in conventional manners. Therefore, the discovered QoS issues can help readers to understand the feasibility of PNRP in a MNAET.

This paragraph gives the reason why PNRP is chosen to be the NR protocol in this paper rather than other NR protocols. According to the survey (Kremer, Markowitch & Zhou, 2002), most of current NR protocols (Zhou & Gollmann, 1997a; Onieva et al. 2003) need a TTP to be an adjudicator while they proceed. The function of a TTP is to decide which participant attempts to take advantages or to cheat in a transaction. However the usage of a TTP violates the nature of a MANET because there is no such node to be called on by either of the two participants in a transaction. Essentially every node acts selfishly in order to minimize the usage of resources. Nodes can not obtain any benefit from performing the role of a judge. These NR protocols therefore definitely conflict with the nature of a MANET.

Alternatively, other NR protocols which can work without the involvement of a TTP (Even, Goldreich & Lempel, 1985; Syverson, 1998; Markowitch & Roggeman, 1999) usually have individual requirements (e.g. equal computational ability for participants or fairness of the protocol is only kept for a short period of time), and these requirements make them unrealistic in a MANET. Based on the abovementioned reasons, existing NR protocols either can not work or have serious problems to implement in a MANET environment.

Furthermore, protocols for certified e-mail in mobile environment were proposed by Park et al. (Park et al. 2003) and by Wang et al. (Wang et al. 2005). The idea of the certified e-mail is similar to an e-mail with NR evidence. However, these approaches still do not meet the requirements of a MANET due to the involvement of a TTP. Additionally, Wang and Guo in (Wang & Guo, 2004) solved the fairness issue in a wireless environment. The idea of keeping fairness between two participants is similar to give them NR evidence to prevent cheating from occurring. The authors provided a method for fairness in a wireless environment. Nonetheless, a TTP is required to solve any disagreement which may occur as always. The involvement of a TTP makes the protocol inappropriate for the requirements of a MANET.

In summary, PNRP is adopted due to no such limitations of abovementioned works. Moreover, this article points out the relevant QoS issues of using PNRP in a MANET, and check its feasibility.

PNRP

PNRP (see Figure 1) is comprised of two parts: 1. sending message m, which is encrypted by symmetric session key K, to the receiver and 2. sending the symmetric session key K with its unique design. The main concept of PNRP is that a sender secretly chooses an integer n to be the number of total iterations of part 2 of PNRP

without telling the receiver. Therefore, the receiver is encouraged (forced) to finish the PNRP because he/she has no idea about when the last iteration is.

The $sSK_A(K)$ is sent in the nth iteration, and this information is considered NRO evidence. Additionally the sender sends $random_i$ in i^{th} iteration, where i=1~n-1. The length of $random_i$, where i=1~n-1, is the same as key K's length. The receiver hence is impossible to tell which iteration contains the actual key K unless he/she knows the chosen number n. If the receiver discovers the size of number n, he/she can obtain key K without sending the corresponding NRR evidence (ack_n). The fairness of PNRP is then broken. Therefore, number n is a critical parameter in PNRP. In this article, notation A/B denotes the sender/the receiver respectively. The detailed explanation will be described in the following section.

Following notations are employed in this paper.

A→B : m: party A sends message m to party B.

SK_P: secret key of party P.

PK_P: public key of party P.

$sSK_P(m)$: digital signature of message m with party P's secret key.

$ePK_P(m)$: encryption of message m with party P's public key.

K(m): symmetric key encryption of message m.

PART 1:

A→B: $sSK_A(K(m))$

B→A: $sSK_B(ack_m)$

PART 2:

1. A→B: $sSK_A(random_1)$
   B→A: $sSK_B(ack_1)$
2. A→B: $sSK_A(random_2)$
   B→A: $sSK_B(ack_2)$

.

n. A→B: $sSK_A(K)$

Figure 1: Procedures of PNRP

PNRP in MANET

Nodes in a MANET encounter different problems, such as limited resources and security issues, from nodes in Internet. The designs of applications in a MANET are affected by these problems, so does PNRP. If PNRP is directly implemented in a MANET environment, this implementation will lead to serious consequences (e.g. too much communication overhead, too many computational overheads, and more resources used). These consequences violate the nature of a MANET. Therefore, this article highlights the QoS issues triggered by using PNRP in a MANET, and makes readers clearly understand the feasibility of PNRP in a MANET.

QOS ISSUES OF PNRP IN A MANET

Basic concepts of QoS are introduced in the section 2. However the description of QoS can be interpreted in different levels. For example, what is QoS of a restaurant? From a broader viewpoint, people can say that every service (the service of a waiter, the service of a chef, and the service of a cashier) is good in this restaurant. On the contrary, from a narrower viewpoint, people can talk about the QoS of a particular employee. Therefore, the term QoS in here represents the QoS of PNRP rather than QoS of network.

The Value of Secret Number n

The purpose of number n is to be the number of total iterations in part 2 of PNRP. The value of n is chosen randomly and secretly. Hence, the receiver is encouraged (forced) to finish the PNRP if he/she intends to obtain message m. Number n is the critical and fundamental factor for PNRP to work.

According to (Adi & Sullivan, 2005), the failure possibility, 1/(n-1), is affected by the size of n. The detailed explanation of failure possibility will be given in the next section. According to the result, larger n makes sure the lower failure possibility. Moreover, another way for the receiver to break PNRP is guessing the secret number n (mentioned in section Ⅲ). Therefore, having larger integer n and keeping n secretly are necessarily in order to reduce the failure possibility. It is effortless for a sender to keep n secretly. However, larger integer n indicates more iterations. More iterations mean more overheads for two participants and the longer execution time of PNRP. This implies how to choose the correct size of n is a difficult task.

Based on abovementioned reasons, the size of n can neither too large nor too small. Larger n can decrease the failure possibility but causes more computational overheads. On the contrary, smaller n leads to completely opposite result. Therefore, it is a challenge for the sender to find the balance between large n and small n.

The Failure Possibility for PNRP

The reason why PNRP can work properly is that the receiver does not know the secretly chosen number n. Since the receiver has no idea when the last iteration (decided by n) is, he/she has no choice but to finish the PNRP correctly. However, if the receiver successfully guesses the number n, he/she can stop the PNRP at the last iteration without sending the final acknowledgement. The chance for the receiver to break PNRP is 1/(n-1). This is the critical flaw of PNRP which can not be improved because the design of PNRP compromises with no TTP. Therefore, the only thing the sender can do is to choose n relatively large to increase the chance of success. In this situation, the receiver is harder to guess the number n. However, this solution encounters the same problem as previous section.

Relation between the Chosen Number n and Number of Encryptions/Decryptions

The feature discussed in this section makes the computational overheads of participants even worse. For the PNRP to work properly encryptions/decryptions are needed in part 2 (see Figure 2). These encryptions/decryptions try to prevent illegal access to the transferred information. On the contrary, they severely affect the performance of PNRP. From the sender's viewpoint, only key K in n^{th} iteration needs to be encrypted. Nevertheless, the receiver needs to encrypt every ack_i since he/she does not realize which one is the actual key K. The intention of these encryptions/decryptions is to prevent other nodes from knowing the transferred data, and the number of them varies with the size of n. Therefore, the size of number n is a critical factor that affects the performance of PNRP.

1. $A \rightarrow B$: $sSK_A(random_1)$
   $B \rightarrow A$: $sSK_B(ePK_A(ack_1))$
2. $A \rightarrow B$: $sSK_A(random_2)$
   $B \rightarrow A$: $sSK_B(ePK_A(ack_2))$

.

n. $A \rightarrow B$: $sSK_A(ePK_B(K))$

Figure 2: Encrypted version of PNRP

Waste of Network Resource

A sender conveys $random_i$ in i^{th} iteration, where i = 1~n-1, and $ePK_B(K)$ in the last iteration in part 2 of PNRP. The lengths of these $random_i$, where i = 1~n-1, are the same as key K. Because the size of key K is normally from 128 bits to 2048 bits based on different symmetric encryption algorithms, neither the length of $random_i$, where i = 1~n-1, nor the length of key K exceeds 2048 bits.

Accordingly, the percentage of payload in a single frame is comparatively low. In a frame, the range of payload is from 0 byte ~ 2312 bytes. However, with the unique design of PNRP, the sender sends at most 2 bytes payload per frame in each iteration. This can not be improved because the design of PNRP has to compromise with no involvement of a TTP. Therefore, the sender can not accumulate several $random_i$, where i = 1~n-1, and sends them at once. An example is given to help readers to understand this situation.

Firstly, it assumes that secret number n=1000, and the size of each payload is 2 bytes. The sum of total payloads needed to be sent is 2000 bytes. Therefore, it theoretically can be sent in a single frame. However, due to the unique design of PNRP, the sender has to send these payloads one by one. The following problems occurred due to the requirement of PNRP. The sender needs to send 1000 frames rather than one frame to the receiver, and each frame contains only 2 bytes payload. The sender needs to perform 999 times more encapsulations; the ratio for the sender to send frames is 1000 to 1. Furthermore, every single frame needs its own header and Frame Check Sequence (FCS). This indicates the number of headers and FCSs increases 1000 times. Moreover, the size of the header and FCS in a single frame is much longer than the payload. In the example, the size of the payload is 2 bytes, but the size of the header and FCS is 34 bytes. The percentage of useful information per frame is 2/(2+34).

Therefore, most of network resources consumed by PNRP are used to send headers and FCSs. The useful information in each frame is comparatively smaller than the sum of the header and FCS. In summary, although PNRP overcomes the involvement of a TTP, the network resources are wasted with its unique design.

The Effect of Time Out issue

In this section, time out issue is going to be introduced, and the difference between time out and the delayed packet will be discussed as well.

Introducing Time Out issue

Time-out issue is one of significant factors of PNRP. Time out (see Figure 3) refers to the time period between a sender sends $random_i$ or $ePK_B(K)$, and receives corresponding $acknowledgement_i$ in a signal iteration. The purpose of time out prevents the receiver from cheating, and the time out period is entirely decided by the sender. If the sender does not receive the corresponding acknowledgement within the time out period, he/she considers that the receiver tries to cheat, and then stops performing the rest of PNRP iterations. The following paragraphs will discuss the consequences of long time out period and short time out period respectively.

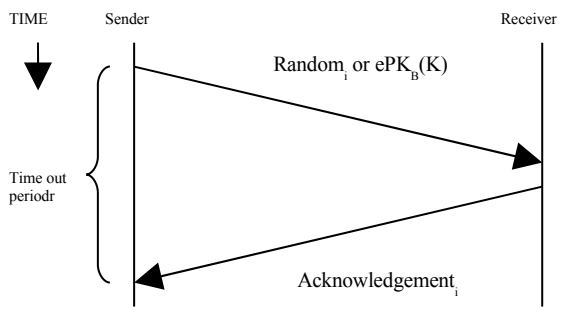

Figure 3: Explanation of Time Out Period

After the sender sends $random_i$ to the receiver at i^{th} iteration, the receiver can use the $random_i$ to test if it can decrypt K(m) as long as the time out period is long enough. Take 3rd iteration as an example, if time out is long enough, the receiver can try $random_3$ to check if it can decrypt K(m). The receiver can do this in every iteration, and PNRP will eventually fail in n^{th} iteration. Therefore the time out period can not be too long otherwise it gives the receiver a better chance to cheat.

On the contrary, the time out period can not be too short. This is because the sender needs to obtain the corresponding acknowledgement within the time out period; otherwise PNRP will be stopped to prevent possible cheating from happening. Therefore, if the time out is too short, the following problem will arise. The sender thinks the receiver tries to cheat because the corresponding acknowledgement does not come back within the time out period. Nevertheless, the real reason for this is because the time out value is too short for the receiver to respond. In this case, it is possible for the sender to misjudge the situation, and the possibility for the sender to stop PNRP is higher.

In conclusion, the value of time out has to be decided carefully. It can be neither too long nor too short. This value affects the QoS of PNRP in a MANET environment.

Delayed packet and Time out Issue

Due to the dynamic topology of a MANET, the path for a sender to send a packet in 2nd iteration may differ from the path in 1st iteration. Moreover, a receiver encounters the same problem. This means $acknowledgement_2$ may have a different path from $Acknowledgement_1$. Since the $acknowledgement_i$ may be delayed because of a different route rather than intention for cheating, it is difficult for a sender to determine the time period for time out. Additionally, it is unrealistic for a sender to tell the difference between delayed message and intention for cheating. Therefore, according to our understanding there is no solution to determine a proper value for the time out period of PNRP in a MANET environment.

DISCUSSION

The abovementioned discussions indicate five QoS issues of using PNRP in a MANET. Actually the first four of them are deeply associated to each other. They all related to the value of secret chosen number n. Since the number n is the unique design of PNRP in order to overcome the problems occurred by no involvement of a TTP, there is no better way to change or to modify the subsequences caused by the number n. Therefore, readers can understand that the first four QoS issues can not be improved in any way.

On the other hand, there exists a possible solution for the dilemma of the fifth QoS issue. This solution is using encryption methods (Garay & Jakobsson, 2002; Rivest, Shamir & Wagner, 1996; Mont, Harrison & Sadler, 2003) which only last for a period of time to solve time out issue. In this article, one of these methods, Time Release Crypto (TRC) (Rivest, Shamir & Wagner, 1996), is used as an example to explain the idea. Furthermore, this argument is supported by (Kremer, Markowitch & Zhou, 2002) as well.

The concept of TRC is that the corresponding receiver can not decrypt the encrypted message until a pre-decided amount of time has passed. The requirement of time out issue of PNRP can be satisfied straightforwardly. As long as the sender uses TRC to encrypt $random_i$, where i=1~n-1, and key K, then it gives the receiver not enough time to cheat. In this case, the receiver has only one choice which is to perform the PNRP properly. However, the subsequent shortcomings are:

1. The sender needs to perform TRC in every step in order to confuse the receiver. This is not a ideal way to perform PNRP. Furthermore this is not good for the sender himself/herself, since he/she needs to spend extra efforts for performing TRCs.
2. At the end of PNRP, the receiver needs to decrypt TRC in the last iteration (n^{th}) in order to obtain the actual key K, even if he/she play honestly in the previous n-1 iterations.

Based on these explanations, we can solve the time out issue but will cause another serious problem (poor performance). At this stage then, it seems there is no perfect solution for this situation and this is a topic which is in need of much further research.

CONCLUSION

In this paper, the possible QoS issues caused by performing PNRP in a MANET are described. Some of these effects are interrelated and due to the unique design of PNRP. This unique design has valuable strength; it allows

the function of PNRP without the involvement of a TTP. The weaknesses of PNRP have a considerable effect on QoS; these issues are unresolved and in need of further research.

REFERENCES

Adi, K., and Sullivan, L. (2005) . "Sufficient Conditions for Ensuring Probabilistic Fairness in Exchange Protocols", 7th International Symposium on Programming and Systems, Algiers.

Ali, H., Inayatullah, M., and Rotenstreich, S. (2004) . "Resource Allocation and QoS in Mobile Ad Hoc Networks", Proceedings of the 2004 international symposium on Information and communication technologies ISICT' 04, June.

Buttyan, L., and Hubaux, J. P. (2003) . "Report on A Working Session on Security in Wireless Ad Hoc Networks", ACM SIGMOBILE Mobile Computing and Communications Review, ACM Press, January, pp. 74-94.

Even, S., Goldreich, O., and Lempel, A. (1985) . "A Randomized Protocol for Signing Contracts", Communications of the ACM, ACM Press, pp. 637-647.

Garay, J. A., and Jakobsson, M. (2002) . "Timed Release of Standard Digital Signatures", Financial Crypto.

Garg, S., Kappes, M., and Mani, M. (2002) . "Wireless access server for quality of service and location based access control in 802.11 networks", Computers and Communications, 2002. Proceedings. ISCC 2002. Seventh International Symposium on, July, pp. 819-824.

Guarnera, M., Villari, M., Zaia, A., and Puliafito, A. (2002) . "MANET: Possible Applications with PDA in Wireless Imaging Environment", The 13th IEEE International Symposium on Personal Indoor and Mobile Radio Communications, Sept, pp. 2394-2398.

Hejmo, M., Mark, B. L., Zouridaki, C., and Thomas, R. K. (2006) . "Design and analysis of a denial-of-service-resistant quality-of-service signaling protocol for MANETs", Vehicular Technology, IEEE Transactions on, May, pp. 743 – 751.

http://www.webopedia.com/TERM/Q/QoS.html

Kremer, S., Markowitch, O., and Zhou, J. (2002) . "An Intensive Survey of Fair Non-Repudiation Protocols", Computer Communications Journal, pp.1606-1621.

Markowitch, O., and Roggeman, Y. (1999) . "Probabilistic Non-Repudiation without Trusted Third Party", Second Conference on Security in Communication Networks'99, Italy, Sept.

Mont, M. C., Harrison, K., and Sadler, M. (2003) . "Data intergrity: The HP time vault service: Exploiting IBE for Timed Release of Confidential Information", Proceedings of the 12th international conference on World Wide Web, May.

Onieva, J., Zhou, J., Carbonell, M., and Lopez, J. (2003) . "Intermediary Non-Repudiation Protocols", Proceedings of 2003 IEEE Conference on Electronic Commerce, June, pp. 207-214.

Park, J. M., Ray, I., Chong, E. K. P., and Siegel, H. J. (2003) . "A Certified E-Mail Protocol Suitable for Mobile Environment", 2003 IEEE Global Telecommunications Conference, Dec, pp. 1394-1398.

Rivest, R. L., Shamir, A., and Wagner, D. A. (1996) . "Time-lock puzzles and timed-release Crypto", MIT laboratory for Computer Science.

Sethi, P., and Barua, G. (2003) . "CRESQ: providing QoS and security in ad hoc networks", Parallel, Distributed and Network-Based Processing, 2003. Proceedings. Eleventh Euromicro Conference on, Feb, pp. 544-550.

Shirey, R. (2000) . "Internet Security Glossary", IETF, RFC 2828, May.

Syverson, P. (1998) . “Weakly Secret Bit Commitment: Applications to Lotteries and Fair Exchange”, Proceeding of the 1998 IEEE Computer Security Foundations Workshop, June, pp. 2-13.

Wang, G., Bao, F., Zhou, J., and Deng, R. (2005) . “An Efficient Certified E-Mail Scheme Suitable for Wireless Mobile Environments”, Proceedings of 2005 IEEE International Symposium on Personal Indoor and Mobile Radio Communications, Sept.

Wang, H., and Guo, H. (2004) . “Achieving fairness in wireless environment”, Proceedings of the 2004 IEEE 6th Circuits and Systems Symposium on Emerging Technologies: Frontiers of Mobile and Wireless Communication, 2004, pp. 117-120.

Zhou, J., and Gollmann, D. (1997a) . "An Efficient Non-Repudiation Protocol", Proceedings of 10th IEEE Computer Security Foundations Workshop, June, pp. 126-132.

Zhou, J., and Gollmann, D. (1997b) . "Evidence and Non-repudiation", Journal of Network and Computer Applications, July, pp. 267-281.

Zhou, L., and Haas, Z. J. (1999) . “Securing Ad Hoc Networks”, IEEE Network Magazine, pp. 24-30.

COPYRIGHT