Packet Leashes: A Defense against Wormhole Attacks in Wireless Networks

IEEE INFOCOM 2003

2003.07.31

林佑青

Outline

- Introduction
- Problem Statement
- Wormhole Attacks
  - Geographical Leashes
  - Temporal Leashes
- TIK Protocol
- Evaluation
- Conclusions

Introduction

- Wormhole attack

  An attacker records packets at one location in the network, tunnels them to another location, and retransmits them into the network.

- Packet leash

  A general mechanism to detect a wormhole attack.

- TIK (TESLA with Instant Key Disclosure)

  An efficient authentication protocol designed for use with temporal leashes.

Problem Statement

- The wormhole attack is particularly dangerous against many ad hoc routing protocols.

- DSR, AODV - use Route Request for route discovery
- DSDV, OLSR, TBRPF - rely on the reception of broadcast packets for neighbor detection
- OLSR and TBRPF -use HELLO packets to detect neighbors

- Any wireless access control system

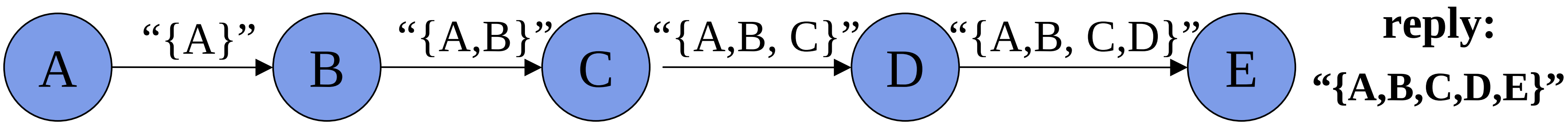
  - an attacker could relay the authentication exchanges to gain unauthorized access

DSR - **D**ynamic **S**ource **R**outing

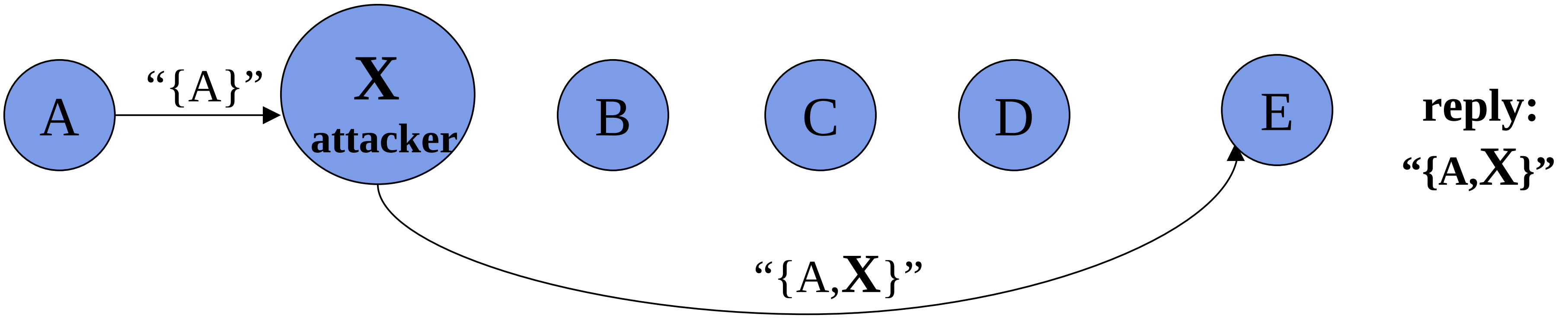
AODV - **A**d Hoc **O**n-Demand **D**istance **V**ector

Route Discovery: 1) flood Route request message through network

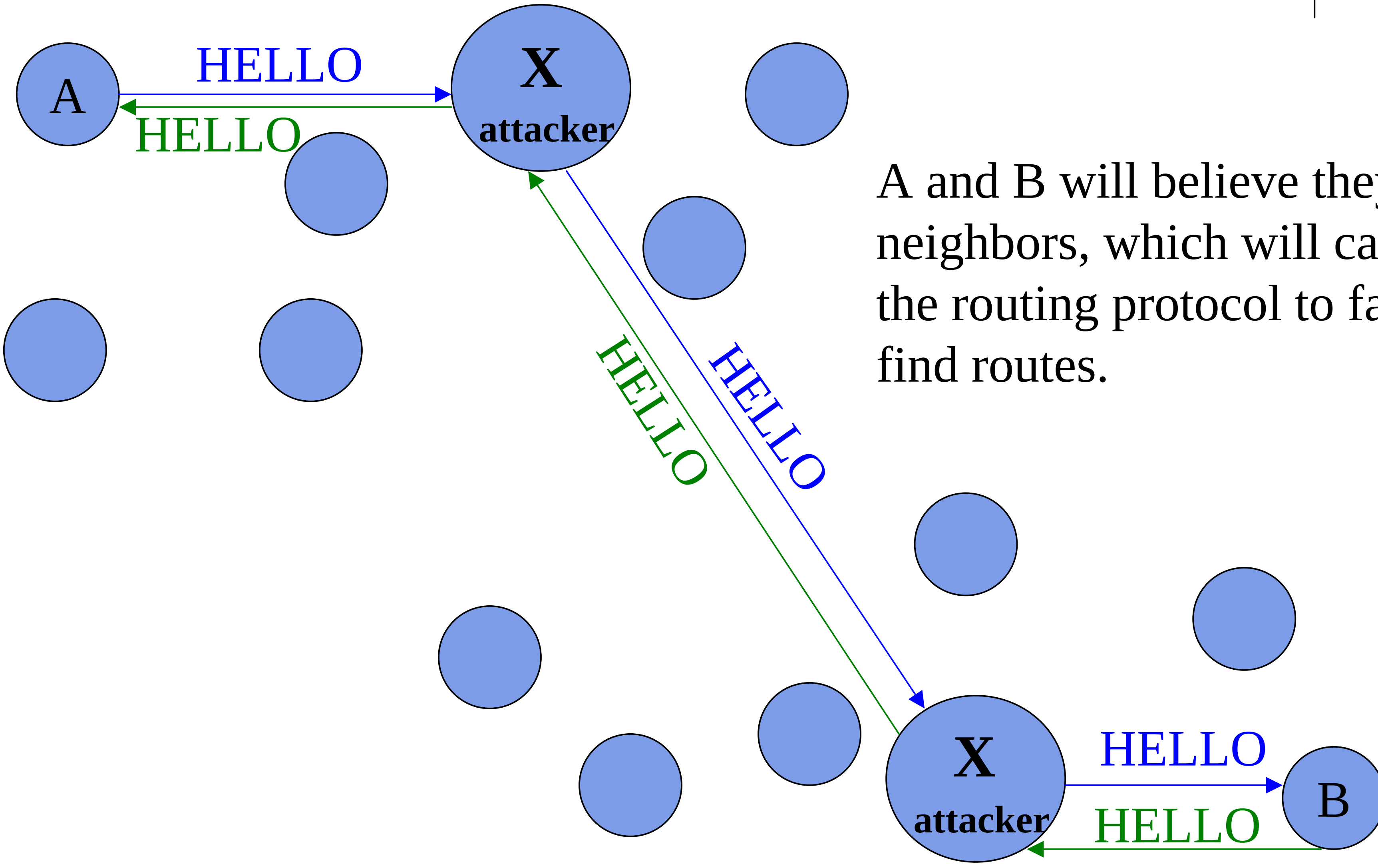
2) request answered with route reply by destination



Wormhole attack:



OLSR and **TBRPF** use HELLO packets to detect neighbors



A and B will believe they are neighbors, which will cause the routing protocol to fail to find routes.

Detecting Wormhole Attacks

- Leash is any information added to a packet designed to restrict the packet's maximum allowed transmission distance

- Geographical leash insures that the recipient of the packet is within a certain distance from the sender.

- Temporal leash ensures that the packet has an upper bound of its lifetime (restricts the maximum travel distance).

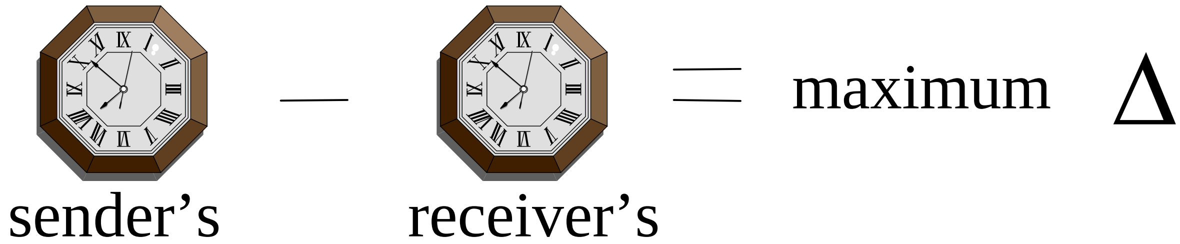
Geographical Leashes

$$dsr \leq \|Ps - Pr\| + 2v^*(tr - ts + \Delta) + \delta$$

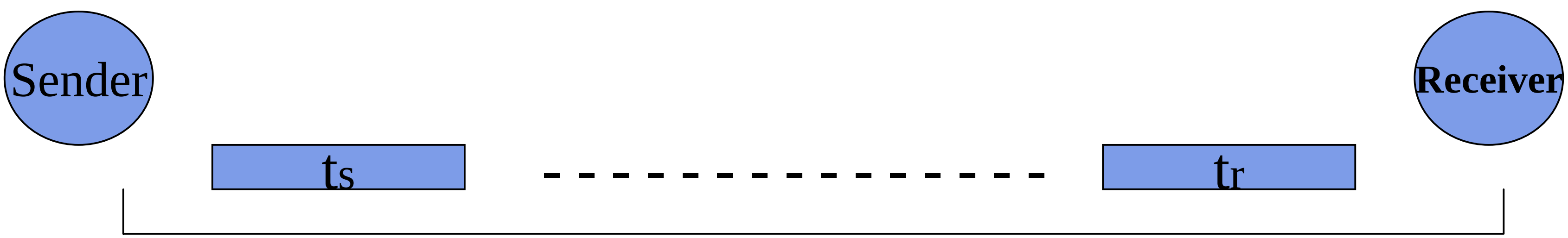
Ps - location of the Sender
Pr - location of the Receiver
ts - time at which Sender sent the packet
tr - time at which Receiver received the packet
v - velocity of any node
δ - maximum relative error in location information
$\pm\Delta$-error in the clocks synchronization

Any authentication technique can be used to allow a receiver to authenticate the location and timestamp in the received packets

Temporal Leashes



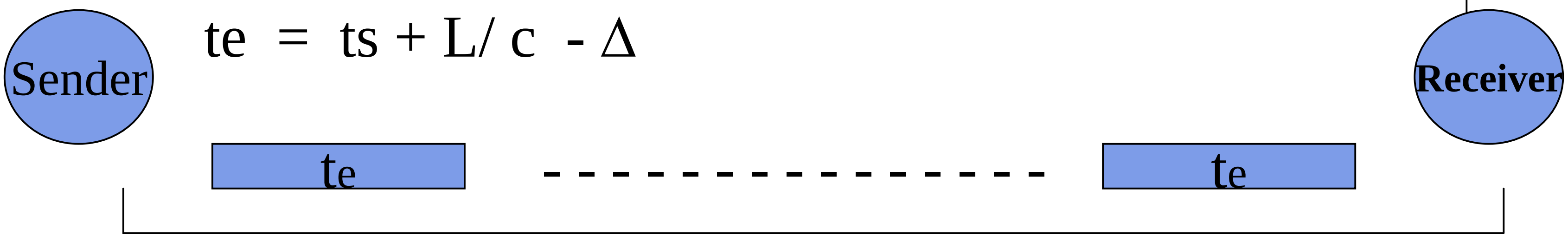
Δ - must be known by all nodes in the network



• te is Expiration time, based on the allowed maximum transmission distance and the speed of light after which the receiver should not accept the packet.

Temporal Leash Construction Details

$$t_e = t_s + L/c - \Delta$$



c - propagation speed of the wireless signal

L - prevents the packet from travelling further than distance L

t_s - time at which Sender sent the packet

t_r - time at which Receiver received the packet

t_e - expiration timer

$\pm\Delta$-error in the clocks synchronization

Receiver needs to authenticate the expiration time:

- Sender and Receiver must share a secret key K
- To send a message M to a receiver R:

$S \rightarrow R$: (M, $HMAC_K$ (M))

Drawbacks in using HMAC in the standard

- n(n-1)/2 keys in network with n nodes
- Key setup is an expensive operation. Impractical in large networks.
- This approach can not efficiently authenticate broadcast packets
- To secure a broadcast packet, add to the packet separate message authentication code -- makes packet extremely large
- Separate HMAC can be avoided by multiple receivers sharing the same key, but it might allow colluding receivers to impersonate the sender

Tree-Authenticated Values

- TIK requires an efficient mechanism for authenticating keys
- Values from a one-way hash chain are very efficient to verify, but only if values in sequence
- For the TIK, values used very sparsely
- One-way hash function is efficient to compute, but computation requires overhead
- Tree structure is used for more efficient authentication of values

Merkle Hash Tree

- To authenticate V0, V1, …Vw-1, place them a leaf nodes of a binary tree

- "blind" all the values with a one-way hash function H to prevent disclosing additional values.

  $V'_i = H(V_i)$

- Use Merkle hash tree construction to commit to the values V'0, ... V'w-1

- Each internal node of the binary tree is derived from its two child nodes

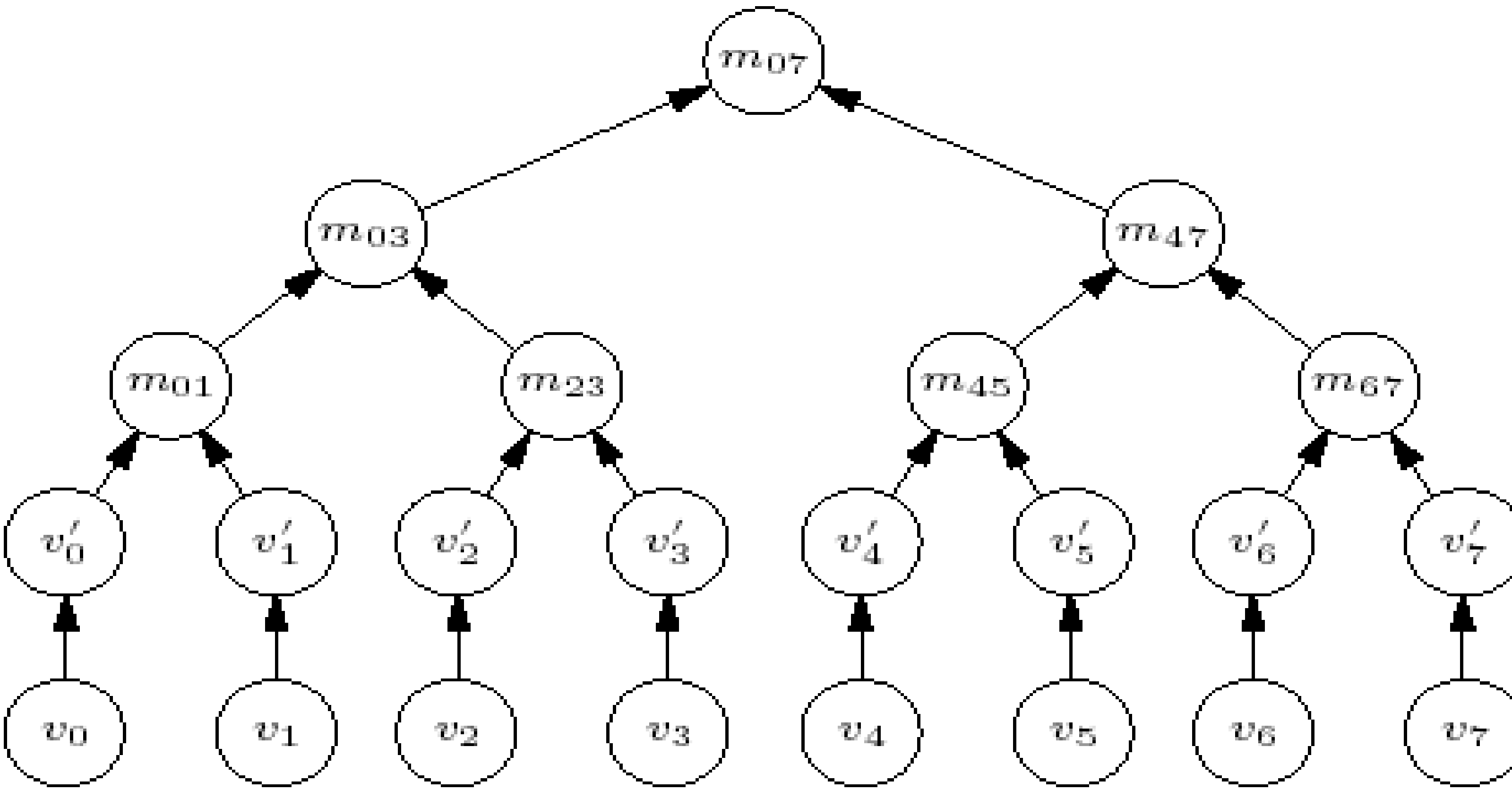
  $m\_parent = H(m\_left \| m\_right)$

•Example:

- Sender want to authenticate key v2
- It includes values v’3, m01, m47
- Receiver with an authentic root value m07 verify that

H[H[m01 || H[H[v2] || v’3]] || m47] == stored m07

- If the verification successful, the receiver knows that v2 is authentic



Hash Tree Optimization

- In TIK, the depth of the hash tree can be large
- Storing the entire tree is impractical
- Store only the upper layers of the tree, recompute lower layer on demand
- Node keeps two trees of depth d,
  - one fully computed and being used
  - one being filled in

TIK Protocol Description

TIK - TESLA with Instant Key Disclosure

- TIK implements a temporal leash and enables the receiver to detect a wormhole attack
- TIK is based on efficient symmetric cryptographic primitives
- TIK requires accurate time synchronization between all communicating parties
- TIK requires each communicating node to know just one public value for each sender, thus enabling scalable key distribution.
- Three stages in TIK protocol:
    - Sender setup
    - Receiver bootstrapping
    - Sending and Verifying Authenticated packets

Sender Setup

- To derive a series of keys $K_0, K_1, \ldots, K_w$:

  $K_i = F_x(i)$, where F is a pseudo-random function,
  x is a secret master key

- Determines a schedule for each of it's keys to expire

  K_0 expires at T_0,
  K_1 expires at $T_1 = T_0 + I$,
  K_i expires at $T_i = T_{i-1} + I = T_0 + i*I$

- Computationally intractable for an attacker to
  - find the master secret key x
  - derive a K_i without x

Receiver Bootstrapping

- Assume all nodes have synchronized clocks with max clock synchronization error Δ
- Assume each receiver knows every sender's
  - hash tree root m
  - associated parameters T0 and I
- This information is sufficient for the receiver to authenticate any packets from the sender

Sending and Verifying Authentication Packets

- Sender sends a Packet P
- Estimates upper bound t_r on the arrival time of the HMAC at the receiver
- Based on t_r, sender picks a key K_i, $T_i > t_r + \Delta$
- Sender discloses the key only after it expires
- Once the receiver gets the authentic key Ki, it can authenticate all packets that carry a message authentication code computed with Ki

Drawbacks

- Message authentication is delayed
- Receiver must wait for the key before it can authenticate the packet
- If nodes are tightly time synchronized, possible to remove authentication delay
- Sender can disclose the key in the same packet that carries the corresponding message authentication code

Sending and Receiving of a TIK packet

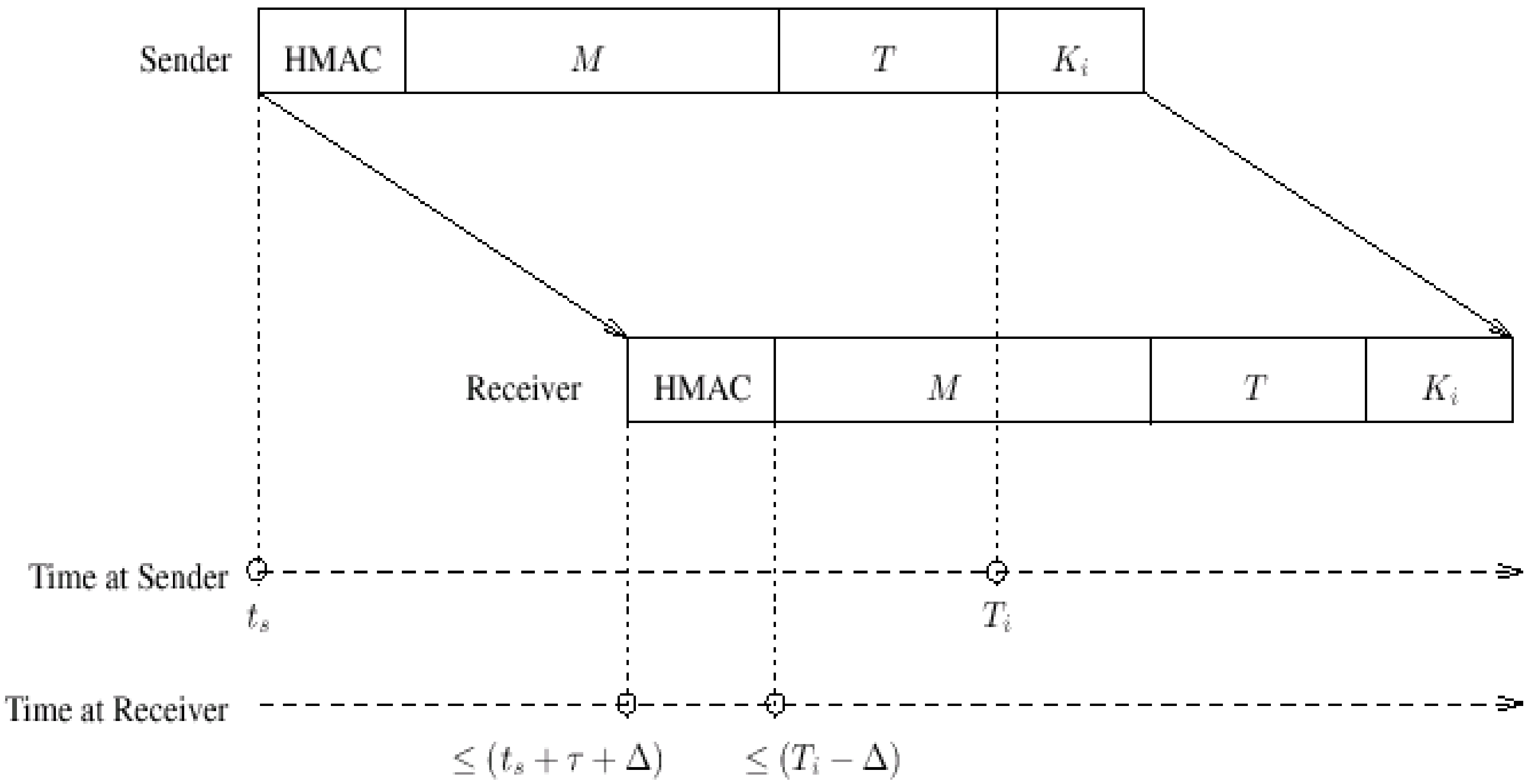


Figure 2: Timing of a packet in transmission using TIK

M - message payload

T - tree authentication values

Ki - key used to generate the HMAC

The TIK packet is transmitted by S as $S \rightarrow R: (HMAC_{Ki}(M), M, T, Ki)$

MAC Layer Considerations

- TDMA MAC protocol may be able to choose the time at which a frame begins transmission

- If MAC protocol uses RTS/CTS handshake, minimum packet size can be reduced by carrying HMAC inside RTC frame.

  A→B: (RTS, $HMAC_{Ki}$ (M))
  B→A: (CTS)
  A→B: (DATA, M, tree values, Ki)

TIK Performance

- Measured computational power and memory currently available in mobile devices

Pentium III	1GHz	1.3 million hashes/second
Compaq iPaq 3870 PocketPC	Linux	222,000 hashes/second

- In terms of memory consumption

iPaq 3870	32MB Flash, 64 MB of RAM
Modern notebooks	hundreds of Mbytes of RAM

Comparison Between Geographic and Temporal Leashes

Temporal Leashes	
pros	cons
Highly efficient, especially used with TIK	Tight time synchronization
	can not be used if max range < c Δ

Geographical Leashes	
pros	cons
Allowing them to detect tunnels through obstacles	increasing computation, network overhead
do not require tight time synchronization	location info increases overhead
can be used until maximum range is < 2vΔ	

Conclusion

- Wormhole

  A powerful attack that can have serious consequences on many proposed ad hoc network routing protocols.

- Packet leashes
  - To detect and defend against the wormhole attack.
  - Geographic and temporal leases.
- TIK

  To implement temporal leashes, and also provides instant authentication of received packets.