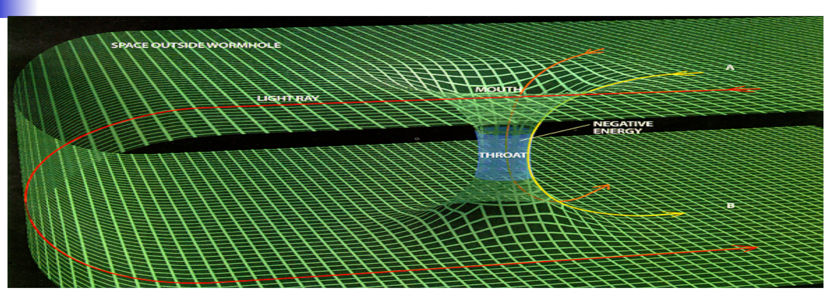
Wormhole Attacks in Wireless Networks

Yiu-Chun Hu, Adrian Perrig and David Johnson

CS818 Presentation By Venkatesh Ramanathan

Introduction



- Wormhole – 'Shortcut' through space and time (Source: wikipedia)
- Origin – Worm burrows through the center of apple instead of traveling the whole distance to get to other side

Introduction

- Wormhole attack – Record packet/bits at one location and tunnel to another location.
- Packet Leashes – To detect wormhole attacks
  - Geographic Leashes and Temporal Leashes
  - Authentication Protocol, TIK, for temporal leashes
- Topology based detection unable to detect wormhole

Introduction

- Tunneled packets arrive with better metric
  - Use wired link, Long range wireless link
- Attacker Can
  - forward each bit instead of waiting for the whole packet.
  - Can create wormhole for packets not addressed to self.
  - Can be performed even when communication has confidentiality/authenticity (no crypto keys required)
  - Invisible at higher layers

Introduction

- Dangerous against ad hoc network routing protocols (DSR, AODV)
  - Tunnel RREQ directly to destination node
  - Destination re-broadcasts copy of RREQ and discard all other RREQ
  - Prevents discovery of routes other than through wormhole
  - Attacker could then drop all data packets (DoS)

Introduction

- OLSR and TBRPF (neighbor discovery protocols)
  - Colluding attackers near nodes A & B wormhole HELLO packets. A & B would believe they are neighbors.
- DSDV
  - If route advertisement is tunneled and A & B not within wireless range, would unable to communicate

Scope

- TIK supports unidirectional and bidirectional wireless links
- Did not consider attacks at physical layer, DoS attacks at MAC layer
- Adversary can place nodes anywhere in the network. Communication between malicious nodes unobservable.
- Using symmetric cryptography as nodes may be resource constrained.
- TIK protocol uses symmetric key cryptography.

Detecting Wormhole Attacks

- Packet Leash – to detect and defend wormhole attacks
- Leash
  - Information added to packet to restrict packet's maximum allowed distance.
  - Designed to protect against wormhole attacks over single hop. Transmission over multiple hops require fresh leash.
- Types:
  - Geographic Leash – Ensure recipient within some distance.
  - Temporal Leash – Upper bound on packet lifetime.

Geographic Leash

- Each node must know its location
- Nodes have loose time synchronization
- $d_{sr} <= || p_s - p_r|| + 2 v (t_r - t_s + \Delta) + \delta$
  - d_{sr} – upper bound on the distance between sender and reciever
  - p_s, p_r – localtions
  - v – maximum velocity of node
  - Δ – Time synchronization error
  - δ – maximum error in location
- Geographic leash can be used to catch an attacker if pretending to be in more than 1 location. (node velocity > maximum node velocity)

Temporal Leash

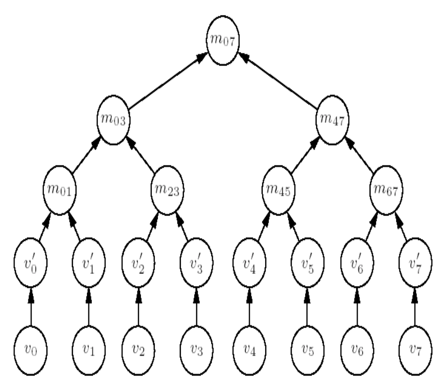
- Nodes must have 'tightly synchronized clocks':
  - maximum difference delta
  - Delta known to all nodes
  - Order of microseconds or hundreds of nano seconds
- Supported hardware
  - LORAN-C – Long Range Navigation Aids
  - WWVB – (NIST time signal) – Used by radio controlled clocks throughout North America
  - GPS, Atomic clocks
- Sender includes time, ts. Receiver computes ts x speed of light and compares with tr. Alternatively, sender includes packet expiration time.
- Digital signature or other authentication scheme to verify timestamps.

Temporal Leashes and TIK

- Sender sets packet expiration time
  - tc = ts + L/c – Δ
  - ts – local time of sender
  - c – speed of signal
  - Delta – time synchronization error
- Receiver checks tr < tc
- Assumes no delay in sending/receiving packets

Merkle Hash Tree – Mechanism for authenticating keys in TIK

- **Values v_o, .. v_{w-1} are placed at leaf nodes**
- **Compute $v_i' = H(v_i)$**
- **Internal node m_{01} = H (v_o'||v_1)**
- **Root value (m_{07}) used to authenticate all leaf values.**
- **To authenticate v_2, sender discloses v_3', m_{01}, and m_{47}**
- **Receiver computes:**
- **H[H[m_{01}||H[H[v_2] || v_3']] || m_{47}]**



Hash Tree Optimization

- Depth of the tree could be quite large (Not practical for storage)
  - $\log_2 [t/I]$; I – interval, t-time between rekeying
  - Solution: Store upper layers and compute lower layers on demand.
  - Reconstructing tree requires 2^{d-1} PRF and $2^d - 1$ application of hash functions.

Hash Tree Optimization

- Number of operations:
  - 2^{D-1} PRF + $2^D - 1$ Hash (D – depth of the tree)
- To choose, d, depth of the tree for on-demand, minimize total storage:

$$\frac{\partial}{\partial d}(2^{D-d+1} - 1 + 2^{d+1} - 2) = 0$$
$$(-\ln 2)2^{D-d+1} + (\ln 2)2^{d+1} = 0$$
$$2^{d+1} = 2^{D-d+1}$$
$$d + 1 = D - d + 1.$$

  - d* = D/2
  - Storage: $2^{\lceil D/2 \rceil + 1} + 2^{\lfloor D/2 \rfloor + 1} - 3.$
  - Tree depth of 34 requires 2.5MB to store

TIK (TESLA with Instant Key Disclosure) Protocol

- Packet Transmission Time >> Time Synchronization Error
- Receiver verifies TESLA security condition (corresponding key has not yet been disclosed) as it receives the packet allowing sender to disclose the key in the same packet.
- TIK implements temporal leash
- TIK requires time synchronization between nodes

TIK

- **Sender Setup**

$\mathcal{F}$: pseudo-random function
$\mathcal{X}$: secret master key
I : expire interval

- Sender uses PRF and master key to derive series of keys Ko, …Kw
- Computationally infeasible for attacker to find master key even if all keys are known (assuming PRF is secure)
- Without master key, attacker could not derive K_i that sender has not disclosed

TIK

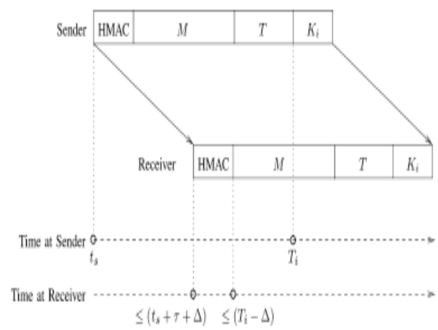
- Sender picks key expiration interval I.
  - Key K_o expires at time T_o, K_1 at T_o+I,..
- Sender constructs merkle hash tree to commit to keys K_0,…K_{w-1}

TIK

- Receiver Bootstrapping
  - Assumes all nodes have synchronized clocks with max error Δ.
  - Receiver knows every senders hash tree root, T_o (key expiration time) and I

TIK – Sending and verifying authenticated packets

- **Senders estimates upper bound on the arrival time of HMAC**
- **Sender picks key K_i that will not expire when receiver gets HMAC**
- **Sender attaches HMAC to packet computed using K_i**
- **Sender discloses K_i and tree authentication values.**



TIK – Sending and verifying authenticated packets

- Receiver verifies that K_i was used to compute authentication.
  - Packet originated from claimed sender.
- TIK eliminates the need for delayed authentication by disclosing key in the same packet.
- Attacker who re-transmits the packet will incur further delay. Receiver thus rejects the packet.

Evaluation

- Computation Power
  - Optimized MD5 hashing (1.3 mill hashes per sec on Pentium III, 222,000 in iPAQ)
- Storage
  - 2.6MB for hash tree storage.
- TIK would need 18% CPU on iPAQ for authentication.
- TIK is not feasible for sensor networks.

Security Analysis

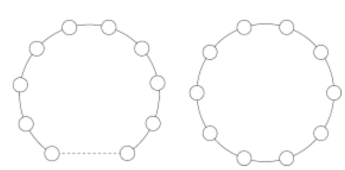
- Packet leashes ensures that attacker is not causing signal to propagate father than specified distance.
- Does not account for the following:
  - Malicious receiver refuse to check the leash
  - Refuse to check authentication
  - Could tunnel packets to another attacker
  - Nodes can claim false time stamp/location.

Geographic Vs Temporal Leashes

- Geographic
  - Can be used with radio propagation model to detect tunnels through obstacles.
  - No tight time synchronization.
  - $d_{sr} \le ||p_s - p_r|| + 2V.(t_r - t_s + \Delta) + \delta$
  - Use when $\delta < c\,\Delta$
- Temporal
  - When used with TIK, less network and computational overhead.
  - $d_{sr} \le c.(t_r - t_s + \Delta)$
  - Use when $\delta >= c\,\Delta$

Related Work

- Topology-Based Approach – Build a model of topology from distance measurements between nodes.



Related Work

- Directional antennas for detecting wormhole attacks using correctly positioned verifier (Hu & Evans).
- Open, Half-Open and Closed worm holes (Wang, et. al.)
  - Open – no higher layer
  - Half-open – one end at higher layer
  - Closed – higher layer
- Radio Frequency Water Marking (authenticates wireless transmission by modulating RF wave form)
- TESLA & TIK
  - TESLA requires looser time synchronization where as TIK better for hop-by-hop authentication (TIK key disclosure along with packet)

Conclusions

- Wormhole attack that exploits routing protocols in ad hoc networks was introduced.
- Presented Packet Leashes (Geographic & Temporal Leashes) to defend against such attacks.
- Presented TIK protocol to authenticate packets received.
  - TIK requires n public keys
  - Node requires 3 – 6 hash function evaluations per interval and 30 evaluations per packet.
  - Less than 3% memory use and 18% CPU use.
  - TIK prevents attacks that cause signal to travel distances longer than radio range

Comments

- Wormhole attack – different form of man in the middle attack
- Geographic Leash – Did not include processing delay, speed of the signal, lower bound on distance
- Temporal leash – TTL
- Network overhead.
- Weak evaluation.
- No experiments.