

Cryptography in Radio Frequency Identification and Fair Exchange Protocols

Gildas Avoine

EPFL, Lausanne, Switzerland

Fair Exchange

Brief Recall and Contributions

Optimistic Fair Exchange Without Centralized TTP

Radio Frequency Identification

Brief Recall and Contributions

Attack on Henrici and Müller's RFID Protocol

Attack on Molnar and Wagner's Technique

Time-Memory Trade-Off in RFID

Fair Exchange

Fair Exchange

Definition

Two-Party Fair Exchange Protocol

An exchange protocol between two parties P_o and P_r is a protocol in which P_o and P_r possess some items m_o and m_r respectively and aim at exchanging them.

We say that the protocol ensures **fairness** if it terminates so that either P_o gets m_r and P_r gets m_o , or nobody gets information about the expected items.

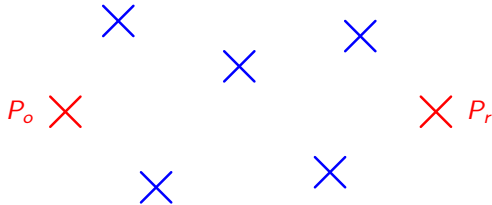
▷ Probabilistic 2-FE and n -FE

- ▶ No centralized trusted third party.
- ▶ Each participant has a guardian angel to prevent misbehavior.
- ▶ Fairness is probabilistic.
- ▶ Probability of unfairness can be made arbitrarily low.
- ▶ Deterministic fairness if majority of honest participants (n -FE).

▷ Optimistic 2-FE relying on neighbors

- ▶ No centralized trusted third party.
- ▶ Fairness relies on the neighbors in the network.
- ▶ Neighbors are involved only in case of conflict.
- ▶ Neighbors learn nothing about the expected items.

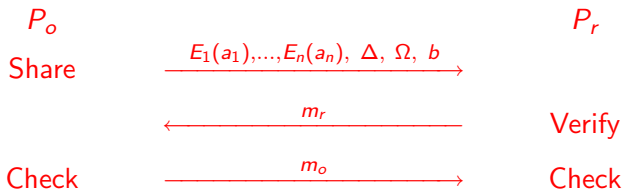
Towards a New Approach



- ▷ We know that some neighbors are **honest**.
- ▷ We don't know **who** is honest.
- ▷ Even honest neighbors are **curious**.

Optimistic Fair Exchange Within a Network

Optimistic 2-FE protocol based on a **publicly verifiable secret sharing**



Initial Agreement Before Exchange

P_o and P_r agree on the mathematical **description** of the items they want to exchange (e.g. $\text{descr}(m) = g^m$).



P_o and P_r establish the **contract**:

$$\Omega = S_o(P_o \| P_r \| \text{descr}(m_o) \| \text{descr}(m_r) \| D \| k).$$

Publicly Verifiable Secret Sharing

A PVSS is a protocol that is used to share a secret m among several participants such that only some **specific subsets** of participants can recover m by collusion and **anybody** can check the shares.

▷ Distribution:



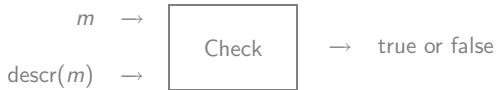
▷ Verification:



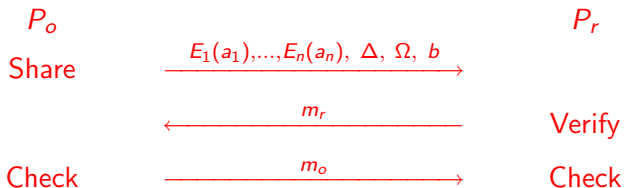
▷ Reconstruction:



Additional Primitives

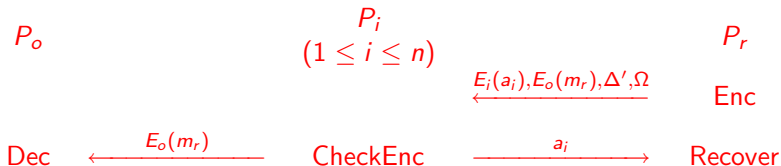


Main Protocol



- ▶ P_o picks a random a and computes b such that $m_o = a + b$.

Recovery Protocol



- ▶ if **CheckEnc**($E_o(m_r), \text{descr}(m_r), \Delta'$) is true and D has not expired, P_i sends a_i to P_r and $E_o(m_r)$ to P_o .
- ▶ After having received k shares, P_r runs **Recover**.
- ▶ From a , P_r computes $m_o = a + b$.

Assumptions on Channels

- ▶ P_r knows a constant $T_{\max} < +\infty$ such that messages from P_r to any neighbor are always delivered within T_{\max} .
- ▶ Recovery protocol is started before $D - T_{\max}$ by P_r .
- ▶ All messages from honest neighbors are **eventually delivered**.

Assumptions on Neighbors

- ▷ \mathcal{P}_{or} : neighbors who honestly collaborate with both P_o and P_r .
- ▷ \mathcal{P}_r : neighbors who may harm P_o by colluding with P_r .
- ▷ \mathcal{P}_o : neighbors who may harm P_r by colluding with P_o .
- ▷ $\mathcal{P}_{\bar{or}}$: neighbors who do not collaborate at all.

Theorem

If $|\mathcal{P}_r| < k \leq |\mathcal{P}_r| + |\mathcal{P}_{or}|$ then fairness is ensured.

- ▷ If P_r is dishonest, P_r should not be able to recover m_o with his colluders only: $|\mathcal{P}_r| < k$.
- ▷ If P_o is dishonest, we must ensure that P_r can recover m_o :
 $k \leq |\mathcal{P}_r| + |\mathcal{P}_{or}|$.

Numerical Examples

Example

If P_o and P_r know that there is a majority of honest neighbors in the network i.e. $|\mathcal{P}_{or}| > \frac{n}{2}$ then we take $k = \lceil \frac{n}{2} \rceil$.

Example

Let's take $n=100$. If P_o knows that at least 40% of the network is honest with him (i.e. $|\mathcal{P}_{or}| + |\mathcal{P}_o| \geq \frac{2n}{5}$) and P_r knows that at least 70% of the network is honest with him (i.e. $|\mathcal{P}_{or}| + |\mathcal{P}_r| \geq \frac{7n}{10}$) then we can take k such that $60 < k \leq 70$.

Protocol Properties

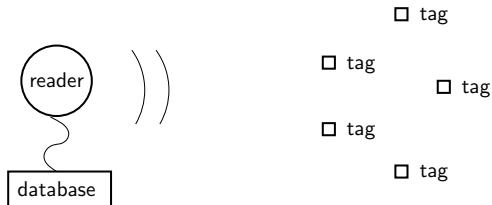
- ▶ First **optimistic fair exchange** protocol which does not rely on a centralized trusted third party.
- ▶ Our protocol ensures **fairness**.
- ▶ Our protocol ensures **privacy**.

Radio Frequency Identification

Definition

RFID

Radio Frequency Identification (**RFID**) is a method of remotely **identifying** objects or subjects using transponders (**tags**) queried through a **radio frequency** channel.



Applications: Barcodes, identification of livestock, access control, e-passports, etc.

Problem

An adversary should not be able to **track** people thanks to the RFID tags they carry.

Goal

Design an RFID protocol that ensures **untraceability** and which relies only on **symmetric** cryptography.

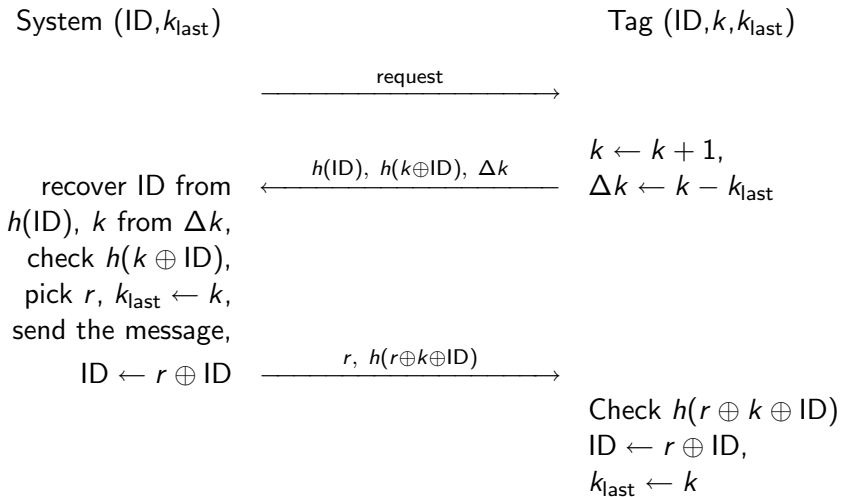
Thesis Contributions in Radio Frequency Identification

- ▶ Link between traceability and communication model.
- ▶ Attacks on existing protocols (JuelsP, HenriciM, SaitoRS, etc.).
- ▶ Attack on Molnar and Wagner's technique.
- ▶ Technique based on a Time-Memory Trade-Off.

Protocols

Protocol	Weaknesses pointed out by
[JuelsP03]	[Avoine04], [ZhangK05]
[VadjaB03]	[VadjaB03]
[GolleJJS04]	[Avoine05], [SaitoRS04]
[Juels04]	[Juels04]
[HenriciM04]	[AvoineO05]
[SaitoRS04]	[Avoine05]
[JuelsW05]	[GilbertRS05]
[WeisSRE02]	
[OhkuboSK03]	
[FeldhoferDW04]	
[MolnarW04]	
[RheeKKW05]	

Henrici and Müller's Protocol



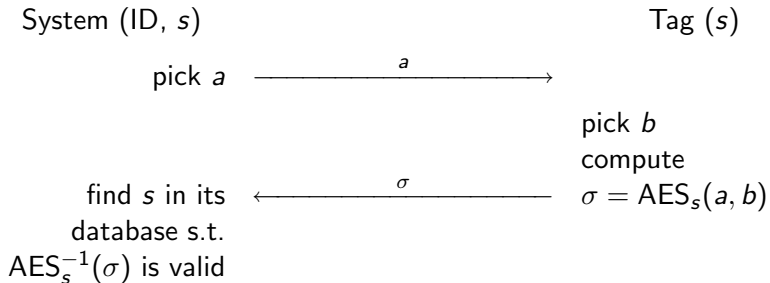
Attacks on Henrici and Müller's Protocol

- ▷ Attack based on lack of randomness.
 - ▶ Taking advantage of the information supplied by Δk .
- ▷ Attack based on desynchronization.
 - ▶ Desynchronizing the counters shared by tag and system.

Protocols

Protocol	Weaknesses pointed out by
[JuelsP03]	[Avoine04], [ZhangK05]
[VadjaB03]	[VadjaB03]
[GolleJJS04]	[Avoine05], [SaitoRS04]
[Juels04]	[Juels04]
[HenriciM04]	[AvoineO05]
[SaitoRS04]	[Avoine05]
[JuelsW05]	[GilbertRS05]
[WeisSRE02]	
[OhkuboSK03]	
[FeldhoferDW04]	
[MolnarW04]	
[RheeKKW05]	

Feldhofer, Dominikus, and Wolkerstorfer's Protocol

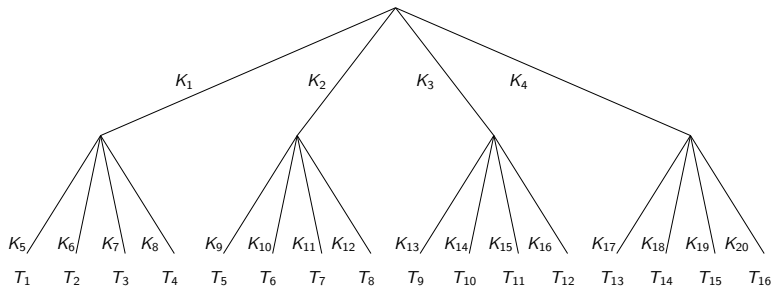


Computation Complexity of Challenge-Response Protocols

- ▶ An **exhaustive search** in the system's database is required to identify one tag.
- ▶ **Complexity too high** in particular in case of inventory.
- ▶ Is it possible to design an RFID protocol with a complexity **better than linear**?
- ▶ Molnar and Wagner proposed a solution that reduces the complexity of any challenge-response from $O(n)$ to $O(\log n)$.

Molnar and Wagner's Tree-Based Technique

- ▶ Each tag stores $\log_{\delta}(n)$ keys.



- ▶ A challenge-response is applied at each level of the tree.
- ▶ Instead of carrying out **1** exhaustive search in a set of size n , $\log_{\delta}(n)$ exhaustive searches are performed in sets of size δ .

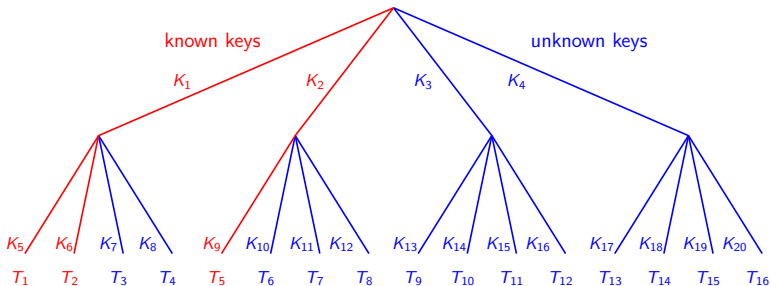
Numerical Example

Example

In a library, we consider 2^{20} tagged books. We assume that the system can carry out 2^{23} operations per second. Identifying one tag requires **0.1 milliseconds** ($\delta = 1024$) and identifying the whole system requires **2 minutes** ($\delta = 1024$) or **2 seconds** ($\delta = 2$).

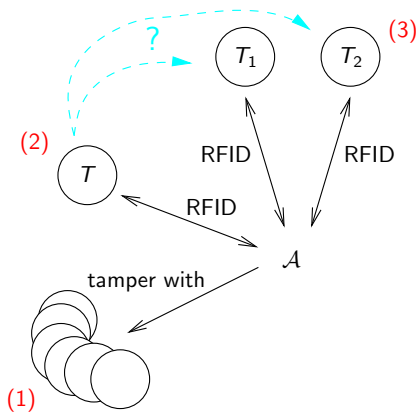
Drawbacks

- ▷ The tags share some keys.
- ▷ Tampering with tags gives information about the other tags.



How to Trace a Tag

- (1) Tamper with k tags.
- (2) Choose any target T and query it at will.
- (3) Query T_1 and T_2 to determine which of the two is T .



Five Cases to Analyze

- ▶ T_1 on **known** branch and T_2 on **unknown** branch: success.
- ▶ T_2 on **known** branch and T_1 on **unknown** branch: success.
- ▶ T_1 and T_2 both on **known** but different branches: success.
- ▶ T_1 and T_2 both on **unknown**: failure.
- ▶ T_1 and T_2 both the same **known** branch: failure at level i but the attack moves on to level $i + 1$.

Probability of Success – Formula

The probability that the attack succeeds is

$$\frac{k_1}{\delta^2} (2\delta - k_1 - 1) + \sum_{i=2}^{\log_{\delta}(n)} \left(\frac{k_i}{\delta^2} (2\delta - k_i - 1) \prod_{j=1}^{i-1} \frac{k_j}{\delta^2} \right),$$

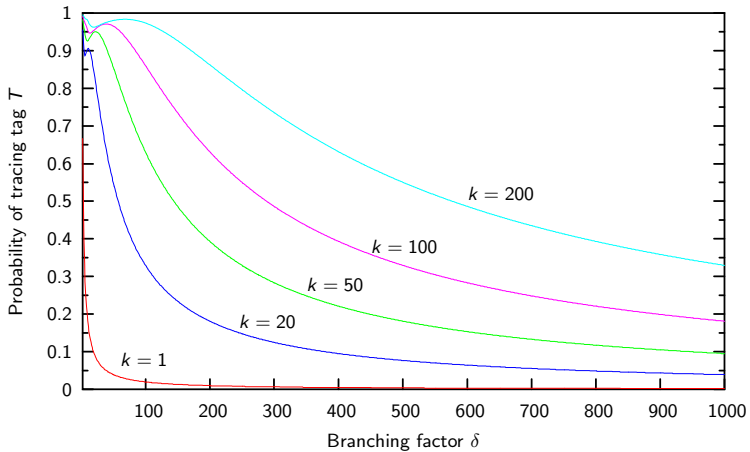
where

$$k_1 = \delta \left(1 - \left(1 - \frac{1}{\delta} \right)^k \right) \quad k_{i>1} = \delta \left(1 - \left(1 - \frac{1}{\delta} \right)^{g(k_i)} \right)$$

and

$$g(k_i) = k \prod_{j=1}^{i-1} \frac{1}{k_j}.$$

Probability of Success – Graph



Probability of Success – Table

$k \backslash \delta$	2	20	100	500	1000
1	66.6%	9.5%	1.9%	0.3%	0.1%
20	95.5%	83.9%	32.9%	7.6%	3.9%
50	98.2%	94.9%	63.0%	18.1%	9.5%
100	99.1%	95.4%	85.0%	32.9%	18.1%
200	99.5%	96.2%	97.3%	55.0%	32.9%

Protocols

Protocol	Weaknesses pointed out by
[JuelsP03]	[Avoine04], [ZhangK05]
[VadjaB03]	[VadjaB03]
[GolleJJS04]	[Avoine05], [SaitoRS04]
[Juels04]	[Juels04]
[HenriciM04]	[AvoineO05]
[SaitoRS04]	[Avoine05]
[JuelsW05]	[GilbertRS05]
[WeisSRE02]	
[OhkuboSK03]	
[FeldhoferDW04]	
[MolnarW04]	
[RheeKKW05]	

Protocol Description

System (ID_i, s_i^1)

Tag (s_i^k)

request →

← $r_i^k := G(s_i^k)$

$s_i^{k+1} = H(s_i^k)$

- ▶ Replay attacks are possible.
- ▶ Ensure forward untraceability.

Computations Needed to Identify one Tag

Receiving r_i^k , the system computes from the initial secrets s_i^1 the hash chains until it finds r_i^k or until it reaches a given maximum limit m on the chain length.

$$\begin{array}{ccccccc} s_1^1 & \rightarrow & r_1^1 & r_1^2 & \dots & \dots & r_1^{m-1} & r_1^m \\ s_2^1 & \rightarrow & r_2^1 & r_2^2 & \dots & \dots & r_2^{m-1} & r_2^m \\ \vdots & \rightarrow & \dots & \dots & \dots & \dots & \dots & \vdots \\ s_i^1 & \rightarrow & \dots & \dots & \boxed{r_i^k = G(H^{k-1}(s_i^1))} & \dots & \dots & r_i^m \\ \vdots & \rightarrow & \dots & \dots & \dots & \dots & \dots & \vdots \\ s_n^1 & \rightarrow & r_n^1 & r_n^2 & \dots & \dots & r_n^{m-1} & r_n^m \end{array}$$

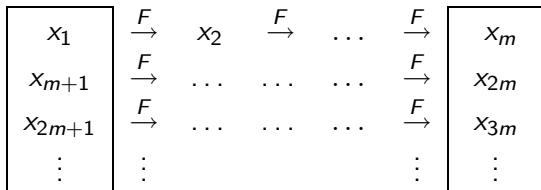
The complexity in terms of hash operation is $2mn$.

Example

In a library, we consider 2^{20} tagged books. We assume that the system can carry out 2^{23} operations per second. Identifying one tag requires 32 seconds.

Hellman's Time-Memory Trade-Off

- ▶ Exhaustive search on $F: X \rightarrow X$ in order to find preimages.



- ▶ Given one output x_i of F that we want to invert, we generate a chain starting at x_i : $x_i \xrightarrow{F} x_{i+1} \xrightarrow{F} x_{i+2} \xrightarrow{F} \dots$
- ▶ We can then regenerate the complete chain and find x_{i-1} .
- ▶ Complexity $T \propto N^2/M^2$.
- ▶ In practice $F: X \rightarrow Y$.

Defining the Function to Invert

- ▷ Difference between system and adversary.
- ▷ We choose F as

$$F : (i, k) \mapsto r_i^k = G(H^{k-1}(s_i^1)) \quad (1 \leq i \leq n, 1 \leq k \leq m).$$

- ▷ F is more complex: i and k are arbitrary results from R and we need $m/2 + 1$ hash operations to compute $F(i, k)$.
- ▷ Brute force requires $n|s|$ memory to store the n initial values s_i^1 to compute F .
- ▷ c is the ratio between the memory used by the trade-off and the memory used by the brute-force.
- ▷ Conversion factor $\mu = |s|/(2|n| + 2|m|)$.

$$T \approx \frac{N^2}{M^2} \gamma \approx \frac{n^2 m^2}{(c-1)^2 \mu^2 n^2} \left(\frac{m-1}{2} + 1 \right) \gamma \approx \frac{m^3 \gamma}{2(c-1)^2 \mu^2}.$$

- ▶ We can optimize by storing **intermediate values**, sacrificing so memory but reducing the average complexity of F .

$$T \approx \frac{n^2 m^2}{(c-x)^2 \mu^2 n^2} \left(\frac{m}{2x} + 1 \right) \gamma.$$

- ▶ The optimal complexity is achieved when $x = \frac{c}{3}$.

$$T_{\text{optimal}} \approx \left(\frac{3m}{2c} \right)^3 \frac{\gamma}{\mu^2}.$$

Numerical Example

Example

In a library, we consider 2^{20} tagged books. The length of the hash chains is 2^7 . We assume that the system can carry out 2^{23} hash operations per second. The system has 1.25 GB RAM. Identifying one tag requires 0.002 milliseconds and identifying the whole system requires about 2 seconds. Precomputations require 17 minutes.

Comparison

Scheme (parameter)	Time (millisecond)
CR/MW ($\delta = 2^{10}$)	0.122
CR/MW ($\delta = 2$)	0.002
OSK/AO (342 MB)	0.122
OSK/AO (1.25 GB)	0.002

Conclusion

Publications

- ▷ Fair Exchange
AV03a, AV03b, AV04, AGGV05, AGGV, Avo03.
- ▷ Radio Frequency Identification
Avo04, ADO05, AO05a, AO05b, AC, Avo, AB06.
- ▷ Odds and Ends
Avo05, AMP04, AJO05, AJO, AJ03, VAJ03, AJO05.
- ▷ RFID lounge and mailing list
<http://lasecwww.epfl.ch/~gavoine/rfid/> (270 persons).

Further Research

- ▶ Formalism of the adversary model in RFID.
- ▶ Reducing identification complexity.
- ▶ Distance bounding protocols based on symmetric cryptography.