International Journal of Computer Science and Network (IJCSN)
Volume 1, Issue 6, Decembers 2012 www.ijcsn.org ISSN 2277-5420

Secure Public Key Protocol for Ad-Hoc Wireless Networks

[1]Nazeer Unnisa Nazima, [2]Shahana Tanveer , [3]Abdul Majeed

[1] IT, Maulana Azad National Urdu University
Hyderabad, A.P, India

[2] CSE, Lords Institute of Engineering & Technology
Hyderabad, A.P, India

[3] CSE, Lords Institute of Engineering & Technology
Hyderabad, A.P, India

Abstract

As part of the security within distributed systems, various services and resources need protection from unauthorized use. Remote authentication is the most commonly used method to determine the identity of a remote client. This paper investigates a systematic approach for authenticating clients by three factors, namely password, smart card, and biometrics. A generic and secure framework is proposed to upgrade two-factor authentication to three-factor authentication. In multi hop wireless networks, selfish nodes do not relay other nodes' packets and make use of the cooperative nodes to relay their packets, which has negative impact on the network fairness and performance. Incentive protocols use credits to stimulate the selfish nodes' cooperation, but the existing protocols usually rely on the heavyweight public-key operations to secure the payment. In this paper, we propose secure cooperation incentive protocol that uses the public-key operations only for the first packet in a series and uses the lightweight hashing operations in the next packets, so that the overhead of the packet series converges to that of the hashing operations.

Keywords— *Network Security, Mobile, Wireless Networks*

1. Introduction

In multi hop wireless networks, selfish nodes do not relay other nodes' packets and make use of the cooperative nodes to relay their packets, which has negative impact on the network fairness and performance. Incentive protocols use credits to stimulate the selfish nodes' cooperation, but the existing protocols usually rely on the heavyweight public-key operations to secure the payment. In this paper, we propose secure cooperation incentive protocol that uses the public-key operations only for the first packet in a series and uses the lightweight hashing operations in the next packets, so that the overhead of the packet series converges to that of the hashing operations. Cooperation incentive protocols can be classified as tamperproof-device (TPD), electronic coin, and central-bank-based protocols. For TPD-based protocols, a tamper-proof device (which cannot be tampered) is installed in each device to store its credits and secure its operation. For electronic-coin-based protocols, a network node buys electronic coins in advance from a centralized accounting center (AC) to pay for relaying its packets. In central-bank based protocols, the intermediate nodes usually compose undeniable receipts and submit them to the AC to update their accounts. In Nuglets, the self-generated and forwarding packets are passed to the tamper-proof device to decrease and increase the credit account, respectively. Two models, called the packet purse model (PPM) and the packet trade model (PTM) have been proposed. In the PPM, the source node pays by loading some credits in the packet, and each intermediate node acquires its payment from the packet. In the PTM, each intermediate node buys the packets from the downstream node and sells them to the upstream nodes and thus the destination node pays the total cost. In CASHnet, for each data packet, the source node's credit account is charged and its signature is attached. The destination node sends back a digitally signed ACK packet to increase the intermediate nodes' credit accounts. The extensive use of digital signature operations for both the data and the ACK packets is not efficient for limited-resource nodes. For SIP, after receiving a packet, the destination node sends back a receipt to the source node that issues a REWARD packet which increments the intermediate nodes' credit accounts. In this paper, we propose Efficient and Secure cooperation Incentive Protocol (ESIP) that uses public-key operations only for the first packet in a series, and uses the efficient hashing operations in the next packets. Security analysis

and performance evaluation demonstrate that the proposed protocol is secure and the overhead is incomparable to the signature-based incentive protocols because the hashing operations dominate the nodes' operations. Moreover, these protocols can also be used for billing the network services without contacting a distant home network register. However, secure incentive protocols usually use signatures to achieve payment no repudiation which is important to prevent payment manipulation and to thwart free riding attacks because the message's integrity is checked at each node in the route. These cryptosystems incur too heavy overhead to be used efficiently in limited-resource nodes.

2. Previous Work

Authentication ensures that a system's resources are not obtained fraudulently by illegal users. Password based authentication is one of the most simple and convenient authentication mechanisms over insecure networks. In 1981, Lamport proposed a remote password authentication scheme by employing a one-way hash chain, which Haller later used to design the famous S/KEY one-time password system. However, one weakness of their scheme is that a verification table should be maintained on the remote server in order to validate the legitimacy of the requesting users; if an intruder can somehow break into the server, the contents of the verification table may be easily modified. Therefore, many password authentication schemes have recognized this problem, and solutions based on smart cards have been proposed, where a verification table is no longer required.

In a typical smart card based password authentication scheme, remote users are authenticated with their smart cards as identification tokens. The card takes as input a password from the user, creates a login message from the given password, and sends the message to a remote server, which then checks the validity of the login message before allowing access to any services or resources. This way the administrative overhead of the authentication server is reduced, and the user only needs to remember his password.

Recently, some biometrics-based remote user authentication schemes have been designed. In 2002, Lee et al. proposed a fingerprint-based scheme using smart cards. It is based on ElGamal's public key cryptosystem, which also does not require password table for authentication. The scheme is novel in that biological information and two secret keys are employed to improve the security.

However, Lin et al. and Ku et al. pointed out in 2004 and 2005 respectively that Lee et al.'s scheme cannot withstand the masquerade attack, in which an adversary can impersonate a legitimate user without knowing the password and passing the fingerprint verification. Later, in ISPEC 2006, Khan et al. also showed that Lee et al.'s scheme was vulnerable to the server spoofing attack. Furthermore, they proposed an improved scheme to enhance the security. Based on the one-way hash function and fingerprint verification, Khan et al.'s scheme needs only to maintain one secret key, and a password verification table is not required on the server. They claimed that their scheme achieved mutual authentication between the user and the server, and thus eliminated the drawback of Lee et al.'s scheme. [1]

User		Server
	[Registration]	
Select ID, PW	$\{ID, PW, F\}$ $\longrightarrow$	$A = h(ID \oplus x)$ $V = A \oplus h(PW \oplus F)$
	$\longleftarrow$ $(Smart\ card)$	Store $\{ID, A, V, F, h(\cdot)\}$
	[Login and Authentication]	
Input ID, PW^* Imprints fingerprint $B = V \oplus h(PW^* \oplus F)$ $B \stackrel{?}{=} A$		
$C_1 = h(B \oplus T)$	$m = \{ID, C_1, T\}$ $\longrightarrow$	Verify ID, T $C_1 \stackrel{?}{=} h(h(ID \oplus x) \oplus T)$
Verify T'' $C_2 \stackrel{?}{=} h(B \oplus T'')$	$\longleftarrow$ $\{C_2, T''\}$	$C_2 = h(h(ID \oplus x) \oplus T'')$

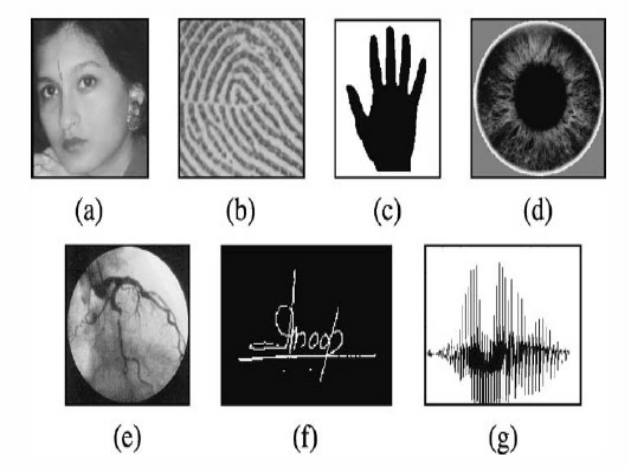
Content owners (such as authors and authorized distributors) are losing billions of dollars annually in revenues due to illegal copying and sharing of digital media. Digital rights management (DRM) systems are being deployed to address this problem. The user authentication, which is an essential part of a DRM system, determines whether a user is authorized to access the content. In a generic cryptographic system the user authentication is possession based. That is, possession of

the decrypting key is a sufficient evidence to establish user authenticity. Because cryptographic keys are long and random, (e.g., 128 bits for the advanced encryption standard (AES), they are difficult to memorize. As a result, the cryptographic keys are stored somewhere (for example, on a computer or a smart card) and released based on some alternative authentication (e.g., password) mechanism, that is, upon assuring that they are being released to the authorized users only. Most passwords are so simple that they can be easily guessed (especially based on social engineering methods) or broken by simple dictionary attacks.

It is not surprising that the most commonly used password is the word "password"! Thus, the multimedia protected by the cryptographic algorithm is only as secure as the passwords (weakest link) used for user authentication that release the correct decrypting key(s). Simple passwords are easy to crack and, thus, compromise security; complex passwords are difficult to remember and, thus, are expensive to maintain.1 Users also have the tendency to write down complex passwords in easily accessible locations. [2]

Further, most people use the same password across different applications and, thus, if a single password is compromised, it may open many doors. Finally, passwords are unable to provide non repudiation; that is, when a password is shared with a friend, there is no way to know who the actual user is. This may eliminate the feasibility of countermeasures such as holding conniving legitimate users accountable in a court of law.

Many of these limitations of the traditional passwords can be ameliorated by incorporation of better methods of user authentication. Biometric authentication refers to verifying individuals based on their physiological and behavioral characteristics such as face, fingerprint, hand geometry, iris, keystroke, signature, voice, etc. It is inherently more reliable than password-based authentication, as biometric characteristics cannot be lost or forgotten (cf. passwords being lost or forgotten); they are extremely difficult to copy, share, and distribute (cf. passwords being announced in hacker websites) and require the person being authenticated to be present at the time and point of authentication (cf. conniving users denying having shared the password). [3]



It is difficult to forge biometrics (it requires more time, money, experience, and access privileges) and it is unlikely for a user to repudiate having accessed the digital content using biometrics. Finally, one user's biometrics is no easier to break than another's; that is, all users have a relatively equal security level, hence, there are not many users who have "easy to guess" biometrics, that can be used to mount an attack against them. Thus, biometrics-based authentication is a potential candidate to replace password-based authentication, either by providing the complete authentication mechanism or by securing the traditional cryptographic keys that contain the multimedia file in a DRM system. [3]

Biometric Matcher

For various reasons mentioned in the earlier section, unlike password or keys, the exact match of biometric identifiers is not very useful. Typically, a practical biometric matcher undoes some of the variations in the biometric measurements to be matched by *aligning* them with respect to each other. Once the two representations are aligned, an assessment of their similarity is measured based on acceptable variations within the aligned representations and is typically quantified in terms of a *matching score*; the higher the matching score, the more similar are the representations.

Let us consider a concrete example of fingerprint matching. The most widely used local features (ridge ending and ridge bifurcation) are based on minute details (*minutiae*) of the fingerprint ridges. The pattern of the minutiae of a fingerprint forms a valid, compact, and robust representation of the fingerprint and it captures a significant component of information in fingerprints. The simplest of the minutiae-based representations constitute a list of triplets, where represents the spatial coordinates in a fixed image-centric coordinate system and represents the orientation of the ridge at that minutia. Typically, a good-quality live-scan fingerprint image has 20–70 minutiae.

Only in the highly constrained fingerprint systems could one assume that the input and template fingerprints depict the same portion of the finger and both are aligned (in terms of displacement from the origin of the imaging coordinate system and of their orientations) with each other; given two (input and template) fingerprint representations, the matching module typically aligns the input and template minutiae and determines whether the prints are impressions of the same finger by identifying *corresponding* minutiae within an acceptable spatial neighborhood of the aligned minutiae. The number of corresponding minutiae is an effective measure of similarity between the matched prints. Fig. 5 illustrates a typical matching process. Even in the best of practical situations, *all* minutiae in input and template prints are rarely matched due to spurious minutiae introduced by dirt/leftover smudges, variations in the area of finger being imaged, and displacement of the minutia owing to distortion of the print from pressing the elastic finger against the flat surface of the acquisition device.[2]

The biometrics authentication system offers several advantages over other security methods. Passwords might be divulged or forgotten, and smart cards might be shared, lost, or stolen. In contrast, personal biometrics, such as fingerprints or iris scans, have no such drawbacks. It is ideally suited for both high security and remote authentication applications due to the nonreturnable nature and user convenience.

Remote authentication is a form of e-authentication in which user credentials, as proof of identities, are submitted over a network connection. Remote authentication poses unique security challenges given its open, uncontrolled and unsupervised nature. There are two problems in applying personal biometricsto remote authentication. One of the most important is obtaining easily some biometric characteristics, so that the results can never be changed. Another is the difficulty of checking whether the device is capable of verifying that a person is alive since the biometric capture devices are remotely located. Because of such problems, the best approach is to integrate biometrics with passwords and smart cards to construct a secure three-factor authentication scheme. Several three-factor authentication schemes have been proposed in the literature.

In 2010, based on the one-way hash function, biometrics verification and smart card, proposed an efficient biometric-based remote user authentication scheme, in which the computation cost is relatively low compared with other related schemes. Recently, showed that Li and Hwang's scheme neither provides proper authentication nor resists the man-in-the-middle attacks. They then presented an improved scheme to fix the problem. In above schemes, the user chose a random number *RC*, and computed $M2 = h(IDijjXS) \copyright RC$ for the output of user login phase. In this article, we show that $h(IDijjXs)$ can easily be obtained by an attacker obtaining an obsolete value of *RC*. Then, without user's password and personal biometrics, the attacker can succeed in either impersonating the user or obtaining the session key. In these schemes, once the template *fi* is leaked, the biometrics authentication is facing a dilemma of how to identify a forgery. In addition, they suffer from replay attacks and DoS attacks. We remedy this situation by suggesting an enhanced scheme. We also demonstrate how the enhanced scheme is efficient. Furthermore, the security of the enhanced scheme will be demonstrated by formal proofs.[3]

The rapid progress of networks facilitates more and more computers connecting together to exchange great information and share system resources . Security is then an important issue for computer networks. Entity authentication is one of the most important security

services . It is, necessary to verify the identities ofthe communication parties when they start a connection.

The concept of ID-based cryptosystems was first proposed by Shamir . The ID-based cryptosystems have the following advantages: neither secret nor public keys need be exchanged, the public key directory table is not needed, and the assistance of a trusted third party is not needed. The secret key corresponding to an ID is fixed and cannot be changed in Shamir's ID based scheme. Therefore, a user with an assigned ID cannot choose his password by himself.

Actually, a user's password is generated by the password generation center, rather than by the user himself. However, users are used to choosing their own passwords . This approach is against other users' habits .

Based on ElGama1's signature and Shamir's ID-based schemes, the concept of timestamps is used in Wang et al. scheme and Lee et al. scheme. These schemes are all based on ID-based schemes; they share the problem that a user cannot change his password after registration. A user could not use his current ID but needs to choose a new one after his password is compromised. Lee et al. proposed fingerprint-based remote user authentication scheme using smart cards based on a synchronized system clock. Time-stamp based authentication scheme can withstand the attack of replaying previously intercepted messages using the systems' timestamp. However, the scheme requires system clock synchronization otherwise the scheme will not work properly. Since network environment and transmission delay is unpredictable, a potential replay attack exists in all schemes that employ the concept of timestamps. [4]

The problem of identity theft, that is, the act of impersonating others' identities by presenting stolen identifiers or proofs of identities, has been receiving increasing attention because of its high financial and social costs. Recent federated digital identity management systems if on one side have improved the management of identity information and user convenience; on the other side do not provide specific solutions to address identity theft. One approach to such problem is the adoption of biometric *identification* and *authentication* systems. These systems are automated methods for recognizing an individual based on some physical characteristics, such as fingerprints, voice, or facial features.
Biometric identification and authentication are differentiated as follows. Biometric identification occurs when an individual provides a sample biometric, sometimes without any additional knowledge, and the system must compare that sample with every stored record to identify a match. This is known as a one-to-many match, and is executed without any corroborating data. By contrast, biometric authentication occurs when an individual presents a biometric sample, and some additional identifying data, such as a photograph or password, which is then compared with the stored sample for that individual. Biometric authentication provides some inherent advantages as compared to other non-biometric identifiers since biometrics correspond to a direct evidence of the personal identity versus possession of secrets which can be potentially stolen. Moreover, most of the times biometric enrollment is executed in-person and in controlled environments making it very reliable for future use.

Biometric authentication poses however several non-trivial security challenges because of the inherent features of the biometric data itself. Addressing these challenges is crucial for the large scale adoption of biometric authentication and its integration with other authentication techniques and with access control systems.

Biometric matching is probabilistic in nature, which implies that two samples of the same individual are never exactly the same. If the two samples are encrypted for security reasons, they need to be decrypted before they can be matched. This raises the issue of key management to enable decryption, and also represents a point of vulnerability in the process. Moreover, it is very hard to revoke and change biometrics in case biometric data are compromised. At the time of enrollment or verification the individuals biometric is read as a template, that is, is a binary file created using distinctive information from a biometric sample, which is then stored in a database or on a token. These templates are often vendor-specific and

therefore the interoperable use of such templates in a distributed system is very difficult if at all possible.

Biometric authentication from a remote location also represents a difficult issue because of the risk of spoofing attacks. The credibility of the output from a biometric matching process depends entirely on the integrity of the sample provided, and whether it was provided by the true owner of the biometric. Older generation biometric capture devices were vulnerable to spoofing attacks, and there is extensive work currently in the area of biometric capture devices to able to withstand different spoofing attacks.

Biometric authentication can be implemented through systems performing the matching either on the *server* or on the *client side*. Depending on whether the matching of the biometric template is executed - at the server or at the client - different security problems arise. In the former case the main issues are related with the large scale and distributed management of biometric templates. The creation of a database of a particular biometric at the server should itself be secure and possibly decentralized. Also, such database would be highly dependent on a particular software or hardware and thus could not be interoperable. Such a system is also CPU intensive because of the matching operations.

Additionally, storing biometric information in repositories along with other personally identifiable information raises several security and privacy risks. These databases are vulnerable to attacks by insiders or external adversaries and may be searched or used outside of their intended purposes. It is important to note that if the stored biometric identifiers of an individual are compromised, there will be severe consequences for the individual because of the lack of revocation mechanisms for biometrics.

Due to the security and privacy problems of server side matching, there have been several efforts in biometric authentication technology using client side matching. Such an approach is convenient as it is relatively simple and cheap to build biometric authentication systems supporting biometric storage at the client end able to support local matching. Nevertheless, systems of such type are not secure if the client device is not trusted; therefore additional cryptographic support is needed.

3. Proposed System

A. Setup Phase

In this module each node stores a unique identity and public/private key pair with a certificate, the public key of the AC, and the required cryptographic data for the key exchange protocol. Each node in a session has to share a symmetric key with the source node to compute the messages' keyed hash values. For efficient implementation, an identity-based key exchange protocol based on bilinear pairing can be used because the nodes do not need to exchange messages to compute the shared keys. The AC generates a prime p, a cyclic additive group (G), and a cyclic multiplicative group of the same order p such that an efficiently computable bilinear pairing. The source node initiates route establishment by broadcasting Route Request Packet (RREQ) that contains its identity (IDS), time stamp (TS), and the identity of the destination node (IDD) and the time to live (TTL). If the time stamp is within a proper range and the TTL is not zero, a network node decrements the TTL, appends its identity, and broadcasts the packet.

B. Data Generation

In this module, the source node initiates a packet series with maximum size by attaching its signature to the identities of the session nodes, TS, and VNS. This signature proves the source node's approval to pay for the session and authenticates its hash chain and links it to the session, i.e., the sender cannot deny generating the hash chain or initiating the session. In order to ensure the hop-by-hop message authenticity and integrity, the message's hash value can be included in the signature but with increasing the receipt size. Therefore, the source node attaches the hash series which contains a truncated keyed hash value for each node. Each intermediate node verifies the source node's signature to ensure that it will be

rewarded for relaying the packets. Then, it verifies its message's truncated hash value to ensure the message authenticity and integrity and relays the packet after dropping its hash value. Each intermediate node saves the source node's signature and VNS to be used in the receipt composition.

C. ACK Generation

Entity In this module, after receiving a data packet, the destination node sends back ACK packet containing a fresh hash value from its hash chain as an approval to pay for the message. Each intermediate node verifies that V_D is generated from hashing, and saves the last hash value to be used in the receipt composition.

D. Receipt and Payment Redemption

In this module if the session is broken after receiving the first data packet, the intermediate nodes compose a receipt for receiving one packet. The payment data includes the identities of the payers and payees (R), the time of the transaction (TS), and the roots of the payers' hash chains. The security token is the hash value of the source and destination nodes' signatures. Attaching the hash of the signatures instead of the signatures can reduce the receipt size significantly. The security token can guarantee that the receipt is undeniable and unforgeable. Since the session is broken before receiving the ACK, the last released hash value from the destination node is V_D. If the last received packet is the ACK, the receipt is composed which is a proof for successfully delivering X messages.

In this module the network nodes periodically submit the receipts to the AC to redeem them. Once the AC receives a receipt, it first checks that the reeipt has not been deposited before using the receipt's unique identifier, i.e., the identities of the nodes in the route and the establishment time. Then, the AC verifies the credibility of the receipt by generating the source and the destination nodes' signatures, and matching the signatures' hash value with the receipt's security token. Finally, the AC counts the packets' number from the hash chains' elements, and clears the receipt according to the rewarding and charging policy.

4. Results

The concept of this paper is implemented and different results are shown below, the proposed paper is implemented in Java technology on a Pentium-IV PC with 20 GB hard-disk and 256 MB RAM with Java Environment. The propose paper's concepts shows efficient results and has been efficiently tested on different Datasets. The Fig 1, Fig 2, Fig 3 and Fig 4 shows the real time results compared.

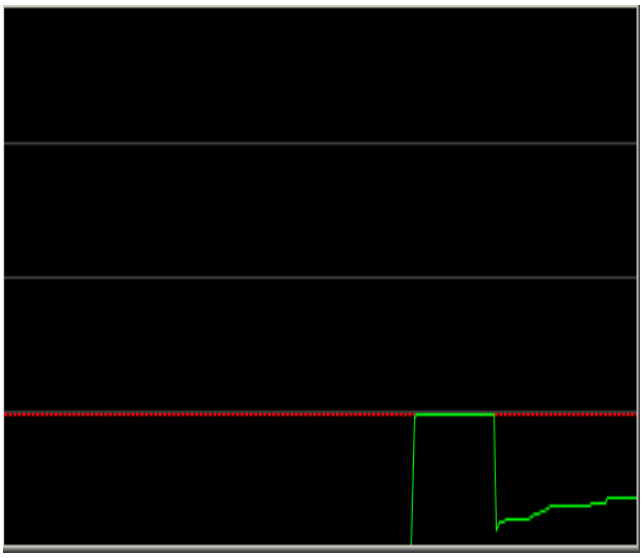


Fig. 1 Time taken by Node to initialize.

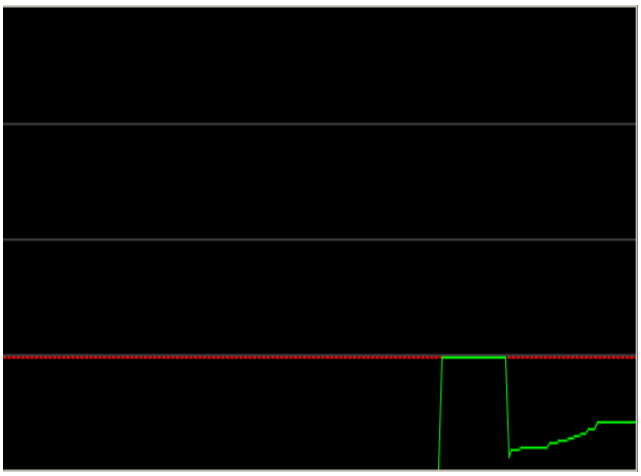


Fig. 1 Time taken by Users to initialize

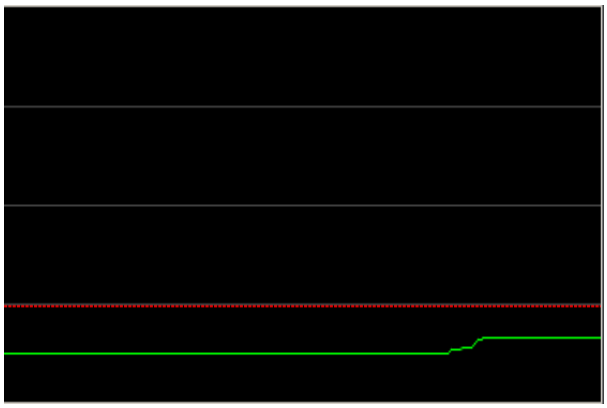


Fig. 3 Time taken by Phase Distribution

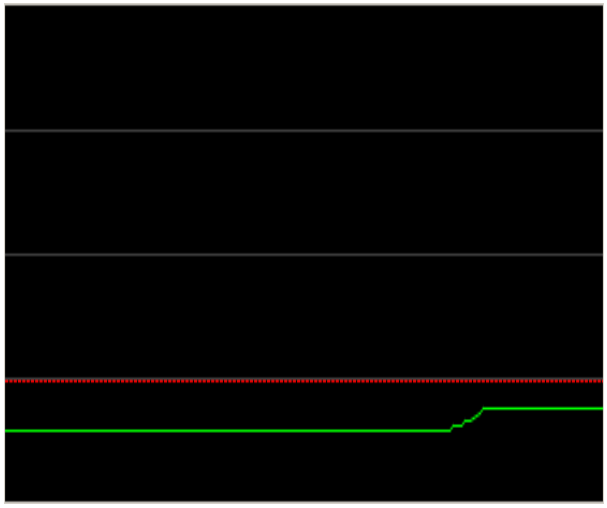


Fig. 4 Time taken by Key Generation

5. Conclusion

In this paper, we have proposed secure cooperation incentive protocol with limited use of public-key cryptography for multi hop wireless networks. The public-key operations are required only for the first packet and the efficient hashing operations are used in the next packets, so for a series of packets, the heavy overhead of the first packet vanishes and the overall overhead converges to that of the lightweight hashing operations. Our security analysis and performance evaluations have demonstrated that ESIP can secure the payment and improve the network performance significantly because the hashing operations dominate the nodes' operations.

Reference

[1] G. Shen, J. Liu, D. Wang, J. Wang, and S. Jin, "Multi-Hop Relay for Next-Generation Wireless Access Networks," Bell Labs Technical J., vol. 13, no. 4, pp. 175-193, 2009.

[2] X. Li, B. Seet, and P. Chong, "Multihop Cellular Networks: Technology and Economics," Computer Networks, vol. 52, no. 9, pp. 1825-1837, June 2008.

[3] A. Abdrabou and W. Zhuang, "Statistical QoS Routing for IEEE 802.11 Multihop Ad Hoc Networks," IEEE Trans. Wireless Comm., vol. 8, no. 3, pp. 1542-1552, Mar. 2009.

[4] Y. Jiang, M. Shi, X. Shen, and C. Lin, "BAT: A Robust Signature Scheme for Vehicular Networks Using Binary Authentication Tree," IEEE Trans. Wireless Comm., vol. 8, no. 4, pp. 1974-1983, Apr. 2009.

[5] P. Gupta and P. Kumar, "The Capacity of Wireless Networks," IEEE Trans. Information Theory, vol. 46, no. 2, pp. 388-404, Mar. 2000.

[6] S. Marti, T. Giuli, K. Lai, and M. Baker, "Mitigating Routing Misbehavior in Mobile Ad Hoc Networks," Proc. IEEE/ACM MobiCom, pp. 255-265, Aug. 2000.

[7] D.B. Johnson, D.A. Maltz, and Y.C. Hu, "The Dynamic Source Routing Protocol for Mobile Ad Hoc Networks (DSR)," technical report, IETF MANET Working Group, Feb. 2007.

[8] P. Michiardi and R. Molva, "Simulation-Based Analysis of Security Exposures in Mobile Ad Hoc Networks," Proc. European Wireless Conf., Feb. 2002.

[9] J. Hu, "Cooperation in Mobile Ad Hoc Networks," Technical Report TR-050111, Computer Science Dept., Florida State Univ., Jan. 2005.