Generalized "Yoking-Proofs" for a Group of RFID Tags

Leonid Bolotnyy and Gabriel Robins

lb9xk@cs.virginia.edu, robins@cs.virginia.edu

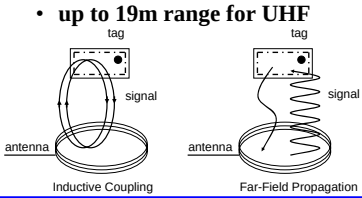
Department of
Computer Science
University of Virginia
School of Engineering & Applied Science
www.cs.virginia.edu/robins

Radio-Frequency Identification (RFID)

Radio-Frequency Identification (RFID) RFID reader

Three types of RFID tags

- Passive / Active / Semi-Active
- up to 19m range for UHF



RFID Tag "Yoking-Proofs"

Problem Statement

- The goal is to generate a proof that a group of passive RFID tags were identified nearly-simultaneously
- Reader should not be able to forge the proof
- The proof is efficiently verifiable off-line by a trusted verifier

Applications – verify that:

- a bottle of medicine was sold together with its instructions
- safety devices were sold together with tools
- matching parts were delivered together
- several forms of ID were presented
- a group of people were present at a meeting

Generalized Tag Group "Yoking-Proof"

Assumptions

- Tags are passive
- Tags have limited computational abilities
- Tags can compute a keyed hash function
- Tags can maintain some state
- Verifier is trusted and powerful

Solution Goals

- Allow readers to be adversarial
- Make valid proofs improbable to forge
- Allow verifier to verify the proof off-line
- Detect replays of valid proofs

Timer on-board a tag

- FCC regulations: protocol termination < 400ms
- Capacitor discharge can implement timeout

"Yoking":

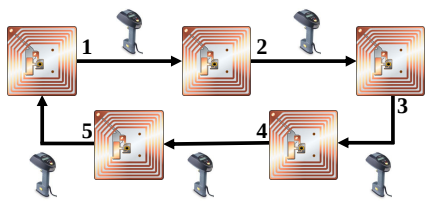
suggests joining together, or the simultaneous presence of multiple tags.

Key Observation:

Passive RFID tags can communicate with each other through the reader.

Key Idea:

Construct a circular chain of mutually dependent message authentication code (MAC) computations.



Notation

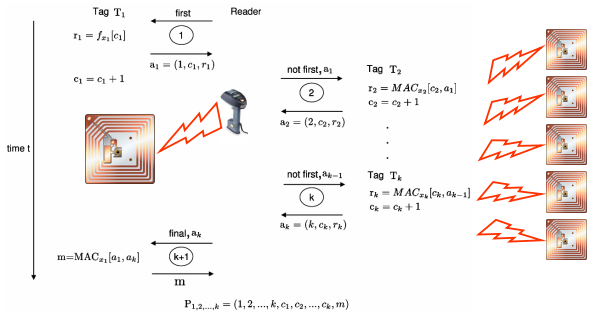
Keyed hash function: $f : \{0,1\}^d \times \{0,1\}^* \rightarrow \{0,1\}^d$

Message authentication code: $MAC : \{0,1\}^d \times \{0,1\}^* \rightarrow \{0,1\}^d$

Computation with a secret x on input m: $f_x[m]$ and $MAC_x[m]$

Theorem

Given random-oracle assumptions for f and MAC, the success probability of proof forgery by an adversary of group yoking protocol is bounded above by 2^{-d}.



Group Yoking Protocol

Algorithm 1: Tag Initialization

for $i = 1$ **to** n **do**
  $c_i = 0$
  $x_i \leftarrow$ choose *randomly* from $\{0,1\}^d$
end

Algorithm 2: Reader

$send(first, 0)$ to T_1
$receive(a_1 = (T_1, c_1, r_1))$ from T_1
for $i = 2$ **to** k **do**
  $send(not\ first, a_{i-1})$ to T_i
  $receive(a_i = (T_i, c_i, r_i))$ from T_i
end
$send(final, a_k)$ to T_1
$receive(m)$ from T_1
construct $P_{1,2,...,k} = (1, 2, \ldots, k, c_1, c_2, \ldots, c_k, m)$

Algorithm 3: Tag

Input: Tag's order in the chain: *mode* and output value from the previous tag in the chain: *value*
switch *mode* **do**
  case *'first'*
    start timer
    $r = f_c[c]$
    $a = (ID, c, r)$
    $store(a)$
    $send(a)$
    $c = c + 1$
    break
  end
  case *'not first'*
    $r = MAC_x[c, value]$
    $c = c + 1$
    $send(a = (ID, c, r))$
    break
  end
  case *'final'*
    if *timer has not timed out* **then**
      $m = MAC_x[a, value]$
      $send(m)$
      timer times out
    else
      abort
    end
    break
  end
end

Algorithm 4: Verifier

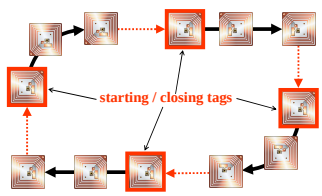
Input: Proof $P = (1, 2, \ldots, k, c_1, c_2, \ldots, c_k, m)$
$r_1 = f_{x_1}[c_1]$
$a_1 = (1, c_1, r_1)$
for $i = 2$ **to** k **do**
  $r_i = MAC_{x_i}[c_i, a_{i-1}]$
  $a_i = (i, c_i, r_i)$
end
$m^* = MAC_{x_1}[a_1, a_k]$
if $m == m^*$ **then**
  return *success*
else
  return *failure*
end

Anonymous Yoking algorithm variant: tags keep their identities private.

Algorithmic Speedups

Key Idea:

Split circular chain into a group of arcs, where each arc consists of a sequence of dependent message authentication code (MAC) computations; adjacent arcs are dependent.



Speedup requires:

- multiple readers or a reader with multiple antennas
- medium access control protocol that avoids collisions

Future Research

- Develop new RFID "yoking-proofs" where tags communicate with each other through the reader
- Reduce the cost of tags to under 5 cents a piece
- Provide "good enough" security and privacy guarantees
- Standardize the technology for different classes of tags
- Devise novel applications that would benefit the society