What Is Broadcast Authentication?

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

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
Advanced Network Security

Topic 3.1 TESLA and EMSS

Outline

• Two schemes
  – TESLA
    • Sender authentication
    • Strong loss robustness
    • High scalability
    • Minimal overhead
  – EMSS
    • Non-repudiation
    • High loss robustness
    • Low overhead

TESLA – Overview

• Timed Efficient Stream Loss–tolerant Authentication
• Based on timed and delayed release of keys by the sender
• High level ideas
  – Sender commits to a random key $K$ and transmits the commitment to the receivers without revealing it
  – Sender attaches a MAC to the next packet $P_i$ with $K$ as the MAC key
  – Sender releases the key in packet $P_{i+1}$ and receiver uses this key $K$ to verify $P_i$
TESLA – Scheme I

• Each packet $P_{i+1}$ authenticates $P_i$
• Problems?
     – Security? Robustness?

TESLA – Scheme I (Cont’d )

• If attacker gets $P_{i+1}$ before receiver gets $P_i$, it can forge $P_i$
• Security Condition
     – $ArrT_i + \delta < T_{i+1}$
     – Sender’s clock is no more than $\delta$ seconds ahead of that of the receivers
     – One simple way: constant data rate
• Packet loss not tolerated

TESLA – Scheme II

• Generate a sequence of keys $\{K_i\}$ with one-way function $F$
     – Randomly generate $K_0$
     – $K_0 = F^n(K_0)$
     – Commitment $K_0 = F^i(K_i)$
• Attacker cannot invert $F$ or compute any $K_j$ given $K_i$ where $j > i$
• Receiver can compute all $K_j$ from $K_i$, where $j < i$
     – $K_j = F^j(K_i)$; $K'_i = F^i(K_j)$
TESLA – Scheme II (Cont’d)

- 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 wait for receiver to get
    $P_i$ before it sends
  - Basic idea: Disclose $K_i$ in $P_{i+d}$ instead of $P_{i+1}$

TESLA – Scheme III (Cont’d)

- Disclosure delay $d = \left\lfloor (\delta_{\text{Max}} + d_{\text{Max}}) r \right\rfloor$
  - $\delta_{\text{Max}}$: maximum clock discrepancy
  - $d_{\text{Max}}$: maximum network delay
  - $r$: packet rate
- Security Condition:
  - $ArrT_i + \delta_i < T_{i+d}$
- Question:
  - Does choosing a large $d$ affect the security?
TESLA – Scheme IV

• Deal with dynamic transmission rates
• Idea
  – 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 intervals rather than on packet indexes

TESLA – Scheme IV (Cont’d)

• Interval index: $i = (t - T_p) / T_{\Delta}$
• $K_i' = F'(K_i)$ for each packet in interval $i$
• $P_j = \langle M_j, i, K_{i,d}, MAC(K_i', M_j, i, K_{i,d}) \rangle$
• Security condition:
  – $i + d < i'$
  – $i' = (t_j + \delta_t - T_p) / T_{\Delta}$
  – $i'$ 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–Immediate Authentication

- $M_{j+vd}$ can be immediately authenticated once packet $j$ is authenticated
- Not to be confused with packet $j+vd$ being authenticated

TESLA – Initial Time Synchronization

- $R \rightarrow S$: Nonce
- $S \rightarrow R$: (Sender Time $t_s$, Nonce, ...)$K_s^{-1}$

$R$ only cares about 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 number of special packets in the stream
- Each packet linked to a signed packet via multiple hash chains

---

EMSS – Basic Signature Scheme

- 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
  – Rabin’s Information Dispersal Algorithm
  – Some upper few bits of hash
• Requires any \( k' \) out of \( k \) packets to arrive
• More robust