Journal of Engineering and Applied Sciences 9 (3): 55-60, 2014
ISSN: 1816-949X

Multicast Authentication Based on Batch Signature

[1]R.M. Kumaresh Babu and [2]M. Anand
[1]Tagore Engineering College,
[2]Bharath University, Chennai, India

Abstract: An overlook the heterogeneity of receivers by letting the sender choose the block size, divide a multicast stream into blocks, associate each block with a signature and spread the effect of the signature across, all the packets in the block through hash graphs or coding algorithms. The correlation among packets makes them vulnerable to packet loss which is inherent in the internet and wireless networks. Moreover, the lack of Denial of Service (DoS) resilience renders most of them vulnerable to packet injection in hostile environments. In this study, researchers propose a novel multicast authentication protocol, namely MABS, including two schemes. The basic scheme (MABS-B) eliminates the correlation among packets and thus, provides the perfect resilience to packet loss and it is also efficient in terms of latency, computation and communication overhead due to an efficient cryptographic primitive called batch signature which supports the authentication of any number of packets simultaneously. Researchers, also present an enhanced scheme MABS-E which combines the basic scheme with a packet filtering mechanism to alleviate the DoS impact while preserving the perfect resilience to packet loss.

Key words: Multimedia, multicast, authentication, signature, DoS

INTRODUCTION

Designing a multicast authentication protocol for efficient secure data transfer process. Efficiency needs to be considered, especially for receivers. Compared with the multicast senderwhich could be a powerful server, receivers can have different capabilities and resources. The receiver heterogeneity requires that the multicast authentication protocol be able to execute on not only powerful desktop computers but also resource-constrained mobile handsets. In particular, latency, computation and communication overhead are major issues to be considered.

Packet loss is inevitable. In the internet, congestion at routers is a major reason causing packet loss. An overloaded router drops buffered packets according to its preset control policy. Though, TCP provides a certain retransmission capability, multicast content is mainly transmitted over UDP which does not provide any loss recovery support. In mobile environments, the situation is even worse. The instability of wireless channel can cause packet loss very frequently. Moreover, the smaller data rate of wireless channel increases the congestion possibility. This is not desirable for applications like real-time online streaming or stock quotes delivering. End users of online streaming will start to complain if they experience constant service interruptions due to packet loss and missing critical stock quotes can cause severe capital loss of service subscribers. Therefore, for applications where the quality of service is critical to end users, a multicast authentication protocol should provide a certain level of resilience to packet loss. Specifically, the impact of packet loss on the authenticity of the already-received packets should be as small as possible. Efficiency and packet loss resilience can hardly be supported simultaneously by conventional multicast schemes.

Digital signature algorithms are computationally expensive, the ideal approach of signing and verifying each packet independently raises a serious challenge to resource-constrained devices. In order to reduce computation overhead, conventional schemes use efficient signature algorithms or amortize one signature over a block of packets at the expense of increased communication overhead or vulnerability to packet loss.

Someone vulnerable to packet injection by malicious attackers. An attacker may compromise a multicast system by intentionally injecting forged packets to consume receivers resource, leading to Denial of Service (DoS). Compared with the efficiency requirement and packet loss problems, the DoS attack is not common but it is still important in hostile environments.

A novel multicast authentication protocol, namely MABS including 2 schemes. The basic scheme (MABS-B) eliminates the correlation among packets and thus,

Corresponding Author: R.M. Kumaresh Babu, Tagore Engineering College, Chennai, India

provides the perfect resilience to packet loss and it is also efficient in terms of latency, computation and communication overhead due to an efficient cryptographic primitive called batch signature which supports the authentication of any number of packets simultaneously.

The batch signature schemes can be used to improve the performance of broadcast authentication. The comprehensive study on this approach and propose a novel multicast authentication protocol called MABS (in short for Multicast Authentication based on Batch Signature). MABS includes 2 schemes. The basic scheme (MABS-B) utilizes an efficient asymmetric cryptographic primitive called batch signature which supports the authentication of any number of packets simultaneously with one signature verification, to address the efficiency and packet loss problems in general environments. The enhanced scheme (MABS-E) combines MABS-B with packet filtering to alleviate the DoS impact in hostile environments. MABS provides data integrity, origin authentication and non repudiation.

Related study: Schemes in Even *et al.* (1996) and Rohatgi (1999) follow the ideal approach of signing and verifying, each packet individually but reduce the computation overhead at the sender by using one-time signaturesor k-time signatures. They are suitable for RSA which is expensive on signing while cheap on verifying. For each packet, however each receiver needs to perform one more verification on its one-time or k-time signature plus one ordinary signature verification. Moreover, the length of one-time signature is too long (on the order of 1,000 bytes). Tree chaining was proposed in Wong and Lam (1998, 1999) by constructing a tree for a block of packets. The root of the tree is signed by the sender. Each packet carries the signed root and multiple hashes. When each receiver receives one packet in the block, it uses the authentication information in the packet to authenticate it. The buffered authentication information is further used to authenticate other packets in the same block. Without the buffered authentication information, each packet is independently verifiable at a cost of per-packet signature verification.

A multicast stream is divided into blocks and each block is associated with a signature. In each block, the hash of each packet is embedded into several other packets in a deterministic or probabilistic way. The hashes form a graph, in which each path links a packet to the block signature. Each receiver verifies the block signature and authenticates all the packets through the paths in the graph. Erasure codes were used in Park *et al.* (2002), Pannetrat and Molva (2003) and Wu and Li (2006).

A signature is generated for the concatenation of the hashes of all the packets in one block and then is erasure-coded into many pieces. Erasure codes make each receiver be capable of recovering the block signature when receiving at least a certain number of pieces.

All these schemes are indeed computationally efficient since each receiver needs to verify only one signature for a block of packets (Wong and Lam, 1998, 1999; Perrig *et al.*, 2000; Park *et al.*, 2002; Pannetrat and Molva, 2003; Wu and Li, 2006). However, they all increase packet overhead for hashes or erasure codes and the block design introduces latency when buffering many packets. Another major problem is that most schemes (Perrig *et al.*, 2000; Park *et al.*, 2002; Pannetrat and Molva, 2003; Wu and Li, 2006) are vulnerable to packet loss, even though they are designed to tolerate a certain level of packet loss. If too many packets are lost, other packets may not be authenticated.

In particular, if a block signature is lost, the entire block cannot be authenticated. Moreover, previous schemes (Even *et al.*, 1996; Rohatgi, 1999; Wong and Lam, 1998, 1999; Perrig *et al.*, 2000; Park *et al.*, 2002; Pannetrat and Molva, 2003; Wu and Li, 2006) target at lossy channels which are realistic in the daily life, since the internet and wireless networks suffer from packet loss. In a hostile environment, however an active attacker can inject forged packets to consume receivers resource, leading to DoS. In particular, schemes in Even *et al.* (1996), Rohatgi (1999), Wong and Lam (1998, 1999) and Perrig *et al.* (2000) are vulnerable to forged signature attacks because they require each receiver to verify each signature whereby to authenticate data packets and schemes in Park *et al.* (2002), Pannetrat and Molva (2003) and Wu and Li (2006) suffer from packet injection because each receiver has to distinguish a certain number of valid packets from a pool of a large number of packets including injected oneswhich is very time-consuming. In order to deal with DoS, schemes in Jeong *et al.* (2005), Karlof *et al.* (2004) and Lysyanskaya *et al.* (2004) were proposed. PARM is similar to the tree chaining scheme (Wong and Lam, 1998, 1999) in the sense that multiple one way hash chains are used as shared keys between the sender and receivers. Schemes in Jeong *et al.* (2005) combine erasure codes with one-way hash chains to detect forged packets. Unfortunately, these schemes (Jeong *et al.*, 2005) are still vulnerable to DoS because they require that one-way hash chains are signed and transmitted to each receiver and therefore, an attacker can inject forged signatures for one-way hash chains. PRABS (Karlof *et al.*, 2004) uses distillation codes to deal with DoS. In particular, valid packets and forged packets are partitioned into disjoint sets and erasure decoding is performed over

each set. BAS simply retransmits each signature multiple times to tolerate packet loss and uses selective verification to tolerate injected forged signatures. LTT (Lysyanskaya *et al*., 2004) uses error correction codes to replace erasure codes in schemes (Park *et al*., 2002; Pannetrat and Molva, 2003; Wu and Li, 2006). The reason is that error correction codes tolerate error packets. These three schemes (Karlof *et al*., 2004; Lysyanskaya *et al*., 2004) are resilient to DoS but they still have the packet loss problem. Some other schemes (Desmedt *et al*., 1992; Perrig *et al*., 2001) use shared symmetric keys between the sender and receivers to authenticate multicast streams. Though, they are more efficient than those using signatures, they cannot provide non repudiation as the signature approach and they require either time synchronization or trustful infrastructures. In this study, researchers focus on the signature approach. Though, confidentiality is another important issue for securing multicast, it can be achieved through group key management. In this study, researchers focus on multicast authentication.

System architecture: The MABS can achieve perfect resilience to packet loss in lossy channels in the sense that no matter how many packets are lost the already-received packets can still be authenticated by receivers. MABS-B is efficient in terms of less latency, computation and communication overhead. Though, MABS-E is less efficient than MABS-B since it includes the DoS defense, its overhead is still at the same level as previous schemes.

About 2 new batch signature schemes based on BLS and DSA and show they are more efficient than the batch RSA signature scheme.

The sender attaches each packet with a mark which is unique to the packet and cannot be spoofed. At each receiver, the multicast stream is classified into disjoint sets based on marks. Each set of packets comes from either the real sender or the attacker. The mark design ensures that a packet from the real sender never falls into any set of packets from the attacker and vice versa. Next, each receiver only needs to perform batch verify over each set. If the result is true, the set of packets is authentic. If not, the set of packets is from the attacker and the receiver simply drops them and does not need to divide the set into smaller subsets for further batch verification. Therefore, a strong resilience to DoS due to injected packets can be provided.

The architecture of the proposed system is shown in Fig. 1. The entire flow of this project is as follows: Multicasting server, multicast clients, multicast router,

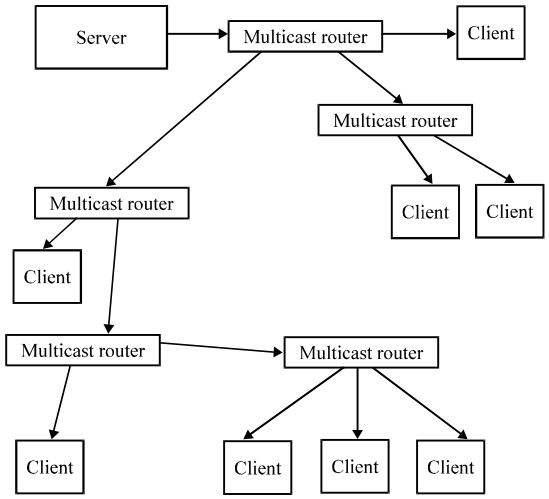

Fig. 1: Overall architecture

batch DSA signature, Merkle tree implementation and packet authentication analysis for efficiency data transformation in the wireless and internet.

Multicast server: Multicast server is used to establish a one-to-many connection to each device in a Virtual LAN (VLAN) or a broadcast domain for each VLAN segment. The multicast server forwards incoming broadcasts only to the multicast address that map to the broadcast address. Server sends the message in secure way by the digital signature method.

Multicast router: Often referred to as an mrouter, a multicast router that is configured to recognize signalling that is received in either multicast or unicast packets. Based on the type of packet identified, multicast routers then determine the routing or distribution of the data that is needed to forward the multicast or unicast packets to their intended destination. By employing a series of algorithms as part of the identification process, mrouters quickly initiate sending orders to the appropriate switches within the network and execute the delivery of the data packet. Routers manage network traffic by maintaining routing tables. These routing tables contain information that specifies which networks and hosts can be reached by which routes. A routing table entry can be either static or dynamic. For one machine to be able to find another over a network there must be a mechanism in place to describe how to get from one to the other. This is called routing. A route is a defined pair of addresses: A destination and a gateway. The pair indicates that if you are trying to get to this destination, communicate through this gateway.

MATERIALS AND METHODS

Basic scheme: The target is to authenticate multicast streams from a sender to multiple receivers. Generally, the sender is a powerful multicast server managed by a central authority and can be trustful. The sender signs each packet with a signature and transmits it to multiple receivers through a multicast routing protocol. Each receiver is a less powerful device with resource constraints and may be managed by a nontrustworthy person. Each receiver needs to assure that the received packets are really from the sender (authenticity) and the sender cannot deny the signing operation (nonrepudiation) by verifying the corresponding signatures.

Batch DSA signature: DSA is popular digital signature algorithm. Unlike RSA which is based on the hardness of factoring two large primes, DSA is deemed secure based on the difficulty of solving DLP. A batch DSA signature scheme was proposed in Lysyanskaya *et al.* (2004) but later was found insecure. Harn improved the security of in Desmedt *et al.* (1992) and Perrig *et al.* (2001). Unfortunately, Boyd and Pavlovski pointed out in Perrig (2001) that Harns research is still vulnerable to malicious attacks. Here, researchers propose a batch DSA scheme based on Harns research and counteract the attack.

Harn DSA: A batch DSA scheme based on Harns research and counteract the attack. p, a prime longer than 512 bits.

q = A 160-bit prime divisor of p-1
g = A generator of z_p^* with order q, i.e., $g^q = 1 \bmod p$
x = The private key of the signer, $0<x<q$
y = The public key of the signer, $y = g^x \bmod p$
h = A hash function generating an output in Z_p^*

Given a message m, the signer generates a signature by: Randomly selecting an integer k with $0<k<q$, computing h = h (p), computing $r = (g^k \bmod p) \bmod q$ and computing $s = rk-hx \bmod q$. The signature for m is (s, r). The receiver can verify the signature by first computing h (m) and then checking whether:

$$\left(\left(g^{sr^{-1}} y^{hr^{-1}}\right) \bmod p\right) \bmod q = r$$

This is because if the packet is authentic, then:

$$\left(\left(g^{sr^{-1}} y^{hr^{-1}}\right) \bmod p\right) \bmod q = \left(\left(g^{(s+hx)r^{-1}}\right) \bmod p\right) \bmod q$$
$$= \left(g^k \bmod p\right) \bmod q = r$$

The batch DSA: In order to counteract the Boyd-Pavlovski attack, the batch DSA makes an improvement to the Harn DSA algorithm. Researchers replace the hash operation hgms in the signature generation and verification process with hgr, ms. All the other steps are the same as those in Harn's scheme.

Enhanced scheme: An enhanced scheme called MABS-E which combines the basic scheme MABS-B and a packet filtering mechanism to tolerate packet injection. In particular, the sender attaches each packet with a mark which is unique to the packet and cannot be spoofed. At each receiver, the multicast stream is classified into disjoint sets based on marks. Each set of packets comes from either the real sender or the attacker. The mark design ensures that a packet from the real sender never falls into any set of packets from the attacker and vice versa. Next, each receiver only needs to perform batch verify () over each set. If the result is true, the set of packets is authentic. If not, the set of packets is from the attacker and the receiver simply drops them and does not need to divide the set into smaller subsets for further batch verification. Therefore, a strong resilience to DoS due to injected packets can be provided.

Figure 2 shows example of Merkle tree. The sender constructs a binary tree for eight packets. Each leaf is a hash of one packet. Each internal node is the hash value on the concatenation of its left and right children. For each packet, a mark is constructed as the set of the siblings of the nodes along the path from the packet to the root. For example, the mark of the packet P3 is fH_4; $H_{1,2}$; $H_{5,8}$ and the root can be recovered, as $H_{1,8} = H((H_{1,2} (H(P_3), H_4)), H_{5,8})$, constructing a Merkle tree is very efficient because only hash operations are performed. Meanwhile, the one-way property of hash operation ensures that given the root of a Merkle tree, it is infeasible to find a packet which is not in the set associated with the Merkle tree and from which there is a path to the root. This guarantees that

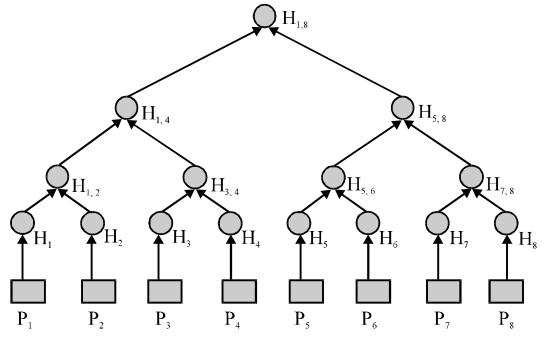

Fig. 2: Merkle Tree

forged packets cannot fall into the set of authentic packets. The hash algorithm can be SHA or MD5 algorithm.

Performance analysis: Researchers consider latency, computation and communication overhead for efficiency evaluation under lossy channels and DoS channels. Researchers compare the computation overhead of three batch signature schemes RSA, BLS and DSA. RSA and BLS require one modular exponentiation at the sender and DSA requires two modular multiplications when r value is computed offline. Usually one c-bit modular exponentiation is equivalent to 1: 5 c modular multiplications over the same field. Moreover, a c-bit modular exponentiation in DLP is equivalent to a c/6-bit modular exponentiation in BLS for the same security level. Therefore, researchers can estimate that the computation overhead of one 1.024-bit RSA signing operation is roughly equivalent to that of 768 DSA signing operations (1,536 modular multiplications) and that of 6 BLS signing operations (each one is corresponding to 255 modular multiplications).

According to the report on the computational overhead of signature schemes on P3 1 GHz CPU, the signing and verification time for 1,024-bit RSA with a 1,007-bit private key are 7.9 and 0.4 msec, for 157-bit BLS are 2.75 and 81 msec and for 1.024-bit DSA with a 160-bit private key (without precomputing r value) are 4.09 and 4.87 m sec. Researchers can observe that for BLS and DSA the signing is efficient but the verification is expensive and vice versa for RSA. Therefore, researchers can save more computation resource at the sender by using the batch BLS and batch DSA than batch RSA. It is also, meaningful to use the batch BLS and batch DSA at the receiver to save computation resources.

Researchers also compare the length of 2 popular hash algorithm MD5 and SHA-1 and the signature length of 3 signature algorithms in Table 1. Given the same security level as 1,024-bit RSA, BLS generates a 171-bit signature and DSA generates a 320-bit signature. It is clear that by using BLS or DSA, MABS can achieve more bandwidth efficiency than using RSA and could be even more efficient than conventional schemes using a large number of hashes.

Table 1: Computational overhead of different batch schemes

Schemes	Sender (per packet)	Receiver (per n packets)
Batch RSA	1 E	1 E+(2n-2) M
Batch BLS	1 E	2 P+(2n-2) M
Batch DSA	2 M	2 E+3n M

E = Modulor exponentiation; M = Modulor multiplaction; p = Pairing

CONCLUSION

To reduce the signature verification overheads in the secure multimedia multicasting, block-based authentication schemes have been proposed. Unfortunately, most previous schemes have many problems such as vulnerability to packet loss and lack of resilience to Denial of Service (DoS) attack. To overcome these problems, researchers develop a novel authentication scheme MABS.

Researchers have demonstrated that MABS is perfectly resilient to packet loss due to the elimination of the correlation among packets and can effectively deal with DoS attack. Moreover, researchers also show that the use of batch signature can achieve the efficiency less than or comparable with the conventional schemes. Finally, researchers further develop two new batch signature schemes based on BLS and DSA which are more efficient than the batch RSA signature scheme.

REFERENCES

Desmedt, Y., Y. Frankel and M. Yung, 1992. Multi-receiver/multi-sender network security: Efficient authenticated multicast/feedback. Proceedings of the 11th Annual Joint Conference of the IEEE Computer and Communications Societies, May 4-8, 1992, Florence, Italy, pp: 2045-2054.

Even, S., O. Goldreich and S. Micali, 1996. On-line/offline digital signatures. J. Cryptol., 9: 35-67.

Jeong, J., Y. Park and Y. Cho, 2005. Efficient DoS resistant multicast authentication schemes. Proceedings of the International Conference on Computational Science and its Applications, May 9-12, 2005, Singapore, pp: 353-362.

Karlof, C., N. Sastry, Y. Li, A. Perrig and J.D. Tygar, 2004. Distillation codes and applications to DoS resistant multicast authentication. Proceedings of the 11th Annual Network and Distributed System Security Symposium, February, 5-6, 2004, San Diego, CA., USA.

Lysyanskaya, A., R. Tamassia and N. Triandopoulos, 2004. Multicast authentication in fully adversarial networks. Proceedings of the IEEE Symposium on Security and Privacy, May 9-12, 2004, Berkeley, CA., USA., pp: 241-253.

Pannetrat, A. and R. Molva, 2003. Efficient multicast packet authentication. Proceedings of the 10th Annual Network and Distributed System Security Symposium, February 6-7, 2003, San Diego, CA., USA.

Park, J.M., E.K. Chong and H.J. Siegel, 2002. Efficient multicast packet authentication using signature amortization. Proceedings of the IEEE Symposium on Security and Privacy, May 12-15, 2002, Berkeley, CA., USA., pp: 227-240.

Perrig, A., R. Canetti, J.D. Tygar and D. Song, 2000. Efficient authentication and signing of multicast streams over lossy channels. Proceedings of the IEEE Symposium on Security and Privacy, May 14-17, 2000, Berkeley, CA., USA., pp: 56-73.

Perrig, A., 2001. The BiBa one-time signature and broadcast authentication protocol. Proceedings of the 8th ACM Conference on Computer and Communications Security, November 5-8, 2001, Philadelphia, PA., USA., pp: 28-37.

Perrig, A., R. Canetti, D. Song and J.D. Tygar, 2001. Efficient and secure source authentication for multicast. Proceedings of the Internet Society Network and Distributed System Security Symposium, February 23-26, 2001, San Diego, CA., USA., pp: 35-46.

Rohatgi, P., 1999. A compact and fast hybrid signature scheme for multicast packet authentication. Proceedings of the 6th ACM Conference on Computer and Communications Security, November 1-4, 1999, Singapore, pp: 93-100-10.1145/319709.319722.

Wong, C.K. and S.S. Lam, 1998. Digital signatures for flows and multicasts. Proceedings of the 6th International Conference on Network Protocols, October 13-16, 1998, Austin, TX., USA., pp: 198-209.

Wong, C.K. and S.S. Lam, 1999. Digital signatures for flows and multicasts. IEEE/ACM Trans. Networking, 7: 502-513.

Wu, Y. and T. Li, 2006. Video stream authentication in lossy networks. Proceedings of the IEEE Wireless Communications and Networking Conference, Volume 4, April 3-6, 2006, Las Vegas, NV., USA., pp: 2150-2155.