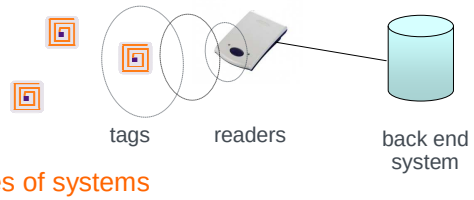
Lightweight Privacy Preserving Authentication for RFID Using a Stream Cipher

Olivier Billet, Jonathan Etrog, and Henri Gilbert
Orange Labs

RFID systems



- many types of systems
  - power supply, range, tag memory/processing capabilities...
  - cost per tag: from a few cents to more than 1$
- many types of applications
  - management of the supply chain
  - ticketing, public transportations
  - access control, automatic tolls
  - anticounterfeiting
  - pets tracking
  - airline luggage tracking

  ....
- quite diverse security and privacy needs
  - a challenging task for lightweight cryptography

security (1)

- main security need: prevent tag cloning and impersonation…
  - this is required in many applications
    - ticketing
    - anticounterfeiting, etc.

- by means of a tag authentication protocol
  - it allows the reader:
    1. to get the tag identity
    2. to corroborate this identity
  - sometimes combined with reader authentication: mutual authentication
  - limited H/W resources: typically less than 3000 GE for auth. in low cost tags

security (2)

much recent research has been focused on lightweight authentication

- emerging authentication solutions based on:
  - a dedicated block cipher
    - e.g. DESXL, PRESENT, KATAN
  - a lightweight stream cipher
    - e.g. GRAINv1 or TRIVIUM from the eSTREAM stream cipher porfolio
  - a symmetric authentication scheme without underlying cipher
    - e.g. SQUASH [Shamir 08], HB^+ family [Juels-Weis 05]
    - promising direction, but not yet fully mature
  - an asymmetric authentication scheme
    - CryptoGPS

privacy of RFID protocols (1)

- main concern
  - by carrying an RFID tag, a person might render her moves and actions traceable by a malicious party equipped with a fake reader
- countermeasure: privacy preserving protocols
  - privacy needs only ⇒ privacy preserving identification
    - library management
    - pets tracking
  - privacy and security needs ⇒ privacy preserving (mutual) authentication
    - ticketing
    - automatic tolls

    …

privacy of RFID protocols (2)

- two main types of privacy preserving protocols
  - (weakly) private: anonymity and unlinkability of the exchanges of any tag for an attacker equipped with a fake reader
  - forward private: after tampering with some tags and reading their internal state, the former attacker must not be able to link this information to their past exchanges [Ohkubo et al., 05]
- the design of private RFID protocols is not yet a fully mature area
  - much progress toward a rigorous definition of privacy models

    [Juels 04, Avoine 05, Juels-Weis 06, van Le et al. 07, Vaudenay 07, Paise-Vaudenay 08…]
  - designing an efficient forward private protocol remains a challenging problem

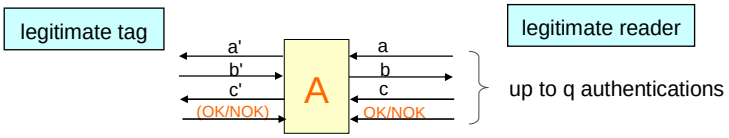
outline

1. adversary models for RFID protocols
   - security, forward privacy, and correctness notions
2. most efficient forward private protocols proposed so far
   - the OSK protocol and its variants
   - the PFP protocol
3. a new forward private RFID authentication protocol
   - PEPS: Private and Efficient Protocol based on a Stream cipher
4. security and privacy proofs for PEPS (outline)
   - in the standard model

security adversary against a (mutual) authentication protocol

- attack model: we consider a powerful active adversary A able to mount a man in the middle impersonation attack (MIM model).

  - phase 1: A observes/ disturbs up to q (mutual) authentication exchanges and is given the reader (and tag) authentication outcome (OK/NOK)



  - phase 2: A interacts once with the reader (or the tag) and tries to impersonate the tag (or the reader) → success or failure

- definition: a (mutual) authentication protocol is (q,T,ε)-secure iff for any security adversary A running in time at most T, Pr(A succeeds) ≤ ε .

forward privacy adversary

- attack model: after tampering with a legitimate tag and reading its internal state, adversary A tries to recognise its past protocol executions

  - phase 1: A observes and disturbs up to q protocol executions of two tags: tag_0 and tag_1 and accesses the OK/NOK outcome

  - phase 2: A accesses up to q protocol executions of tag b, $b \in_R \{0,1\}$, reads its internal state, and tries to guess the value of b
    → A outputs a guess b'

- definition: the protocol is (q,T,ε)-forward private iff for any adversary A running in time at most T, $| \Pr(b'=b) - 1/2 | \leq \varepsilon$.

correctness

- purpose: undisturbed (mutual) authentication attempts of legitimate tags to a legitimate reader must succeed with overwhelming probability

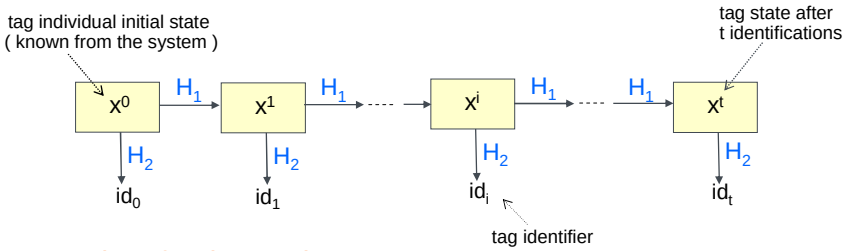
  - even after q interactions of malicious adversary A with the system.
    this allows to capture resistance to denial of service attacks (DoS)

- definition: an RFID protocol is (q,T,ε)-correct iff the probability that, after q interactions with a DoS adversary of running time at most T, an undisturbed authentication of a legitimate tag fails is at most ε.

forward private identification protocol OSK

[Ohkubo-Suzuki-Kinoshita 03]

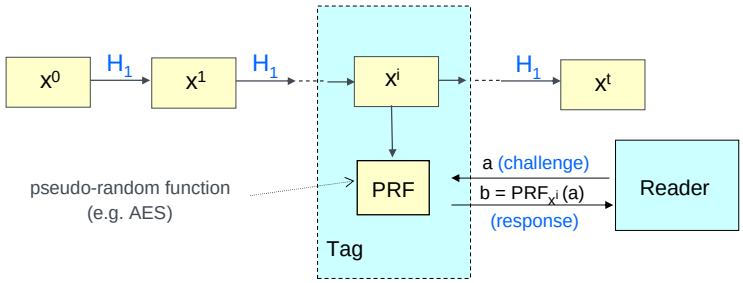
- uses two distinct one-way hash functions H_1 and H_2



- procedure for the reader
  - for each of the N active tags compute sliding "hash chain" $[id_{j+1}, \ldots, id_{j+\omega}]$ of length ω

    the received identifier id_i is searched in the N hash chains
    - "naive" approach → $O(N\omega)$ operations per identification
    - time/memory trade-off → faster search [Avoine-Oechslin 05]
- properties
  - forward private (up to some limitations)
  - not secure (replay of identities collected by a false reader)

OSK variant: authentication protocol [Avoine et al. 05]

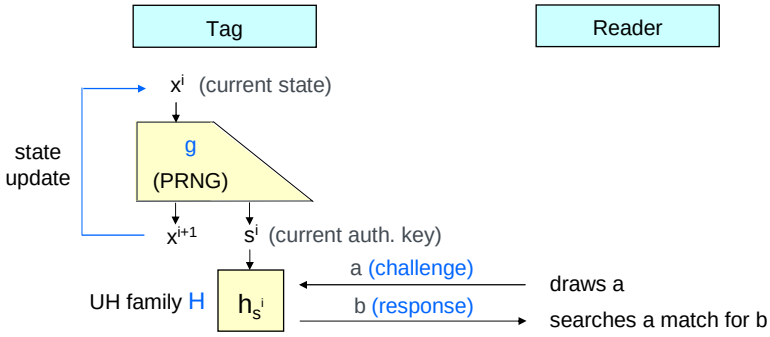


- hash chain based verification as in OSK

  (a slight modification of the protocol allows a faster verification)
- forward private up to the same limitations as OSK
- provably secure in the random oracle model

the PFP authentication protocol [Berbain et al. 09]

- uses two lightweight cryptographic ingredients:
  - a PRNG g, a universal family of hash functions H



- procedure for the reader similar to [Avoine et al. 05]
  - for each tag use sliding key chain $[s^{i+1}, s^{i+2}, \ldots, s^{i+\omega}]$ of length ω

limitations of OSK, its variants, and PFP

- if ω is too small, OSK, these protocols are vulnerable to DoS attacks [JW06]
  - very long key chains needed to thwart DoS ($\omega \approx 2^{20}$) $\rightarrow$ perf. penalty in reader
  - moreover these DoS attacks compromise the forward privacy
- it can be shown that in a symmetric setting it is impossible to reconcile
  - full forward privacy
  - full resistance to DoS attacks by an adversary with infinite desynchronisation capability

scheme	forward privacy	DoS resistance	complexity (reader)	complexity (tag)	provable security
OSK family	😐 up to DoS	😐 $< \omega$	☹ if N, ω large	☹ $\approx$ 6000 GE	😐 RO model
PFP	😐 up to DoS	😐 $< \omega$	☹ if N, ω large	☺ $\approx$ 3500 GE	☺ std. model

objective of PEPS

- aim

  find a more realistic balance than the former protocols between:
  - forward privacy
  - resistance to DoS
  - computational efficiency in the tag and the reader

  in order to get a fully practical protocol

- approach
  - use one single cryptographic ingredient: a lightweight stream cipher
  - slightly relaxed forward privacy requirement: almost forward privacy
    - this allows to achieve full resistance to DoS attacks
    - a similar notion was considered in [van Le et al. 07]

almost forward privacy

- adversary model

  same two-phase adversary model as for forward privacy

  - phase 1: A observes and disturbs up to q protocol executions of two tags: tag_0 and $tag_1$and accesses the OK/NOK outcome
  - phase 2: A accesses up to q extra protocol executions for tag b, $b \in_R \{0,1\}$, reads its internal state, and tries to guess the value of b

    → A outputs a guess b'

  except we now assume that at least one undisturbed authentication of tag_0 and tag_1 takes place between phase 1 and phase 2

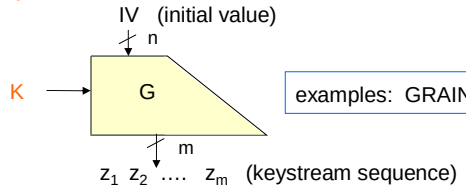
  😐 slight relaxation of the unlinkability requirements: this seems acceptable in practice

  ☺ removes the need to update the tag state after failed authentications

- definition: a protocol is (q,T,ε)-almost forward private iff for any f.p. adversary A running in time at most T, $| \Pr(b'=b) - 1/2 | \leq \varepsilon$.

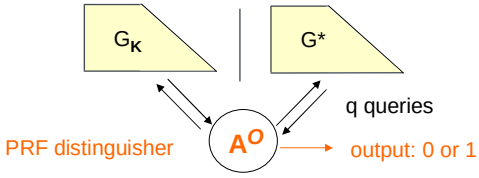
PEPS' single cryptographic ingredient

- stream cipher



examples: GRAIN v1 (~1300 GE) or TRIVIUM

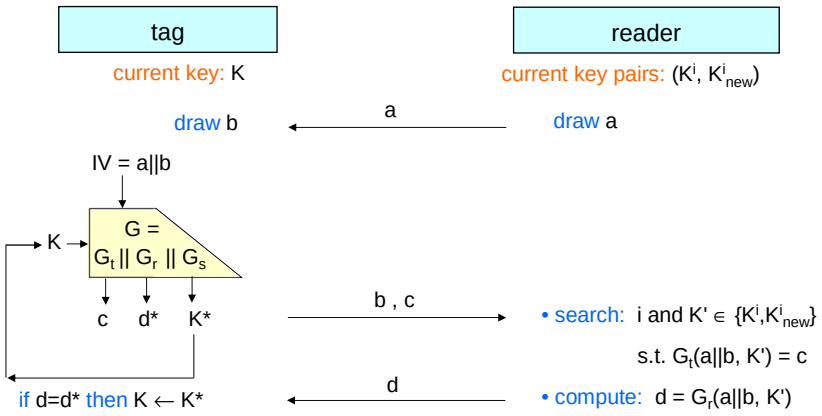
- security of a stream cipher: $G = \{G_K\}$ must be a PRF [BG07]



$Adv^{PRF}_{G}(t,q)$ must be "negligible" for sufficiently large values of q and t

(17 / 22)

privacy preserving mutual auth. scheme PEPS



- easy bit by bit processing in the tags: only K* requires some extra memory
- very compact: ≈ 1700 GE if Grain-v1 is used (rough estimate)

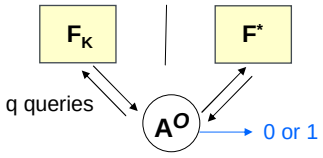
(18 / 22)

proofs: prerequisite

issue: streamcipher outputs are reused as keys in the key chains

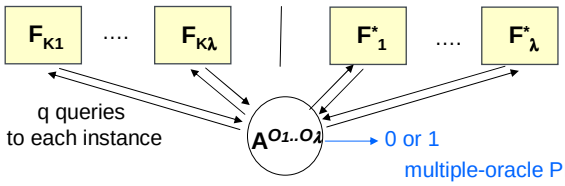
⇒ testing experiments involve multiple keys instead of one

- usual PRF distinguishers (one single oracle) are inadequate



PRF advantage: $Adv_F^{PRF}(q,T)$

- multiple-oracle PRF distinguisher notion required here



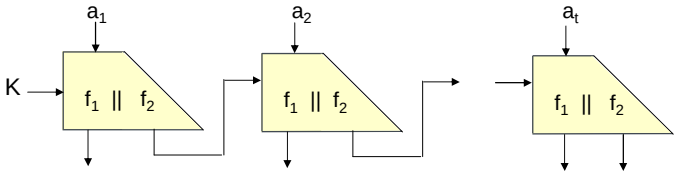
multiple-oracle PRF advantage: $Adv_F^{PRF}(\lambda,q,T)$

- link beween both notions: $Adv_F^{PRF}(\lambda,q,T-T_F) \le \lambda Adv_F^{PRF}(q,T)$

(19 / 22)

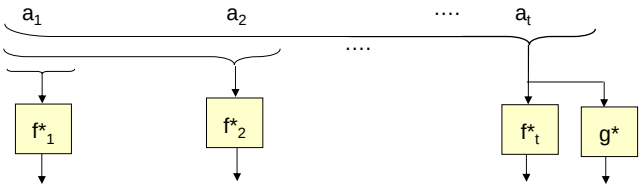
security proofs: core theorem

- function associated with a key chain (the ai represent successive IV values)



F_t

- vs ideal function (same nested inputs structure)



F^*_t

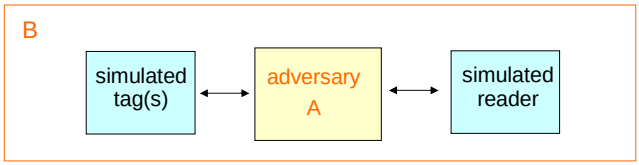
- theorem:

if f is a (q,T,ε)-PRF then F_t is $(q,T-(t-1)T_F,((t-1)q+1)\ \varepsilon)$-indistiguishable from F^*_t

(20 / 22)

security, almost f.p., and correctness theorems

- Th. (security): if G is a (q,T,ε)-secure PRF then PEPS is $(q\text{-}1,T\text{-}q(q\text{-}1)T_g,\varepsilon_1)$-secure with $\varepsilon_1=(q(q\text{-}1)+1)\varepsilon+(q\text{-}1)2^{-n/2}+2^{-l}$
- Th. (almost-fp): if G is a $(2q+1,T,\varepsilon)$-secure PRF then PEPS is $(q,T\text{-}q(2q+1)T_g,\varepsilon_2)$-almost forward private with $\varepsilon_2=2(q(q+1)+1)\varepsilon+q2^{1-n/2}+2^{1-l}+(2q+1)((2q(2q+1)+1)+((2q+1)(q-1)+1))\varepsilon$.
- Th. (correctness): if G is a (q,T,ε)-secure PRF with $T \geq NT_c+(1+2N)T_g$ then PEPS is $(q\text{-}1,T\text{-}q(q\text{-}1+N)T_g,\varepsilon_3)$-correct with $\varepsilon_3=N(q(q\text{-}1)+1)\varepsilon+(q\text{-}1)2^{-n/2}+N2^{-l}+N(N\text{-}1)2^{-2l}+N(N\text{-}1)(N\text{-}2)2^{3-2l}+4N\varepsilon$.
- rough outline of the proof methodology
  if there exists a security (or almost f.p., or correctness) adversary A, it can be converted into a (multiple-oracle) distinguisher B between F_t and F_t^*.



conclusion

- main features of the PEPS scheme
  - truly practical performance profile (tag + reader)
  - strong security and privacy properties, provable in the standard model

scheme	forward privacy	DoS resistance	complexity (reader)	complexity (tag)	provable security
OSK family	😐 up to DoS	😐 < ω	☹ if N, ω large	☹ ≈ 6000 GE	😐 RO model
PFP	😐 up to DoS	😐 < ω	☹ if N, ω large	☺ ≈ 3500 GE	☺ std. model
PEPS	😐 almost-f.p.	☺ infinite	😐 O(N)	☺☺ ≈ 1700 GE	☺ std. model

  - can be derived from any stream cipher
    … or more generally from any input- expanding PRF