The BiBa One-Time Signature and Broadcast Authentication Protocol-Summary

Vishu Gupta (vgupta6)
vgupta6@uiuc.edu

April 18, 2002

1 Summary

The BiBa broadcast authentication protocol is based on the BiBa one-time signature. The BiBa one-time signature is a new signature construction that uses one-way functions without trapdoors. BiBa features a low verification overhead and a relatively small signature size. In comparison to other one-time signature schemes, BiBa has smaller signatures and is at least twice as fast to verify (which probably makes it one of the fastest signature schemes to date for verification).

According to the authors the main challenges faced in the design and broadcast protocol are efficient generation and verification of the information used for authentication, real-time authentication, individual message authentication, robustness to packet loss, scalability and small size of authentication information.

SEALs are entities precomputed by the sender and used for authentication. For a seal s, the public key $f_s = F_s(0)$ is told to the receivers which they for verifying the seal. Each message is attched with two seals falling to the same hash bin when hashed with the hash function corresponding to the message. Attacker can know all the seals over time.

Two things:

1. reducing probability of attacker to forge signature when he knows a certain number of seals. Multiple-two way hashes and multi-way collision approaches or multi-round schemes.

2. allowing attacker to know minimum number of active seals.

For the second issue, they propose the **one-way seal chains**. The concept of salt-chains used to generate seal chains to ensure that the seals are reasonably small while still secure because of big salt.

Two extensions:

1. SEAL boundary around the disclosed seals and the adjacent seals are used for signing. Time synchronization is important here. This does not tolerate packet losses.

2. We include the boundary information in the message itself but then it has high communication overhead.

Fully parallelized BiBa implementation requires two sequential hash function computations for signing.

2 PROS

- Its a novel scheme (different from RSA).
- Biba has smaller signatures.
- At least twice as fast to verify.
- Although the signature generation overhead is higher than previous schemes, it can be trivially parallelized.
- Enjoys exponentially increasing security such that it is secure if the signer only has modest computation resources.
- It is hard for an adversary to forge a signature (if implemented correctly).
- Lightweight verification.
- Achieves high level of security.
- The BiBa authentication protocol satisfies 5 of the 6 desired properties of an efficient broadcast authentication protocol (except that the sender overhead to generate the authenticationn information is in general higher than for previous approaches; BUT can be parallelized).
- The construction of the broadcast authentication protocol presented is GENERAL and can also be applied to other signature schemes based on one-way functions without trapdoors.
- The protocol is scalable.
- The algorithm seems reasonably straightforward to understand (and implement) and seems as though it should correctly produce the results that the authors intend (although the particular protocol used in this context needs to be subjected to peer review, of course).
- Finding the probability of an adversary compromising the protocol or the performance of the protocol (measured in expected number of hashes) is easily done using math found in undergraduate probability courses.
- As described in Section 4, the protocol can be extended to favor computation/communication overhead vs. packet loss. This might be used as a crude way of "tuning" the algorithm for a specific use.
- The protocol could be directly applied to sending authenticated broadcasts to a large group of embedded systems (e.g., an ad-hoc network of sensors).

3 CONS

- The BiBa public key is large and the signature generation overhead is higher than previous schemes based on one-way functions without trapdoors.
- It requires a moderate computation overhead when the sender generates authenticaion information.
- It requires time synchronization, which can be annoying.
- A lot was written about how much information is revealed for each packet. It would have been nice to have seen an analysis how often keys need to be renewed. For example, a high quality, two hour streaming movie would almost definitely reveal a ton of information about the key.

4 Poll

- Strong Reject : 0
- Weak Reject : 0
- Weak Accept : 5
- Strong Accept : 9