Critique on The TESLA Broadcast Authentication Protocol

Summary:

TESLA (Time Efficient Stream Loss-tolerant Authentication) broadcast authentication protocol is an efficient protocol with low communication and computation overhead, based on loose time synchronization between the sender and the receivers. The main assumptions of TESLA are as follow:

1. The receivers are loosely time synchronized with the sender.
2. Assume that a mechanism allows a receiver to learn the authenticated public key K_s
3. The receiver's clock is time synchronized up to a maximum error of Δ.
4. The one way hash functions F, $F^{'}$ are secure PRFs, and the function F furthermore provides weak collision resistance.

As long as these are satisfied, we can defense the threat module like:

1. Attackers with secret key forges data and impersonate the sender, such as forging a TESLA packet that the receivers will authenticate successfully.
2. Denial of service for digital signature. (verification for it is often computationally expensive)

By using time to achieve asymmetric properties and MAC as symmetric cryptographic primitive, TESLA can obtain both integrity and authenticity, also gives an efficient way to send packet to multiple receivers.

Strengths:

1. As the name of the protocol, it can tolerant packet loss by using one-way chains, which can compute the previous interval keys with one key.
2. With low computation and communication overhead, but also have the advantage of asymmetric-key properties.
3. The attacker cannot forge a TESLA packet or impersonate the sender even if he/she captures the packet, due to the time limit between sender and the receivers.

Weaknesses:

1. Since the receivers have to store the valid packets before the sender to disclose the keys, DoS attack may do it trick when the attacker captures packets from the sender, gets the key and forges fake data and fake MAC to the receivers. As the packet are similar, the receivers have to store lots of fake data, which may lead to this kind of DoS attack success.
2. Message exchange cannot be too frequently, since the waiting time for key disclosure will accumulate.

Reflection:

1. TESLA can tolerant packet lost due to the design of the key-chain, but it doesn't mean that the content of the pocket can be recovered. If it can tolerant packet lost, how to retrieve the lost pocket should be a problem.
2. I would like to learn more about the length of the chain, which determines the computation time and the storage requirement. If possible, the paper could provide more data information for it.

Reference: A. Perrig, R. Canetti, J. D. Tygar, and D. Song, "The TESLA broadcast authentication protocol," in RSA CryptoBytes, 2005.