Terrorist Fraud resistant nanosecond-scale Distance Bounding

Boris Škorić
TU Eindhoven

Crypto Working Group
May 13, 2011
Terrorist fraud resilient distance Bounding with Analog components
Work in progress.

Joint work with
Srdjan Čapkun, Aanjhan Ranganathan, Nils Ole Tippenhauer
(System Security Group, ETH Zürich).
• Types of relay attack
• Distance bounding
  - Swiss Knife
• Analog hardware
  - challenge reflection
  - channel selection
  - limitations
• New scheme
  - modified analog circuit
  - adapted Swiss Knife
Why distance bounding?

Authentication alone may not be sufficient
• physical access to buildings etc.
  - watch out for relay attack

Two main types of attack
• Mafia Fraud
• Terrorist Fraud
Relay attacks: Mafia Fraud

**Authentication without distance checking**
- Correct response
- from legitimate tag
- ... but attacker gets access!

Famous urban myth: Mig-in-the-middle attack
Relay attacks: Terrorist Fraud

More powerful than Mafia fraud:
• legit device does not have to be tricked
• device can provide more info than just response
Countermeasures

What to do against relay attacks?

• Ask the prover where he is
  - but he could be lying
• Signal strength
  - can be spoofed
• Measure the distance to the prover
  - “distance bounding”
  - nothing travels faster than light $c = 2.99792458 \cdot 10^8 \text{ m/s}$
  - infer distance from traveling time of signal

  300 meters per microsecond
Distance bounding

Demand response within time $t_{\text{max}}$
- travel time to distance $x_{\text{max}}$ and back
- allow some “slack” time for computations
- dist. measurement & demonstration of knowledge at the same time

$$t_{\text{max}} = 2 \frac{x_{\text{max}}}{c} + t_{\text{slack}}$$

$$x_{\text{spoofable}} = \frac{1}{2} c t_{\text{max}} = x_{\text{max}} + \frac{1}{2} c t_{\text{slack}}$$

has to be very small
Distance bounding: practical problems

\( t_{\text{slack}} \) must be very small

- no (heavy) computations
  - addition lasts too long
  - *but still cryptographic challenge-response protocol!*

- delays inside prover device become problematic
  - missed cycles, bus speed, etc

- no error correction
  - live with transmission errors
Solving the practical problems

• no (heavy) computations
  - split protocol into slow and quick phase
  - prover creates LUT in slow crypto phase
  - verifier: unpredictable selection from LUT in quick phase

• delays inside prover
  - LUT sitting right “next to” emitter

• no error correction
  - decide afterwards if there were transmission errors
Swiss Knife protocol (Kim et al. 2008)

Reader has DB \{ID, x\}

Random $N_A$;  
random $d$ (Hamm.weight $m$)

$N_A, d$  
$N_B$

Tag (ID, x)

Random $N_B$

$Z^0 = f_x(C_B, N_B)$;  
$Z^l = Z^0 \oplus x$;

For $i = 1$ to $m$ \{ $j =$ index of next 1 in $d$; 

$R^0_i = Z^0_j$;  
$R^l_i = Z^l_j$ \}

For $i = 1$ to $m$

Random bit $c_i$; start clock

$c_i'$  
$r_i$

Stop clock; store $\Delta t_i$

Find matching (ID, $x$) in DB;  
compute $R^0, R^l$;

$err_c = \#\{i: c_i' \neq c_i\};$

$err_r = \#\{i: c_i' = c_i \land r_i \neq R^{c_i}_i\};$

$err_t = \#\{i: c_i' = c_i \land \Delta t_i > \Delta t_{max}\};$

if $err_c + err_r + err_t \geq T$ reject;

$t_A = f_x(N_B)$

$t_B, c_1', \ldots, c_m'$  
$t_B = f_x(c_1', \ldots, c_m', \text{ID}, N_A, N_B)$

$t_A$  
Check $t_A$
Still too slow!

State of the art hardware:

• analog → digital conversion: 50 ns
• all conversion steps together: 170 ns
  (26 meters)

Only analog processing is fast enough!
Analog challenge-response

Rasmussen & Čapkun 2010

- Brands-Chaum with analog response.
- **CRCS**: Challenge Reflection with Channel Selection.

Challenge: unpredictable signal $c(t)$ at frequency $f_c$

Response: reflection of $c(t)$ at shifted frequency

![Graph showing challenge and response frequencies](image)
Challenge Reflection with Channel Selection

< 1 nanosecond!

Mixer

Voltage Controlled Oscillator (VCO)

Response

< 1 nanosecond!
Security of Rasmussen-Čapkun

- [Same as Brands-Chaum]
- Secure against Mafia Fraud
- NOT against Terrorist Fraud
  - need AD conversion for challenge interpretation

Swiss Knife

Rapid bit exchange

For $i = 1$ to $m$

Random bit $c_i$; start clock

Stop clock; store $\Delta t_i$

$r_i = \begin{cases} R^0_i & \text{if } c_i' = 0 \\ R^1_i & \text{if } c_i' = 1 \end{cases}$
Generalized CRCS

Doing the register choice with analog hardware

• Challenge $c(t)$ at freq $\omega_0$ or $\omega_1$.
• Two CRCS circuits in parallel.

Danger: malicious verifier may read out both registers.
Protocol adaptation

• Problem: readout of both registers
  - attacker learns the secret
  - detection takes time
  - need to respond immediately
• Solution: masking
  - commit to random mask
  - do rapid part with masked registers
  - if no cheating, then open commitment
Verifier has DB \{ID, x\}

<table>
<thead>
<tr>
<th>Random (N_A); random (d) (Hamm.weight (m))</th>
<th>Prover (ID, (x))</th>
</tr>
</thead>
<tbody>
<tr>
<td>(N_A, d)</td>
<td>(\text{Random } M^0, M^1, N_B); (F = f_x(M^0, M^1))</td>
</tr>
<tr>
<td>(F, N_B)</td>
<td>(Z^0 = f_x(C_B, N_B); Z^1 = Z^0 \oplus x;)</td>
</tr>
</tbody>
</table>

For \(i = 1\) to \(m\) \{ \(j = \text{index of next } 1\) in \(d\); \(R^0_i = Z^0_j; R^1_i = Z^1_j\) \}

\(T^0 = R^0 \oplus M^0; T^1 = R^1 \oplus M^1\)

---

**Rapid bit exchange using CRCS**

<table>
<thead>
<tr>
<th>Random bit (b_i); Random signal (c_i(t)) at freq. (\omega_{b_i}) Record (r_i(t)) and delays (\Delta t_i)</th>
<th>For (i = 1) to (m) Circuit reflects signal at freq. (\omega_{b_i} + (2T^{b_i}<em>i - 1)\omega</em>\Delta) Slow interpretation of (b_i').</th>
</tr>
</thead>
<tbody>
<tr>
<td>(c_i(t))</td>
<td>(r_i(t))</td>
</tr>
</tbody>
</table>

---

Find matching (ID, \(x\)) in DB; Check if \(F = f_x(M^0, M^1)\); Compute \(T^0, T^1\);

\(\text{err}_b = \#\{i: b'_i \neq b_i\};\)

\(\text{err}_f = \#\{i: b'_i = b_i \land r_i \text{ has wrong freq.}\};\)

\(\text{err}_r = \#\{i: b'_i = b_i \land r_i \text{ differs too much from } c_i\};\)

\(\text{err}_t = \#\{i: b'_i = b_i \land \Delta t_i > \Delta t_{\text{max}}\};\)

Reject if \(\text{err}_b + \text{err}_f + \text{err}_r + \text{err}_t\) too large;

\(t_A = f_x(N_B)\)

Proceed only if no cheating detected;

\(t_B = f_x(b_1',...,b_m',\text{ID},M^0,M^1,N_A,N_B)\)

\(t_A\) Check \(t_A\)
• Distance bounding: absurd timing requirements
• Analog challenge-response, CRCS
  - secure against Mafia Fraud, but not Terrorist Fraud
• Generalized CRCS and extra masking step
  ➡ nanosecond-scale responses
  ➡ security against Terrorist Fraud
  - not restricted to Swiss Knife
• Embarrassingly trivial-looking