World Applied Sciences Journal 29 (9): 1123-1128, 2014
ISSN 1818-4952

DOI: 10.5829/idosi.wasj.2014.29.09.1408

Efficient Multicast Authentication Batch Signature

P. Gayathri and A. Jeyamathi

Department of Information Technology,
Bharath University, Chennai, 600073, India

Abstract: Multicast [1] is defined as sending data from a single sender to a group of receivers. While multicasting the data, it should be provided with security services. Generally, security services are classified into five areas namely data integrity, data origin, authentication [2], non-repudiation and confidentiality. This project work supports the first three security services, which are implemented using a cryptographic technique called batch signature [3]. Designing a batch signature protocol is quite difficult and hence certain issues have to be considered, which include efficiency, packet loss, computation latency and communication overhead. In internet, normally congestion is sure to happen at certain period which results in packet loss [4] have to be reduced. For applications like online streaming, video conferencing and online games, the quality of services is low to the end users. Therefore, MABS protocol should provide a certain level of packet loss recovery. In this paper, we have proposed two schemes namely, basic scheme and enhanced scheme. The basic scheme normally supports three algorithms namely batch RSA algorithm [5], batch BLS algorithm [6] and our batch DSA algorithm [7, 8], in which these algorithms mainly involves to address efficiency and packet loss problems in network environment [9]. Enhanced scheme combines basic scheme and merkle tree generation [10] (packet filtering technique) which completely eliminates the forged packets in order to overcome the DOS (Denial of Service) attack. Hence, our comparison study predicts that MABS-E is less efficient than MABS-B due to DOS defense. In our new approach, mod-inversion operation is one of the most important factors which influence the computation time of digital signature and verification, thus improvements to this point has been proposed by implementing new approach called PDSA.

Key words: Packet loss · DSA algorithm · Batch BLS algorithm · MABS-B

INTRODUCTION

The purpose of this software specification (SS) is to establish the major requirements & Specification necessary to develop this Software Systems for the Developers. The overall objective of the Team Project is to establish a Client Server Oriented project. The goal of this document is the same as any requirements document, to lie out all requirements of the application in order to have both the developers and the end users maintaining the same understanding and expectations from the application. The project requirements will define, in general terms, the setup of the web site, topics for available information concerning the Software Project Management.. This project work supports the first three security services, which are implemented using a cryptographic technique called batch signature [3].

System Design: System architecture describes about the overall communication between the sender and the receiver. This overall system architecture includes four phases namely,

- Packetization Phase.
- Key Generation Phase.
- Signature Generation Phase.
- Signature Verification Phase.

Correspoding Author: P. Gayathri, Department of Information Technology, Bharath University, Chennai, 600073, India.

Packetization Phase: In this phase, sender sends the message as input which has been divided into 'n' number of packets, where 'n' denotes number of packets. Thus, packets are sent to signature generation phase for hashing.

Key Generation Phase: In this phase, sender generates both the public key and private key, in which private key is used of signing the packets in encryption process, whereas public key is used for verifying the packets in decryption process [11]. Thus, the public key of the sender is stored in Central Authority (CA) where, it's responsible for creating key pair, distributing the private key, publishing the public key and revoking the keys when necessary [12]. In spite of public and private key generation, it also generates some common system parameters which has been passed as an input to signature generation phase.

Signature Generation Phase: In this phase, sender performs three operations namely, hashing, encryption and multicast. First in hashing operation, incoming packets are hashed to produce the hash code by using hashing algorithm. Secondly, in encryption operation, the hash code generated from the previous operation along with sender's private key is used to generate the signature of the message. Finally, signature along with message has been multicast to the group of receivers through multicast operation.

Signature Verification Phase: In this phase, receiver first computes the hashed code from message using same hash algorithm which is used in sender's side. In decryption process, receiver decrypts the signature using sender's public key issued by certificate authority. Thus, the hash code of the message and the signature has been compared and verified [13]. If the verification succeeds, the message is authenticated and it will be delivered to the group of receivers. If the verification fails, receiver simply drops the packet, which leads to less latency, communication and computational overhead.

The above four phases form the overall system architecture for PDSA protocol which includes three algorithms RSA [14], BLS [10], DSA [1] and our proposed algorithm PDSA. Thus, all the three algorithms perform above four phases which produces different computation cost depends on signing and verification time taken by each algorithm at the sender and receiver side [15]. Next section includes the detailed explanation for all three algorithms of basic scheme and our proposed algorithm.

Existing Algorithms:

Batch RSA Algorithm: Some system parameters are defined as:

- P,Q – two large prime number.
- To calculate $N = P * Q$ and $\varphi(N) = (P\text{-}1)\ (Q\text{-}1)$
- d, the private key of sender.
- e, the public key of sender, $ed \equiv 1 mod \quad \varphi(N)$.
- m, message as input.

Signature Generation: Sender first computes hash code h=h(m) and sign the hash code using sender's private key, which leads to the generation of the signature as σ.

Compute $\sigma = (h(m))^d \ mod\ N$

Signature Verification: The receiver verify the signature by checking whether signature is equal to hash code of the message using sender's public key, which is mathematically represented as,

$\sigma^e = h(m)\ mod\ N$

Signature Verification for Batch of Messages: The receiver can verify the batch signature by first computing h=h (m_i), hash code of 'n' number of packets and then checking whether batch signature is equal to the hash code of the message.

$$\left(\coprod_{i=1}^{n}\sigma_i\right)^e \bmod N = \prod_{i=1}^{n} h_i \bmod N$$

This is because, if the batch of packets is authentic, then mathematically defined

$$\left(\coprod_{i=1}^{n}\sigma_i\right)^e \bmod N = \prod_{i=1}^{n}\sigma_i^e \bmod N$$

$$= \prod_{i=1}^{n}\sigma_i^{ed} \bmod N, = \prod_{i=1}^{n} h_i \bmod N$$

Message will be delivered to the receiver. Suppose if the batch of packets is unauthenticated, receiver simply drops the packet. Thus, if the sender does not have enough resource, a pair of (e, d) with comparable sizes can achieve a certain level of trade-off between computation efficiency and security at the sender part.

Proposed Algorithm: Some system parameters are defined as:

- p, a prime longer than 512 bits.
- q, a 160-bit prime divisor of p-1
- g, a generator of $\square_p^*$ with order q, i.e. $g^p = mod\ p$
- x, the private key of the signer, $0 < x < q$
- y, the public key of the signer, $y = g^x\ mod\ p$.
- h( ), a hash function generating an output in $\square_q^*$

Signature Generation:

- Randomly selecting an integer k with $0 < k < q$
- Computing $h = h(r,m)$
- Computing $r = (g^p\ mod\ p)\ mod\ q$.
- Computing $S = (k - hx)\ mod\ q$.

The signature for message '*m'* is (*r, s*).

Signature Verification: The receiver can verify the signature by first computing $h = h(r,m)$ and then checking whether verification signature is equal to r.

$((g^s y^h)\ mod\ p)\ mod\ q = r$

Mathematical proof for signature verification

$((g^s y^h)\ mod\ p)\ mod\ q) = r$
$= ((g^s y^h)\ mod\ p)\ mod\ q)$
$= ((g^s y^{hx})\ mod\ p)\ mod\ q)$
$=((g^k\ mod\ p)\ mod\ q = r$

Signature Verification for Batch of Message: The receiver can verify the batch signatures by first computing $h_i = h(r_i m_i)$, i=1,….n and then checking whether batch verification signature is equal to batch of r(signatures),

$$\left(\left(g^{\sum_{i=1}^{n} s_i} y^{\sum_{i=1}^{n} h_i}\right) \bmod p\right) \bmod q = \prod_{i=1}^{n} r_i \bmod q$$

This is because, if the batch of packets is authentic, then mathematically defined

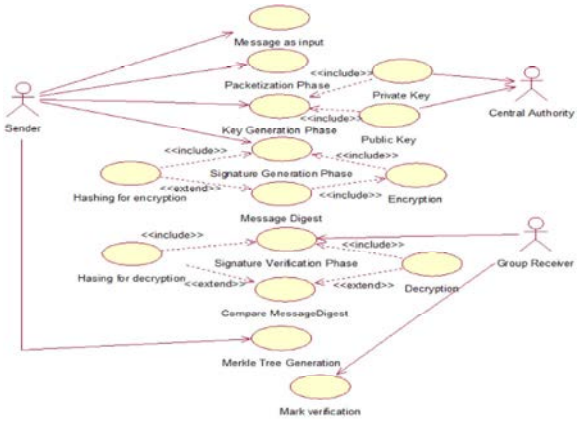
$$\left(\left(g^{\sum_{i=1}^{n} s_i} y^{\sum_{i=1}^{n} h_i}\right) \bmod p\right) \bmod q, \left(\left(g^{\sum_{i=1}^{n} s_i} g^{\sum_{i=1}^{n} h_x}\right) \bmod p\right) \bmod q$$

$$\left(\left(g^{\sum_{i=1}^{n} (s_i + h_i x)}\right) \bmod p\right) \bmod q, \left(g^{\sum_{i=1}^{n} k_i} \bmod p\right) \bmod q = \prod_{i=1}^{n} r_i \bmod q$$

The idea proposed mainly concerned with eliminating a multiplication and an inversion operation there by, we have reduced computation time compared with traditional digital signature algorithm which shows the improvement in both signing and verification process respectively. While dealing with security issue, it has been overcome by hashing the signature(r) along with message so that attacker cannot hack the signature in network environment. Thus, the security services like authenticity and data integrity has been achieved with good improvement in our proposed approach.

UML Diagrams

Usecase Diagram:



In our algorithm, Sender sends the message as input. At sender side, there exist three phases in which message is divided into number of packets in packetization phase. The public key is generated with the help of private key in key generation phase. In key generation phase, then signature is generated by computing hash value and encrypting the hash value using private key which result is message digest at signature generation phase. At receiver side, signature verification takes place in which it includes calculation of message digest from message and signature is decrypted with the help of sender's public key which results in message digest. Then the message is authenticated if it succeeds in verification.

State Chart Diagram: State chart diagram gives brief explanation about each process in which it has start and stop activity. The data flow describes about the flow of data that happens sequentially. The process includes packetization phase, Key generation phase, merkle tree generation at sender side and signature verification phase mark verification phase takes place at receiver side. Finally the receiver receives the message which is authenticated.

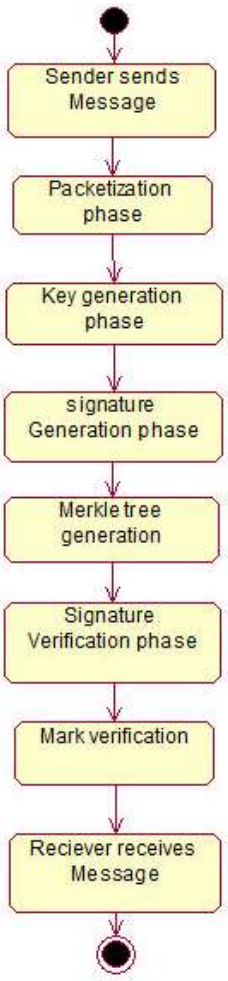


Fig. 6.1: 6 State Chart Diagram of PDSA Protocol

Dataflow Diagrams: Dataflow diagrams mainly include three levels namely level 0, level 1and level 2. In DFD, level 0 usually includes the overall view about multicast Authentication System. The rectangle box indicates source and destination, in which they are depicted as sender and receiver. The rounded rectangle Multicast Authentication System indicates the process. The data flow line indicates the data in motion.

In Level 1 DFD, the diagram gives brief explanation about our algorithm. The sender and receiver are source and destination of this diagram. The rounded rectangle indicates process which includes five processes namely Key generation, signature generation, signature verification and also merkle tree construction and mark verification for enhanced scheme. The horizontal open rectangle indicates the data store which is depicted as D1, D2 and d3 for Key store Data and signature respectively. The data flow line indicates the data in motion. Thus, the overall process of this project work has been represented in this level 1 data flow diagram. Individual process of each phase is explained in level 2 Data flow diagram. For instance we have explained the receiver side process as signature verification phase at the level 2 Data flow diagram.

In Level 2 DFD, it explains about detailed description about the important module in our algorithm. Here we explained about the verification process in detail. In verification process, the receiver first computes message digest through hashing. Then receiver decrypts the signature using sender's public key in which it leads same message digest as computed before. Thus message is authenticated. In case of merkle tree, root value is recovered for verification of messages.

CONCLUSION

In this paper, we have developed a novel authentication [14] schemes namely, MABS which is perfectly resilient to packet loss due to the elimination of correlation among packets and also effectively deals with DOS attack. Efficiency in delivering the packets also improved using batch signatures. Finally, we have implemented batch signatures schemes based on DSA [16], which are more efficient than batch RSA signature schemes [17]. Efficiency is improved mainly by using single signature verification at the receiver side. Further we have developed to improve the efficiency by reducing the signature verification time. Apart from Harn batch DSA and our batch DSA, we have developed a new scheme PDSA which has improved the computation overhead of signing and verification process at the sender and the receiver side by eliminating a multiplication and mod-inversion operation and also it achieves the same security level achieved by our batch DSA by hashing the signature along with its message which prevents packets forgery from the attackers in lossy channels [18]. Finally, we conclude that our proposed batch digital signature algorithm is both efficient and high secure in network environment [19]. As is well known that existing digital signature algorithms are computationally expensive, the ideal approach of signing and verifying each packet independently raises a serious challenge to resource-constrained devices which can be overcome by our protocol. We also made an analysis study on latency in which our proposed idea is more efficient when compared to HDSA and ODSA. Thus, we conclude that our analysis study about signing &verification time and latency on

Proposed Batch Signature Scheme (PDSA) is having an better improvement in efficiency and packets forgery when compared to the other existing approach Harn Batch DSA(HDSA) and Our Batch DSA(ODSA) [20-24].

REFERENCES

1. Lam, S.S. and W.K. Wong, 1999. Digital Signatures for Flows an Multicasts,' IEEE/ACM Trans. Networking, 7(4): 502-513.
2. Canetti, R. Perrig D. Song and J.D. Tygar, 2000. Efficient Authentication and Signing of Multicast Streams over Lossy Channels, Proc. IEEE Symp. Security and Privacy (SP '00), pp: 56-75.
3. Cui, S., C.W. Chan and P. Duan, 2006. An Efficient Identity- Based Signature Scheme with Batch Verifications, Proc. First Int'l Conf. Scalable Information Systems, pp: 246-250.
4. Deering, S.E., 1988. Multicast Routing in Internetworks and Extended LANs', Proc. ACM SIGCOMM Symp. Comm. Architectures and Protocols, pp: 55-64.
5. Harn, L., 1998. 'Batch Verifying Multiple RSA Digital Signatures',IEEE Electronic Letters, 34(12):1219-1220.
6. Boneh, D., B. Lynn and H. Shacham, 2001. Short Signatures from the Weil Pairing, Proc. Seventh Int'l Conf. Theory and Application of Cryptology and Information Security Advances in Cryptology (ASIACRYPT '01), pp: 514-532.
7. Lim, C.H. and P.J. Lee, 1994. Security of Interactive DSA Batch Verification, IEEE Electronic Letters, 30(19): 1592-1593.
8. Bellare, M., J.A. Garay and T. Rabin, 1998. Fast Batch Verification forModular Exponentiation and Digital Signatures, Proc. Advances in Cryptology (EUROCRYPT '98), pp: 236-250.
9. Lysyanskaya, A. R. Tamassia and N. Triandopoulos, 2004. Multicast Authentication in Fully Adversarial Networks, Proc. IEEE Symp. Security and Privacy (SP '04), pp: 241-253.
10. Miner, S. and J. Staddon, 2001. Graph-Based Authentication of Digital Streams', Proc. IEEE Symp. Security and Privacy (SP '01), pp: 232-246.
11. Udayakumar, R., V. Khanna, T. Saravanan and G. Saritha, 2013. Retinal Image Analysis Using Curvelet Transform and Multistructure Elements Morphology by Reconstruction, Middle-East Journal of Scientific Research, ISSN: 1990-9233, 16(12): 1798-1800.
12. Udayakumar, R., V. Khanna, T. Saravanan and G. Saritha, 2013. Cross Layer Optimization For Wireless Network (Wimax), Middle-East Journal of Scientific Research, ISSN: 1990-9233, 16(12): 1786-1789.
13. Thooyamani, K.P., V. Khanaa and R. Udayakumar, 2013. An Integrated Agent System for E-mail Coordination using Jade, Indian Journal of Science and Technology, ISSN: 0974-6846, 6(6): 4758-4761.
14. Chong, E.K.P., J.M. Park and H.J. Siegel, 2003. Efficient Multicast Stream Authentication Using Erasure Codes, ACM Trans. Information and System Security, 6(2): 258-285.
15. Thooyamani, K.P. V. Khanaa and R. Udayakumar, 2013. A frame work for modelling task coordination in Multi-agent system, Middle-East Journal of Scientific Research, ISSN:1990-9233, 15(12): 1851-1856.
16. Naccache, D., D. M'Raihi, S. Vaudenay and D. Raphaeli, 1995. Complexity Trade Offs with the Digital Signature Standard, Proc. Workshop Theory and Application of Cryptographic Techniques Advances in Cryptology (EUROCRYPT '94) pp: 77-85.
17. Adleman, L., R.L. Rivest and A. Shamir, 1978. A Method for Obtaining Digital Signatures and Public-Key Cryptosystems, Comm. ACM, 21(2): 120-126.
18. Yun Fang, Yuguang and Zhou, 2007. Multicast Authentication Over Lossy Channels, Military Communication Conference, pp: 1-7.
19. Lam, S.S. and C.K. Wong, 1998. Digital Signatures for Flows and Multicasts, Proc. Sixth Int'l Conf. Network Protocols (ICNP '98), pp: 198-209.
20. Sibghatullah Nasir, 2013. Microfinance in India Contemporary Issues and Challenges. Middle-East Journal of Scientific Research, 15(2): 191-199.
21. Mueen Uddin, Asadullah Shah, Raed Alsaqour and Jamshed Memon, 2013. Measuring Efficiency of Tier Level Data Centers to Implement Green Energy Efficient Data Centers, Middle-East Journal of Scientific Research, 15(2): 200-207.
22. Hossein Berenjeian Tabrizi, Ali Abbasi and Hajar Jahadian Sarvestani, 2013. Comparing the Static and Dynamic Balances and Their Relationship with the Anthropometrical Characteristics in the Athletes of Selected Sports, Middle-East Journal of Scientific Research, 15(2): 216-221.

23. Anatoliy Viktorovich Molodchik, 2013. Leadership Development. A Case of a Russian Business School, Middle-East Journal of Scientific Research, 15(2): 222-228.

24. Meruert Kylyshbaevna Bissenova and Ermek Talantuly Nurmaganbet. The Notion of Guilt and Problems of Legislative Regulations of its Forms. The Notion of Guilt in the Criminal Law of Kazakstan, Middle-East Journal of Scientific Research, 15(2): 229-236.