Introduction

- Model
  - wireless sensor network
  - multi-hop
  - severely constrained devices
- Goal
  - providing confidentiality of transmitted message
- Problems
  - devices can be captured by adversary
  - all secrets stored can be accessed easily
- Solution
  - message partition
  - routing algorithm
  - adversary needs to capture specific subset of nodes to learn the message

Protocol

- Initialization
  - source splits $M$ into $I$ parts, and sends them to $I$ nodes in next layer
- Step
  - each of $I$ devices splits the message part again into $I$ parts and sends them to $I$ devices in next layer
  - each device chooses the same $I$ devices to send the message to (See Routing).
- at each layer, a subset of $I$ nodes participates in message $M$ transmission
  - $I$ is a protocol (forking) parameter
  - there are $I^2$ message parts traveling between each layer

Model

![Exemplary distribution of nodes and layers](image)

- layered structure (possibly constructed ad-hoc)
  - $L_1, L_2, \ldots, L_t$
  - assuming no overlaps
  - assuming $n$ devices in each layer
  - each pair of nodes in consecutive layers share a symmetric key
  - nodes in consecutive layers are in transmission range of each other

Adversary

- wants to learn the message transmitted through the network
- can capture some devices
- can eavesdrop communication in the network
- can retrieve all information stored on captured device

Protocol: Message Partition

- Initialization
  - source chooses $r$ at random and sends it to $I$ devices in next layer
  - the devices basing on $r$ determine set of receivers in next layer
- Step
  - each device generates new random value $r'$ and sends it along with its $I$ message parts
  - each device in the following layer has the same set of $I, r'$ and can determine receivers in next layer

Protocol: Routing

- Initialization
  - each layer, a subset of $I$ nodes participates in message $M$ transmission

Attack scenarios

- Nonadaptive Attack
  - Adversary chooses devices before transmission
- Random Attack
  - Adversary chooses devices at random (e.g. trying to locate them in grass)
- Adaptive attack
  - Nodes chosen after transmission

Security analysis

- Nonadaptive Attack
  - It is optimal for the adversary to choose devices from one layer
  - Theorem: Corrupting nodes from one layer adversary chance for learning the message for $I \leq K < n$ is $\binom{n}{I}/\binom{n}{I}$
- Random Attack
  - Theorem: Chance for learning the message is
  \[
  S_n = \sum_{i=1}^{I} (-1)^{i+1} \binom{I}{i} \binom{K}{i} < \frac{1}{2} \left( 1 + \frac{1}{p} \right)^I - \left( 1 - \frac{1}{p} \right)^I 
  \]
- Adaptive attack
  - Attack when single message is transmitted is trivial
  - For $N$ messages going simultaneously we have the following
  - Theorem: For $\log(N)N < Np$ following relation holds:
  \[
  \Pr \left[ \max_{n=1}^{K} \frac{Np}{1 + 1.5, N \log (N) } \right] \leq \frac{1}{N} 
  \]
  - Where $\max_{n=1}^{K}$ denotes number of learned messages and $p = \binom{n}{I}/\binom{n}{I}$

Summary and Extensions

- Secret sharing can employ some error correcting codes to improve robustness
- MAC sum can be attached in order to prevent modifications of transmitted message
- Parameter $I$ is a trade-off between security and communication complexity
- $I^2$ is not much as even $I = 2$ is significant security improvement

\[ \text{Created with IPython notebooks http://nbviewer.inkp.net/nbviewer/inkp.net/nbviewer/notebooks/inkp.net/nbviewer/notebooks/IPython/notebooks} \]