TÜBİTAK

Deploying OSK on Low-resource Mobile Devices

Gildas Avoine – Muhammed Ali Bingöl
Xavier Carpent – Süleyman Kardaş

RFIDsec 2013, Graz
July 10, 2013

RFIDsec 2013

Authors

- Gildas Avoine

Université catholique de Louvain, Belgium

- Muhammed Ali Bingöl

TUBİTAK BİLGEM, Turkey

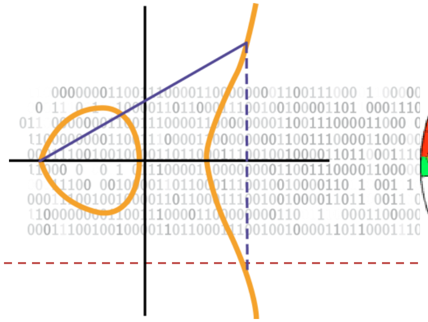
- Xavier Carpent

Université catholique de Louvain, Belgium

- Süleyman Kardaş

TUBİTAK BİLGEM, Turkey

TUBITAK National Research Institute of Electronics & Cryptology



Outline

TÜBİTAK

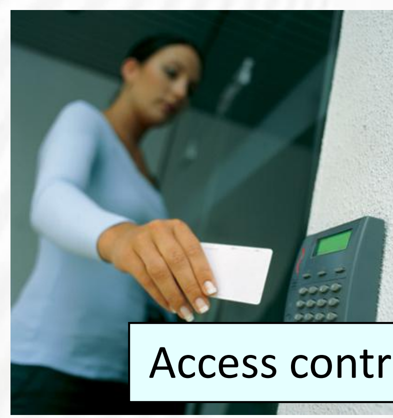
- Motivation
- Forward privacy
- OSK
- TMTO & OSK/AO
- Algorithms & Experiments
- Conclusion

Some RFID Applications

Passports

ID Cards

Public transportation


Access control

Toll Pay

Mass user authentication

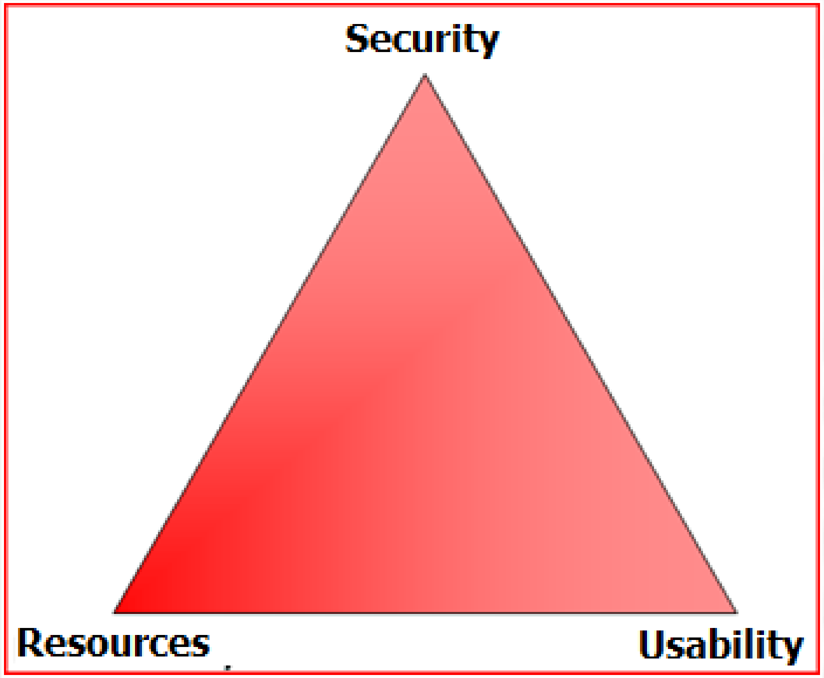
Montreal

- Montreal Metro system has transported over 7 billion passengers as of 2010, roughly equivalent to the world's population.
- Montreal Metro system has 1,241,000 daily passengers.
- In Istanbul, 6,5 million people have RFID card for public transportation. About 1 million of them have registered RFID card with private information.

Requirements

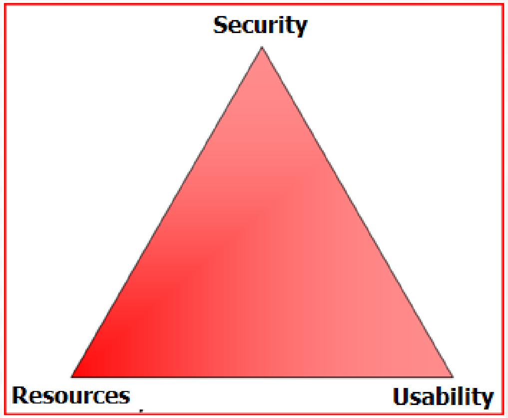
- 200 milliseconds can be dedicated to grant or deny the access to a customer in a flow.
- Some applications require mobile authentication mechanism.

Security vs Resources vs Usability

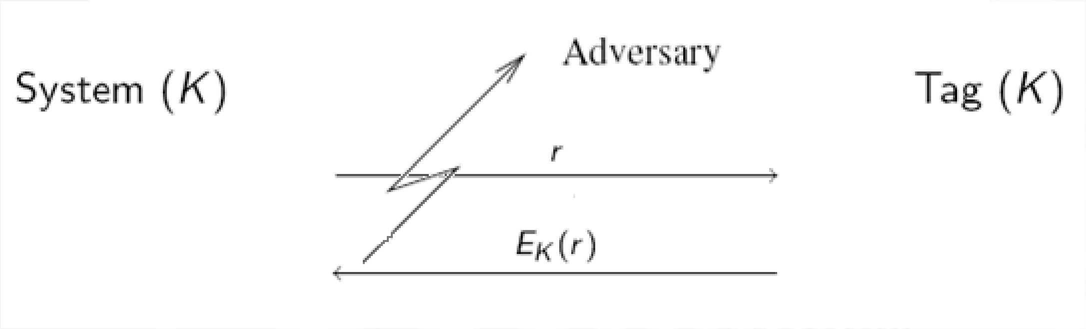


What we aim & What we have?

- Security
  - Authentication
  - User Privacy
  - Forward Privacy
    - Tags are not tamper resistant
- Usability
  - Fast authentication time
    - Less than 200 ms
  - Operating device
    - Handheld devices
    - Low power consumption
- Resources
  - Low computation ability
    - 200.000 hashes/sec $\cong 2^{17.5}$ sec
  - Low User memory
    - Up to 256 MB (RAM)
  - Symmetric Crypto



What is a private protocol ?



- **Need:** Design an RFID protocol that allows only authorized system to identify or authenticate a tag.

  An adversary is neither able to identify it nor trace it.

➢ Information needs to be randomized for each interaction.

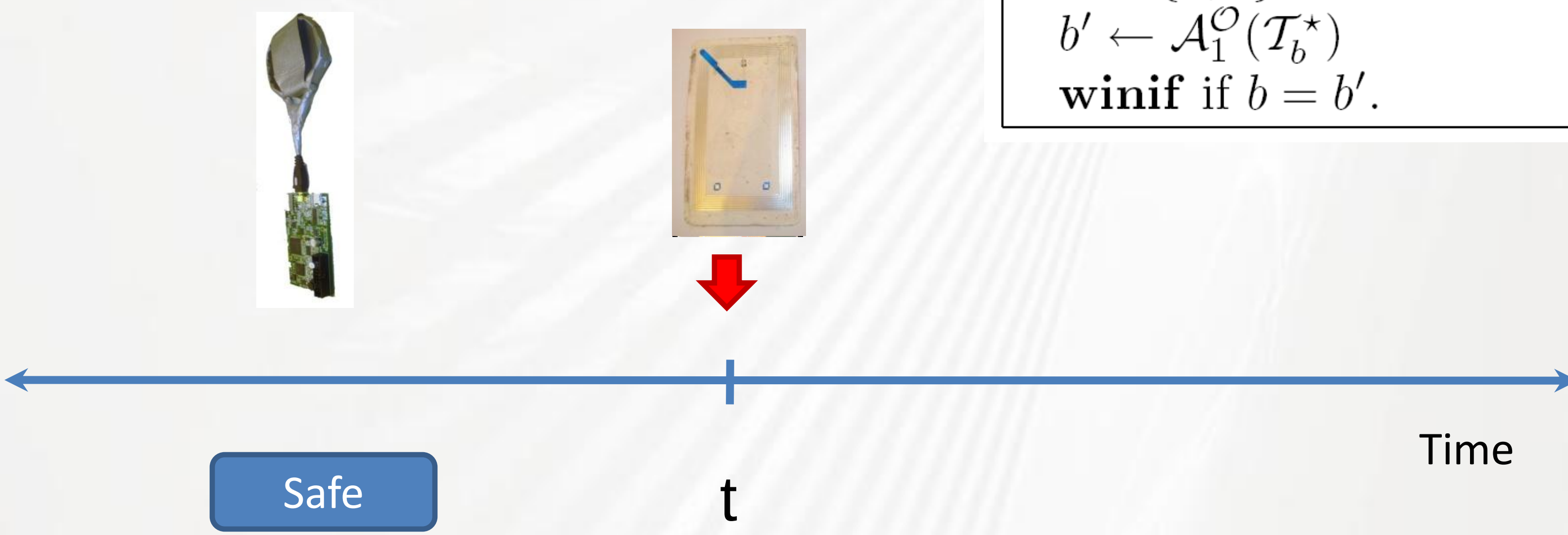
Privacy Definitions

- **Privacy:** Given a set of readings between tags and readers, an adversary must not be able to find any relation between any readings of a same tag or set of tags.
- **Forward Privacy**: Given a set of readings between tags readers and given the fact that *all information* stored in the involved tags has been *revealed* at time t, the adversary must not be able to find any relation between any readings of a same tag or set of tags that occurred at a time $t' \leq t$.

RFID Privacy Model

Forward Privacy

$$
\begin{array}{l}
\textbf{Priv-Game}_{\Pi,\mathcal{A}}(\eta) \; : \\
\quad (\mathrm{sk}, \mathrm{pk}) \leftarrow \mathsf{SetupSystem}(1^{\eta}) \\
\quad (\mathcal{T}_0^{\star}, \mathcal{T}_1^{\star}) \leftarrow \mathcal{A}_0^{\mathcal{O}}(\mathrm{pk}) \\
\quad b \leftarrow \{0, 1\} \\
\quad b' \leftarrow \mathcal{A}_1^{\mathcal{O}}(\mathcal{T}_b^{\star}) \\
\quad \textbf{winif} \text{ if } b = b'.
\end{array}
$$



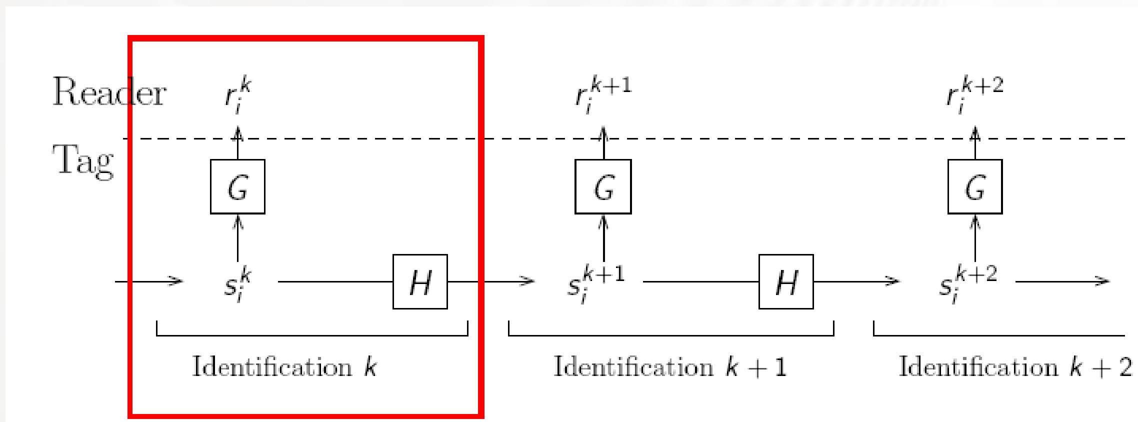
A forward private protocol - OSK

Ohkubo-Suzuki-Kinoshita (2003 – RFID Privacy Workshop - MIT)

- Each tag needs 2 hash functions G and H (in theory).
- Each tag needs an EEPROM capable of storing an identifier.
- The personalisation of a tag T_i consists in storing in its memory a random identifier s_i^1, which is also recorded by the database of the system.
- Thus, the database initially contains the set $\{s_i^1 \mid 1 \leq i \leq n\}$.

OSK Protocol

$\mathcal{R}$		$\mathcal{T}_i$
	$\xrightarrow{request}$	
find s_i^0 so that $G(H^k(s_i^0)) = \sigma$	$\xleftarrow{\sigma=G(s_i^k)}$	$s_i^{k+1} \leftarrow H(s_i^k)$



Back-end Database

s_1^0	$\longrightarrow$	r_1^0	r_1^1	r_1^2	$\dots$	r_1^{m-1}	r_1^m
$\dots$	$\longrightarrow$	$\dots$	$\dots$	$\dots$	$\dots$	$\dots$	$\dots$
s_j^0	$\longrightarrow$	$\dots$	$\dots$	$\dots$	$r_i^k = G(H^k(s_j^0))$	$\dots$	r_j^m
$\dots$	$\longrightarrow$	$\dots$	$\dots$	$\dots$	$\dots$	$\dots$	$\dots$
s_n^0	$\longrightarrow$	r_n^0	r_n^1	r_n^2	$\dots$	r_n^{m-1}	r_n^m

How to identify a tag !

- Online Computation
- Full Storage
- Time-Memory Trade-off (TMTO)

Online Computation

- Example:

  Number of tags : 2^{20}

  Life time of the tags: 2^7

  $N = 2^{27}$

Computation capability of server (hashes/sec)	Avg Authentication Time (sec)
2^{22}	$16\ sec$
2^{20}	$64\ sec$
$2^{17.5}$	$\cong 360\ sec$

Full Storage

- Example:

  Number of tags : 2^{20}

  Life time of the tags: 2^{7}

  Response size: 128 bits

  Total = 2^{34} bits $= 2$ GB

<u>Our limitations:</u>

Authentication time $\leq$ 200 ms
User Memory $\leq$ 256 MB RAM

Time-Memory Trade-Off (TMTO) method

TÜBİTAK

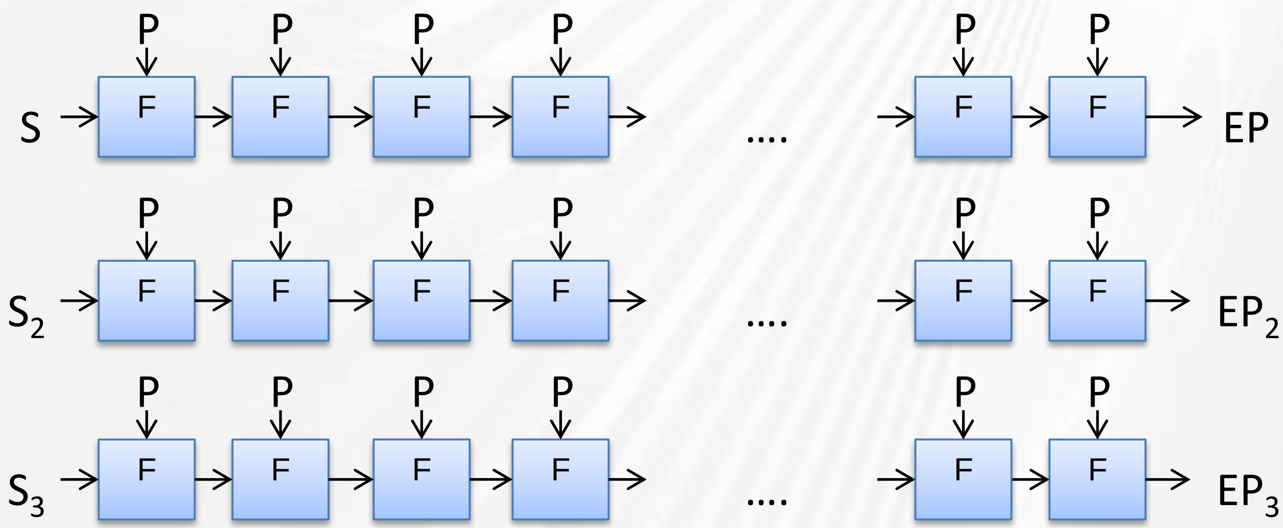
- The basic idea of the **TMTO** method is to find a trade-off between the exhaustive search and the exhaustive storage (table look-up).
- In TMTO method a pre-computation table is constructed only once.
- Only the first and the last elements of each chain are stored and sorted according to the last elements.

TMTO (Hellman's Tables)

Usually used for inverting one-way functions.

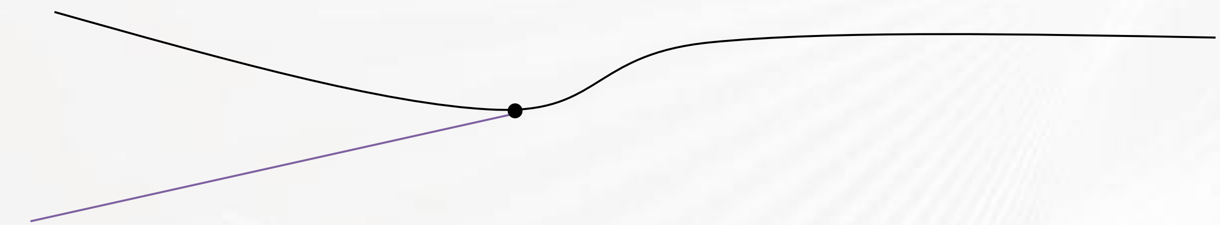
If C has more bits than the key, then a reduction has to be performed before the next encryption

1. Choose a starting point, S
2. Choose a plaintext, P
3. C = F(P,S)
   - The result becomes the key for the next encryption in the chain
4. Repeat until endpoint, EP, reached
5. Go back to step 1

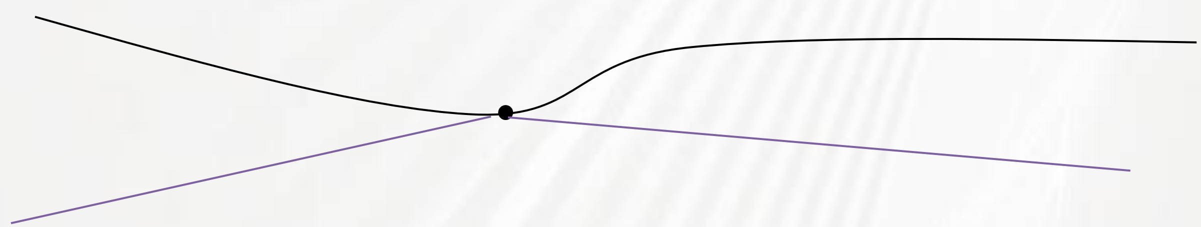


Using Rainbow tables

Using same R functions



Using different R functions



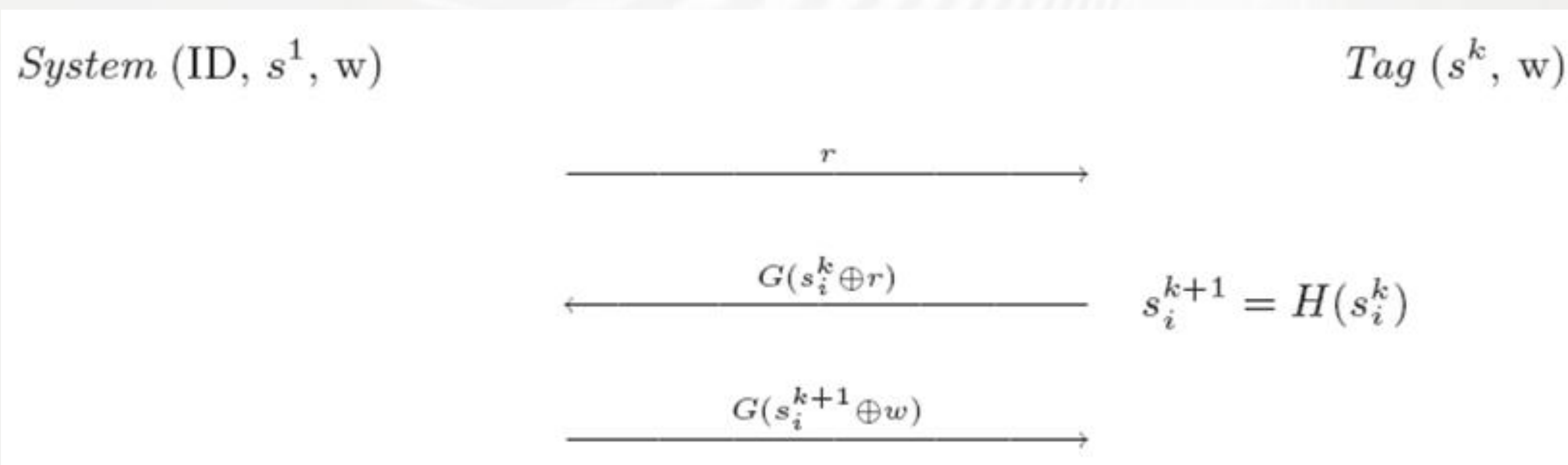
Because R is different in each chain, they diverge again

OSK with Time Memory Trade-offs

(OSK/AO) 2005

[1] Gildas Avoine and Philippe Oechslin, A Scalable and Provably Secure Hash Based RFID Protocol, PerSec 2005.

[2] Gildas Avoine, Etienne Dysli, and Philippe Oechslin, Reducing Time Complexity in RFID Systems. SAC 2005



Functions for OSK/AO

$$\mathcal{F}: (i, j) \longmapsto \mathcal{G}\left(H^{j}(S_i^0)\right) = r_i^j$$

$$\mathcal{R}: r_i^j \longmapsto (i', j')$$

where $1 \le i, i' \le n$ and $0 \le j, j' \le L$

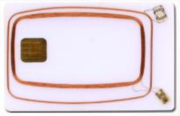
Rainbow Table Generation

$$m_t \left\{ \begin{array}{l} (i,j) \xrightarrow{F} r_i^j \xrightarrow{R_1^v} (i',j') \xrightarrow{F} r_{i'}^{j'} \xrightarrow{R_2^v} (i'',j'') \ldots \xrightarrow{R_t^v} (i^{(')},j^{(')}) \\ \vdots \qquad\qquad\qquad \vdots \qquad\qquad\qquad \vdots \end{array} \right.$$

$$m_t \left\{ \begin{array}{l} (i,j) \xrightarrow{F} r_i^j \xrightarrow{R_1^v} (i',j') \xrightarrow{F} r_{i'}^{j'} \xrightarrow{R_2^v} (i'',j'') \ldots \xrightarrow{R_t^v} (i^{('')},j^{('')}) \\ \vdots \qquad\qquad\qquad \vdots \qquad\qquad\qquad \vdots \end{array} \right.$$

Experiment Devices

TÜBİTAK

Table Constructer	Reader	RFID Tag
		
Processor: 2.8 GHz RAM: 4 GB Windows 7 – 64 bit Prog Lang: Java	LG Optimus 4X P880 Android 4.1 NFC enabled phone Processor: 1.5 GHz	Basic card ZC 7.5 EEPROM: 32 kB RAM: 2.9 kB

25

Steps

Upload the Tables
into the NFC phone

Construct the Tables

1) Rapid hash table
2) TMTO tables

Initial seeds

Tag Identification

Construction of Tables

TÜBİTAK

Algorithm 2 Construction of $Table_v\ (j,\ m_1,\ v)$

Require: $1 \leq j,\ 1 \leq m_1 \leq n \times j\ ,\ v \geq 1$

```
table ← {∅}
for i = 1 to ⌈m1/j⌉ do
    for k = 0 to j do
        nextResp ← F(i, k)
        for w = 1 to t − 1 do
            z[ ] ← R_w^v(nextResp)
            nextResp = F(z[0], z[1])
        end for
        z[ ] ← R_t^v(nextResp)
        if z ∉ table then
            add the record {(i, k); (z[0], z[1])} into table
        end if
        if (i − 1) × j + k ≥ m1 then
            break
        end if
    end for
end for
clean table
return table
```

Identification

TÜBİTAK

Algorithm 3 Identify ($Table_v$, TagResp)

Require: TagResp $\in \{0,1\}^\lambda$, $v \geq 1$
Ensure: TagResp $\leftarrow \mathcal{G}(y)$

```
for q = t down to 1 do
    nextResp ← TagResp
    for i = q to t − 1 do
        z[ ] ← R_i^v(nextResp)
        nextResp ← F(z[0], z[1])
    end for
    z[ ] ← R_t^v(nextResp)
    if z ∈ Table_v then
        {z'; z} ← Table_v(z)
        nextResp ← F(z'[0], z'[1])
        for w = 1 to q − 1 do
            z̃[ ] ← R_w^v(nextResp)
            nextResp ← F(z̃[0], z̃[1])
        end for
        if nextResp = TagResp then
            return true
        end if
    end if
end for
return false
```

Our Reduction Function

Algorithm 4 Compute $\mathcal{R}_w^v(val[.])$

Require: $v \geq 0$, $w \geq 1$

Ensure: $i \in \mathbb{Z}_n$, $j \in \mathbb{Z}_L$

$i \leftarrow Int32(val[v, v+3]) + w$

$j \leftarrow Int32(val[v+1, v+4]) + w$

$i = i \ mod \ n$

$j = j \ mod \ L$

return $\{i, j\}$

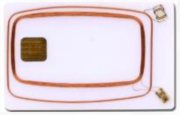
Our one-way functions

- $\mathcal{H}(S_i^j) : AES_K(S_i^j) \oplus S_i^j = S_i^{j+1}$,
- $\mathcal{G}(S_i^j) : AES_K(S_i^j + 1) \oplus (S_i^j + 1) = r_i^j$

Matyas- Meyer-Oseas construction

Tables generation
takes 1 hour
(including all processes)

187,750 hash/sec
256 MB for user memory



Hash calc: 25 ms
Comm time: 20 ms
Total : 70 ms in avg

31

Experiment Results on NFC Phone

SETTING	I	II
Memory	253 MB	113 MB
Identification time on phone	15.26 ms	117.54 ms
Total authentication time	< 100 ms	< 200 ms

Length of the chains of the TMTO (t)	27	72
Number of chains of the TMTO (m_t)	8,968,214	3,566,605
Rapid-hash parameter ($\varkappa$)	22	43
Number of Rainbow tables	4	4
Authentication rate	99.9%	99.9%

Each experiment is run 1,000,000 times

Conclusion

- We have implemented a forward private protocol on
  - NFC-compliant android cellphone
  - ZC7.5 contactless tag
- The implementation is suited to
  - large-scale applications
  - Low-resource devices
- Memory consumption

  < 256 MB
- Average identification time

  < 200 ms

THANK YOU !!

QUESTIONS ?

muhammedali.bingol@tubitak.gov.tr

Supplementary Page

Protocol Comparisons

TÜBİTAK

Class	Shared secrets			Hash-chains					Counter-based		
Protocol	CHT plain	CHT with auth	CTI	OSK	OSK/AO with Auth	OSK/BF with Auth	O-RAP	O-FRAP	RIP+	YA-TRAP*	YA-TRAP* &fwd
Main reference	[26]	[26]	[2]	[54]	[5], [8], [10]	[52]	[18]	[68]	[66]	[66]	[66]
Year of publication	2009	2009	2010	2003	2005	2008	2006	2007	2007	2007	2007
Identification/ Authentication	auth.[A]	auth.	auth.	id.	auth.	auth.	auth.	auth.	auth.	auth.	auth.
Off-line Computation Complexity[B]	0	0	$O(NC)$[C]	$2N$[C]	$\frac{NM^2}{2}$ ([8])[D]	$2NM$[C]	0	0	N/counter update[E]	N/counter update[E]+ Lamport Chain	N/counter and key update[E] + Lamport Chain
Normal case online complexity	$O(\sqrt{N})$	$O(N^a)$	4	2	$O(N^{2/3})$[F]	$M(\epsilon N+3)$ on average	1	2	0[E]+1[G]	0[E]+1[G]	0[E]+1[G]
Desynchronized case online complexity	N/A	N/A	N/A	lower than $2N(M-1)$[H]	$O(N^{2/3})$[F]	$M(\epsilon N+3)$ on average	$O(N)$	$O(N)$	out of order after desync.[H]	out of order after desync.[I]	out of order after desync.[I]
Memory Complexity	$2\sqrt{N}$	$2N^a+N$	$O(N)$[J]	N	$O(N^{2/3})$[F]	$\frac{NM\log\epsilon}{-\log^2 2}$	$2N$	$3N$	N[E]	N[E]	N[E]
Tag Computation	2 PRFs + 1 Nonce	3 PRFs + 1 Nonce	5 hashes	2 hashes	3 hashes	3 hashes	2 hashes	4 hashes	2 hashes + 1 Nonce	$\nu+2$ hashes + 1 Nonce	$2\nu+2$ hashes + 1 Nonce
Tag Resources	PRF, PRNG	PRF, PRNG	PRNG, Hash func.	Hash func.	Hash func.	Hash func.	Hash func.	Hash func.	PRNG, Hash func.	PRNG, Hash func.	PRNG, Hash func.
Privacy	no	no★	no★	yes[K]	yes◇,[K]	no★	no★,[K]	no★,[K]	not private after desync.	not private after desync.★,[L]	not private after desync.★,[L]
Forward-privacy	no	no	no[M]	yes[K]	yes[K]	no[M]	no	no★	no	no	no[N]
Desynchronization resistance	N/A	N/A	yes	yes up to M consecutive[O]	yes up to M consecutive[P]	yes up to M consecutive[O]	no[Q]	no[Q]	no[R]	yes[S]	yes[S]
Impersonation Resistance	no★	no	yes	N/A	yes	yes	yes	yes	yes	yes	yes

Gildas Avoine, Muhammed Ali Bingöl, Xavier Carpent, Siddika Berna Ors Yalçin, "*Privacy-friendly Authentication in RFID Systems: On Sub-linear Protocols based on Symmetric-key Cryptography*" accepted from **IEEE Transactions on Mobile Computing (TMC)**.