

**NC STATE UNIVERSITY** Computer Science

# CSC 774 Network Security

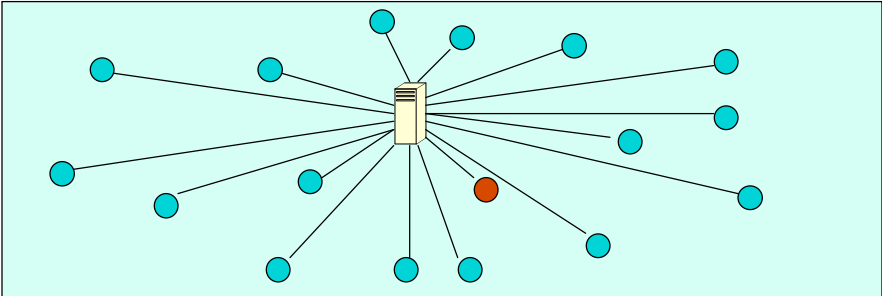
---

## Topic 6. Broadcast Authentication

CSC 774 Network Security Dr. Peng Ning 1

## What Is Broadcast Authentication?


- One sender; multiple receivers
  - All receivers need to authenticate messages from the sender.



**NC STATE UNIVERSITY** Computer Science CSC 774 Network Security Dr. Peng Ning 2

## Challenges in Broadcast Authentication

- Can we use symmetric cryptography in the same way as in point-to-point authentication?
- How about public key cryptography?
  - Effectiveness?
  - Cost?
- Research in broadcast authentication
  - Reduce the number of public key cryptographic operations



NC STATE UNIVERSITY Computer Science

# CSC 774 Network Security

---

## Topic 6.1 TESLA and EMSS

CSC 774 Network Security Dr. Peng Ning 4

## Outline

- Two Schemes
  - TESLA
    - Sender Authentication
    - Strong loss robustness
    - High Scalability
    - Minimal overhead
  - EMSS
    - Non-Repudiation
    - High loss robustness
    - Low overhead

## TESLA - Properties

- Low computational overhead
- Low per packet communication overhead
- Arbitrary packet loss tolerated
- Unidirectional data flow
- No sender side buffering
- High guarantee of authentication
- Freshness of data

## TESLA – Overview

- Timed Efficient Stream Loss-tolerant Authentication
- Based on timed and delayed release of keys by the sender
- Sender commits to a random key  $k$  and transmits it to the receivers without revealing it
- Sender attaches a MAC to the next packet  $P_i$  with  $k$  as the MAC key
- In  $P_{i+1}$  packet sender “decommits” to the key and receiver uses this key  $k$  to verify  $P_i$
- Need a security assurance

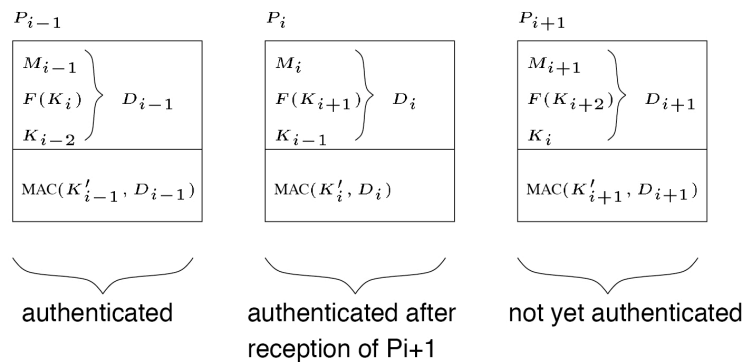
## TESLA – Scheme I

- $P_{i-1} = \langle D_{i-1}, \text{MAC}(K'_{i-1}, D_{i-1}) \rangle$
- $D_{i-1} = \langle M_{i-1}, F(K_i), K_{i-2} \rangle$
- $M_{i-1}$  = Message
- $F(K_i)$  = commitment to  $K_i$
- $K'_i = F'(K_i)$
- Each packet  $P_{i+1}$  authenticates  $P_i$
- Problems ??

## TESLA – Scheme I (contd )

- If attacker gets  $P_{i+1}$  before receiver gets  $P_i$ , it can forge  $P_i$
- Security Condition
  - $\text{Arr}T_i + \delta_t < T_{i+1}$
  - Sender's clock is no more than  $\delta_t$  seconds ahead of that of the receivers
- Packet loss not tolerated

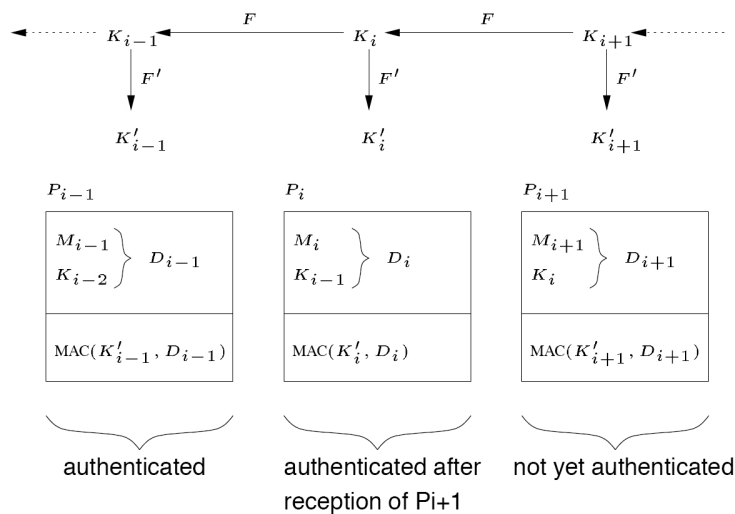
## TESLA – Scheme I (Contd)



## TESLA – Scheme II

- Generate a seq of keys  $\{ K_i \}$
- $F^v(x) = F^{v-1}( F(x) )$
- $F^0(x) = x$
- $K_0 = F^n(K_n)$
- $K_i = F^{n-i}(K_n)$
- Attacker cannot invert  $F$  or compute any  $K_j$  given  $K_i$ ;  $j > i$
- Receiver can compute all  $K_j$  from  $K_i$ ;  $j < i$
- $K_j = F^{i-j}(K_i)$ ;  $K'_i = F'(K_i)$

## TESLA – Scheme II (Contd)



## TESLA – Scheme III

- Remaining problems with Scheme II
  - Inefficient for fast packet rates
  - Sender cannot send  $P_{i+1}$  until all receivers receive  $P_i$
- Scheme III
  - Does not require that sender waits for receiver to get  $P_i$  before it sends  $P_{i+1}$
  - Basic idea: Disclose  $K_i$  in  $P_{i+d}$  instead of  $P_{i+1}$

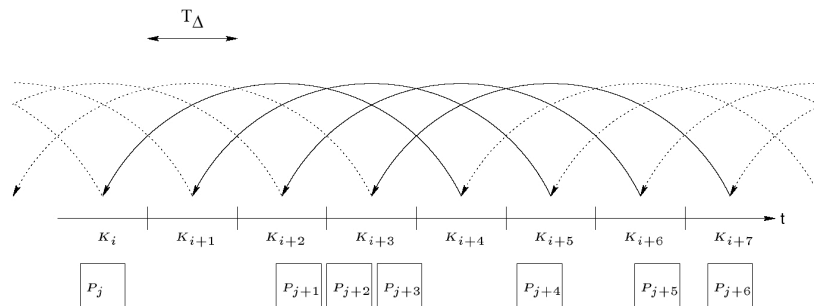
## TESLA – Scheme III (Cont'd)

- Disclosure delay  $d = \lceil (\delta_{tMax} + d_{NMax})r \rceil$ 
  - $\delta_{tMax}$ : maximum clock discrepancy
  - $d_{NMax}$ : maximum network delay
  - $r$ : packet rate
- Security Condition:
  - $ArrT_i + \delta_t < T_{i+d}$
- Does choosing a large  $d$  affect the security?

## TESLA – Scheme IV

- Deals with Dynamic transmission rates
- Divide time into intervals
- Use the same  $K_i$  to compute the MAC of all packets in the same interval  $i$
- All packets in the same interval disclose the key  $K_{i-d}$
- Achieve key disclosure based on interval basis than on packet index basis

## TESLA – Scheme IV (Cont'd)





## TESLA – Scheme IV (Cont'd)

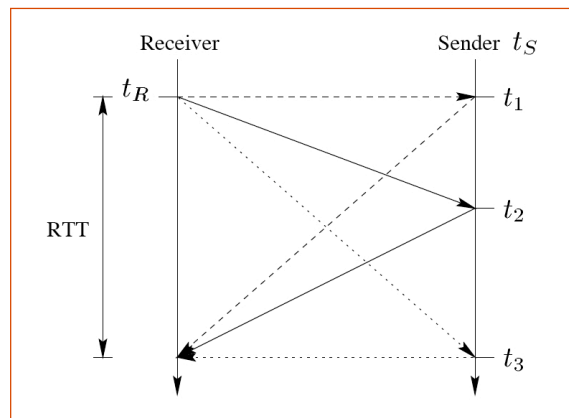
- Interval index:  $i = (t - T_0)/T_\Delta$
- $K_i' = F'(K_i)$  for each packet in interval  $i$
- $P_j = \langle M_j, i, K_{i-d}, \text{MAC}(K_i', M_j) \rangle$
- Security condition:
  - $i + d > i'$
  - $i' = (t_j + \delta_t - T_0)/T_\Delta$ 
    - $i'$  is the farthest interval the sender can be in

## TESLA – Scheme V

- In Scheme IV:
  - A small  $d$  will force remote users to drop packets
  - A large  $d$  will cause unacceptable delay for fast receivers
- Scheme V
  - Use multiple authentication chains with different values of  $d$
- Receiver verifies one security condition for each chain  $C_i$ , and drops the packet if none is satisfied

## TESLA – Initial Time Synchronization

- R→S: Nonce
- S →R: {Sender Time  $t_S$ , Nonce, ...} $K_S^{-1}$



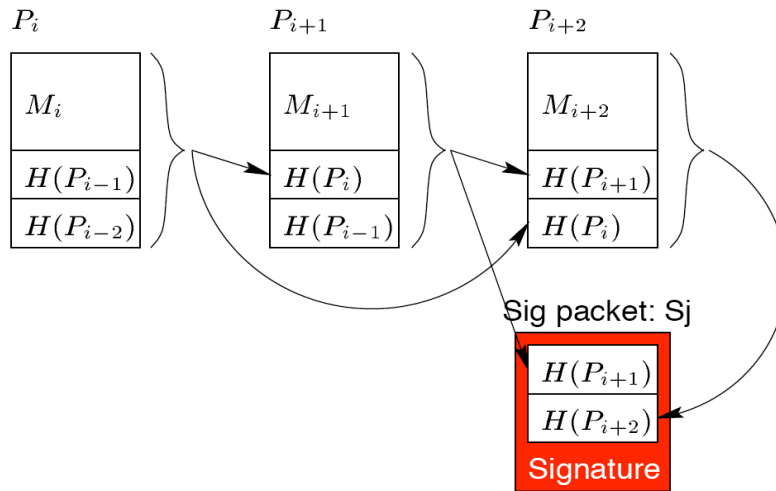
R only cares the maximum time value at S.

Max clock discrepancy:  
 $\Delta_T = t_S - t_R$

## EMSS

- Efficient Multichained Streamed Signature
- Useful where
  - Non Repudiation required
  - Time synchronization may be a problem
- Based on signing a small no. of special packets in the stream
- Each packet linked to a signed packet via multiple hash chains

## EMSS – Basic Signature Scheme



## EMSS – Basic Signature Scheme (Cont'd)

- Sender sends periodic signature packets
- $P_i$  is verifiable if there exists a path from  $P_i$  to any signature packet  $S_j$

## EMSS – Extended Scheme

- Basic scheme has too much redundancy
- Split hash into  $k$  chunks, where any  $k'$  chunks are sufficient to allow the receivers to validate the information
  - Rabins Information Dispersal Algorithm
  - Some upper few bits of hash
- Requires any  $k'$  out of  $k$  packets to arrive
- More robust