Wormhole detection

-- the wormhole attack
-- centralized and decentralized wormhole detection algorithms

Introduction

- many wireless networking mechanisms require that the nodes be aware of their neighborhood
- a simple neighbor discovery protocol:
  - every node broadcasts a neighbor discovery request
  - each node that hear the request responds with a neighbor discovery reply
  - messages carry node identifiers → neighboring nodes discover each other's ID
- an adversary may try to thwart the execution of the protocol
  - prevent two neighbors to discover each other by jamming
  - create a neighbor relationship between far-away nodes
    - by spoofing neighbor discovery messages (can be prevented by message authentication techniques)
    - by installing a wormhole (cannot be prevented by cryptographic techniques alone)
What is a wormhole?

- a wormhole is an out-of-band connection, controlled by the adversary, between two physical locations in the network
  - the adversary installs radio transceivers at both ends of the wormhole
  - it transfers packets (possibly selectively) received from the network at one end of the wormhole to the other end via the out-of-band connection, and re-injects the packets there into the network

- notes:
  - adversary’s transceivers are not regular nodes (no node is compromised by the adversary)
  - adversary doesn’t need to understand what it tunnels (e.g., encrypted packets can also be tunneled through the wormhole)
  - it is easy to mount a wormhole, but it may have devastating effects on routing

Effects of a wormhole

- at the data link layer: distorted network topology

  (a)  

  (b)  

  (c)  

  (d)  

  (e)  

  (f)  

- at the network layer:
  - routing protocols may choose routes that contain wormhole links
    - typically those routes appear to be shorter
    - flooding based routing protocols (e.g., DSR, Ariadne) may not be able to discover other routes but only through the wormhole
  - adversary can then monitor traffic or drop packets (DoS)
Wormholes are not unique to ad hoc networks

Classification of wormhole detection methods

- centralized mechanisms
  - data collected from the local neighborhood of every node are sent to a central entity
  - based on the received data, a model of the entire network is constructed
  - the central entity tries to detect inconsistencies (potential indicators of wormholes) in this model
  - can be used in sensor networks, where the base station can play the role of the central entity

- decentralized mechanisms
  - each node constructs a model of its own neighborhood using locally collected data
  - each node tries to detect inconsistencies on its own
  - advantage: no need for a central entity (fits well some applications)
  - disadvantage: nodes need to be more complex
Statistical wormhole detection

- each node reports its list of believed neighbors to the base station
- the base station reconstructs the connectivity graph (model)
- a wormhole always increases the number of edges in the connectivity graph
- this increase may change the properties of the connectivity graph in a detectable way (anomaly)
- detection can be based on statistical hypothesis testing methods (e.g. the $\chi^2$-test)

Examples

- a wormhole that creates many new edges may increase the number of neighbors of the affected nodes
  ⇒ distribution of node degrees will be distorted

- a wormhole is usually a shortcut that decreases the length of the shortest paths in the network
  ⇒ distribution of the length of the shortest paths will be distorted
Multi-dimensional scaling

- the nodes not only report their lists of neighbors, but they also estimate (inaccurately) their distances to their neighbors
- connectivity information and estimated distances are input to a multi-dimensional scaling (MDS) algorithm
- the MDS algorithm tries to determine the possible position of each node in such a way that the constraints induced by the connectivity and the distance estimation data are respected
  - the algorithm has a certain level of freedom in “stretching” the nodes within the error bounds of the distance estimation
- let us suppose that an adversary installed a wormhole in the network
  - if the estimated distances between the affected nodes are much larger than the nodes’ communication range, then the wormhole is detected
  - hence, the adversary must also falsify the distance estimation \( \rightarrow \) distances between far-away nodes become smaller
  - this will result in a distortion in the virtual layout constructed by the MDS algorithm

Examples

- in 1D:

  ![1D graph](image1.png)

  connectivity graph \hspace{1cm} reconstructed virtual layout

- in 2D:

  ![2D graph](image2.png)
Packet leashes

- packet leashes ensure that packets are not accepted “too far” from their source
- geographical leashes
  - each node is equipped with a GPS receiver
  - when sending a packet, the node puts its GPS position into the header
  - the receiving node verifies if the sender is really within communication range
- temporal leashes
  - nodes’ clocks are very tightly synchronized
  - when sending a packet, the node puts a timestamp in the header
  - the receiving node estimates the distance of the sender based on the elapsed time and the speed of light
    \[ d_{\text{est}} < v_{\text{light}}(t_{\text{rcv}} - t_{\text{snd}} + \Delta t) \]
  - note: \( v_{\text{light}} \Delta t \) must be much smaller than the communication range

TESLA with Instant Key-disclosure (TIK)

**idea:** authentication delay of TESLA can be removed in an environment where the nodes’ clocks are tightly synchronized

- by the time the sender reveals the key, the receiver has already received the MAC
- security condition: \( t_f < t_i - \Delta t + \tau_{\text{pkt}} \)
- note: \( \Delta t \) must be very small or otherwise packets must be very long
Mutual Authentication with Distance-bounding (MAD)

--- initialization phase ---
\[ r \in \{0, 1\}^t, s \in \{0, 1\}^t \]
compute commitment \( c_a = H(r|s|) \)
compute commitment \( c_b = H'(s|r) \)

--- distance-bounding phase ---
\( \alpha_i = r_i \)
\( \beta_i = s_i \)
\( \alpha_i = \alpha_i \oplus \beta_{i-1} \)
measure delay between \( \alpha_i \) and \( \beta_i \)
\( \beta_i = s_i \oplus \alpha_i \)
measure delay between \( \alpha_i \) and \( \beta_i \)

--- authentication phase ---
\( s_i = \alpha_i \oplus \beta_i \) (\( i = 1, \ldots, t \))
\( r_i = \alpha_i \) and \( r_i = \alpha_i \oplus \beta_{i-1} \) (\( i = 2, \ldots, t \))

\[ \mu_a = \text{mac}_{k_a}(r|u|s|s_1|\ldots|s_t|s_1) \]
\[ \mu_b = \text{mac}_{k_b}(r|u|s|s_1|\ldots|s_t|r_1) \]
\[ \overrightarrow{\mu_a'} \overrightarrow{\mu_b'} \]
verify \( c_a \) and \( \mu_a \)
verify \( c_b \) and \( \mu_b \)

- MAD allows precise distance estimation without synchronized clocks

Wormhole detection

Using position information of anchors

- anchors are special nodes that know their own positions (GPS)
- there are only a few anchors randomly distributed among regular nodes
- two nodes consider each other neighbors only if
  - they hear each other and
  - they hear more than \( T \) common anchors
- anchors put their location data in their messages
- transmission range of anchors (\( R \)) is larger than that of regular nodes (\( r \))
- wormholes are detected based on the following two principles:
  1. a node should not hear two anchors that are 2\( R \) apart from each other
  2. a node should not receive the same message twice from the same anchor
**Principle 1**

- x hears anchors in $A_x$ and in $A_o$
- $P_1$ is the probability that it hears two anchors that are farther away from each other than $2R$
- the probability that there is at least one anchor in an area of size $S$ is $(1-e^{-\lambda S})$, where $\lambda$ is the density of anchors
- $P_1 \geq (1-e^{-\lambda S_x})(1-e^{-\lambda S_O})$, where $S'_x$ is the size of $A'_x$ and $S'_O$ is the size of $A'_O$
- this lower bound is maximum when $S'_x = S'_O$

**Principle 2**

- when $x$ and $O$ are closer than $2R$, the discs $A_x$ and $A_O$ overlap
- if there is an anchor in the intersection $A_{xO}$, then the messages of that anchor is heard twice by $x$
  - first directly and then from transceiver $D$ who receives it from $O$ through the wormhole
- the probability $P_2$ of detection is equal to the probability that there is at least one anchor in $A_{xO}$
- $P_2 = 1-e^{-\lambda S_{xO}}$
Wormhole detection with directional antennas

- when two nodes are within each other’s communication range, they must hear each other’s transmission from opposite directions
- if nodes x and y communicate through a wormhole, then this condition is not always satisfied:

  ![Diagram](image)

- but this doesn’t always work:

  ![Diagram](image)

Using verifiers – the idea

- if y and x were real neighbors and y heard x in zone 4, then every node in y’s zone 4 would be a neighbor of x
- if they are not real neighbors, then there may be a node v in y’s zone 4 that is not a neighbor of x (v and x don’t hear each other from opposite directions)
- such a v can be used by y as a verifier

![Diagram](image)
### Conditions for being a verifier

- if node $y$ hears $v$ in the same zone in which it hears $x$, then $y$ may hear both $x$ and $v$ through the wormhole
  - for a valid verifier $v$, $y$ must hear $v$ and $x$ from different zones (i.e., $Z_{yx} \neq Z_{yx}$ must hold)

- if $v$ hears $x$ in the same zone in which $y$ hears $x$ (i.e., $Z_{vx} = Z_{yx}$), then they may both hear $x$ through the wormhole’s transceiver
- if, in addition, $x$ happens to hear the other transceiver of the wormhole in zone $Z_{yx}$, then $x$ can establish neighbor relationships with both $y$ and $v$
  - for a valid verifier $v$, $v$ must hear $x$ from a zone different from the one in which $y$ hears $x$ (i.e., $Z_{vx} \neq Z_{yx}$ must hold too).

### Using verifiers – the mechanism

- $y$ accepts $x$ as a neighbor if
  - they hear each other from opposite zones
  - there’s at least one valid verifier $v$ such that $x$ and $v$ hear each other from opposite zones
- how does this detect wormholes?
  - let us assume that $y$ hears $x$ through the wormhole
    - one end of the wormhole is near to $x$, the other end is in zone $Z_{yx}$
  - let us further assume that $v$ is a valid verifier
    - first condition ($Z_{yv} \neq Z_{vy}$) is satisfied
      - $y$ hears $v$ directly (since $y$ hears $v$ from a zone different from $Z_{vy}$)
      - $x$ hears both $y$ and $v$ through the wormhole
    - second condition ($Z_{vx} = Z_{yx}$) is satisfied
      - $x$ and $v$ cannot hear each other from opposite zones
        - let’s assume that $Z_{vx} = Z_{yx}$
          - we know that $x$ hears both $y$ and $v$ through the wormhole $\Rightarrow Z_{vx} = Z_{yx}$
          - in addition, we know that $Z_{vx} = Z_{yx}$ (otherwise $y$ would not consider $x$ as a potential neighbor)
            - $Z_{vx} = Z_{yx} = Z_{vy} \Rightarrow Z_{vx} = Z_{yx}$ (contradicts the second condition)
        - no valid verifier $v$ exists such that $x$ and $v$ hear each other from opposite zones $\Rightarrow y$ will not accept $x$ as a neighbor
Summary

- A wormhole is an out-of-band connection, controlled by the adversary, between two physical locations in the network.
- A wormhole distorts the network topology and may have a profound effect on routing.
- Wormhole detection is a complicated problem.
  - Centralized and decentralized approaches:
    - Statistical wormhole detection
    - Wormhole detection by multi-dimensional scaling and visualization
    - Packet leashes
    - Distance bounding techniques
    - Anchor assisted wormhole detection
    - Using directional antennas
  - Many approaches are based on strong assumptions:
    - Tight clock synchronization
    - GPS equipped nodes
    - Directional antennas
    - ...
- Wormhole detection is still an active research area.