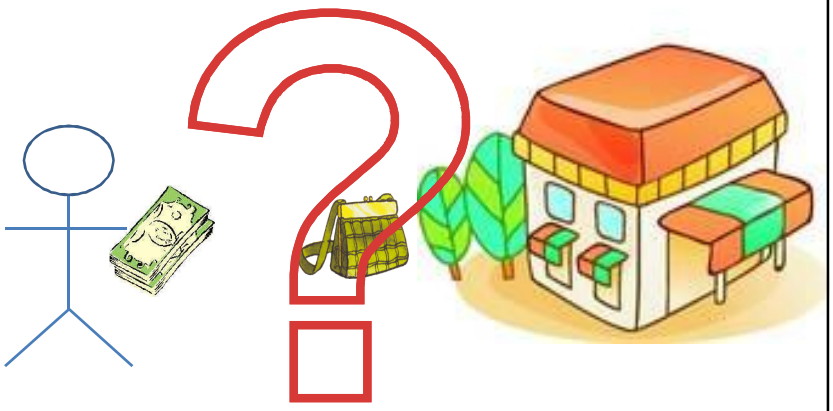
Lotta alla contraffazione tramite RFID: autenticazione forte di tag RFID basata su PUF*

Bernardo Palazzi
bernardo@bernardo.it
Lavoro svolto in collaborazione con:
Prof. Giuseppe Di Battista, Ing. Maurizio PIZZONIA

21 Ottobre 2009

* FIRB project "Advanced tracking system in intermodal freight transportation", Grant number RBIP06BZW8
PUF-based tags were kindly supplied by Verayo, Inc.

Acquistiamo un bene...



Vero o Falso

2

Il problema della Contraffazione

- produzione e commercializzazione di merci che recano - illecitamente - un marchio identico ad un marchio registrato;
- Alcuni esempi:
  - **merci**: pezzi di ricambio, componentistica (elettrica, elettronica e industriale), giocattoli, ecc.
  - **prodotti**: alimenti, farmaci, cosmetici, tabacchi lavorati, ecc.

3

Entità della contraffazione

La Commissione Europea e l'Organizzazione Mondiale delle Dogane attribuiscono al fenomeno della contraffazione il 7% della merce scambiate a livello mondiale, un valore tra i 200 e i 300 miliardi di euro

in 10 anni il fatturato dell'industria del falso è aumentato del 1600%

Un'analisi della Commissione ha stimato che all'interno dell'Unione Europea, le merci contraffatte rappresentano:
- dal 5% al 10% delle vendite di pezzi di ricambio di autoveicoli
- il 10% delle vendite dei CD e di audiocassette
- il 16% delle vendite di film (videocassette e DVD)
- il 22% delle vendite di calzature e articoli d'abbigliamento
- il 15% delle vendite alimentari
- il 10% delle vendite di medicinali

(CONVEGNO MODA E INDUSTRIA DELLA CONTRAFFAZIONE
23 Gennaio 2007 Museo Nazionale di Capodimonte NA)

4

Che danni comporta?

- Il **produttore** soffre danni molto elevati dovuti a:
  - **perdite dirette:** mancata vendita di beni causata dalle differenze di prezzo tra prodotti originali e contraffatti
  - **perdite indirette:** quando il consumatore acquista un bene contraffatto credendolo originale
  - **svalutazione del marchio e dell'immagine:** significato simbolico di marchi famosi
  - **perdite a lungo termine:** i danni alla produttività risultanti dalle mancate vendite possono avere ripercussioni negative sul mercato del lavoro
  - **perdita degli investimenti in ricerca e sviluppo**

Il **produttore** inoltre non ha modo di avere un riscontro immediato dei beni contraffatti che vengono venduti anche tramite canali legali

6

Quali strumenti per difenderci?

- Il **rivenditore** non ha strumenti di verifica che non siano facilmente aggirabili da un **trasportatore** disonesto
- L'**acquirente** non ha strumenti di verifica che non siano facilmente aggirabili da un **rivenditore** disonesto

Il bene contraffatto può quindi essere venduto da un rivenditore più o meno ignaro a un acquirente finale che non ha alcuna possibilità immediata, semplice ed economica di verifica dell'autenticità

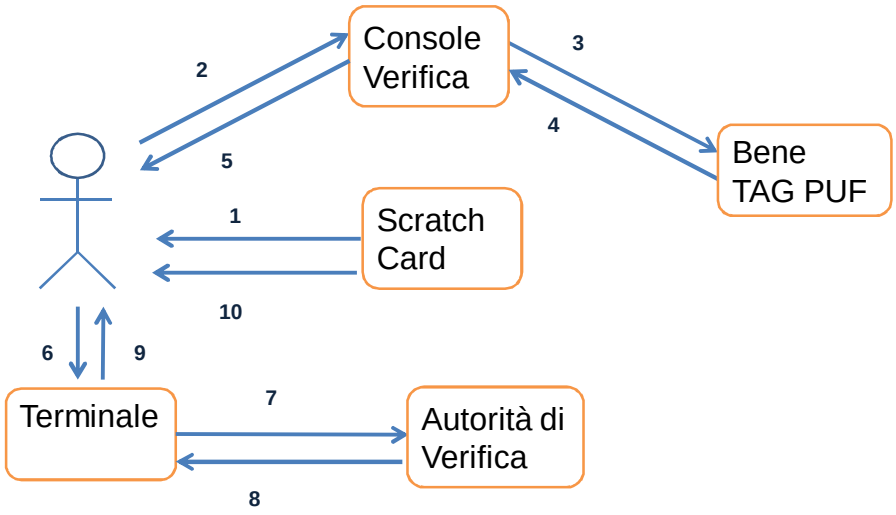
7

Nuovo Modello di Verifica

- Un nuovo modello per permettere all'utente finale di verificare l'autenticità del bene:
  - Non avendo alcuna fiducia:
    - Nel rivenditore e nella catena logistica per il trasporto del bene
    - Negli strumenti messi a disposizione dal produttore presso il punto vendita
    - Nella rete di comunicazione verso l'autorità di verifica (per es. il produttore)
  - Fidandosi solo di strumenti propri (per es. cellulare)

8

Soluzione Proposta



9

Procedura

1. l'utente ottiene un nuovo challenge da una scratch card, rimuovendo permanentemente la pellicola di protezione. Assieme al challenge possono essere opzionalmente fornite delle informazioni di verifica da usare nel punto 8.
2. l'utente inserisce il challenge sulla console di verifica
3. la console di verifica interroga la PUF del tag apposto sul bene da verificare
4. il tag fornisce la response al challenge fornito in accordo alla sua PUF caratteristica
5. la console di verifica mostra la response all'utente
6. l'utente prepara un messaggio di testo contenente dati relativi al numero seriale e al modello del bene da verificare, il challenge e il response ricevuto

10

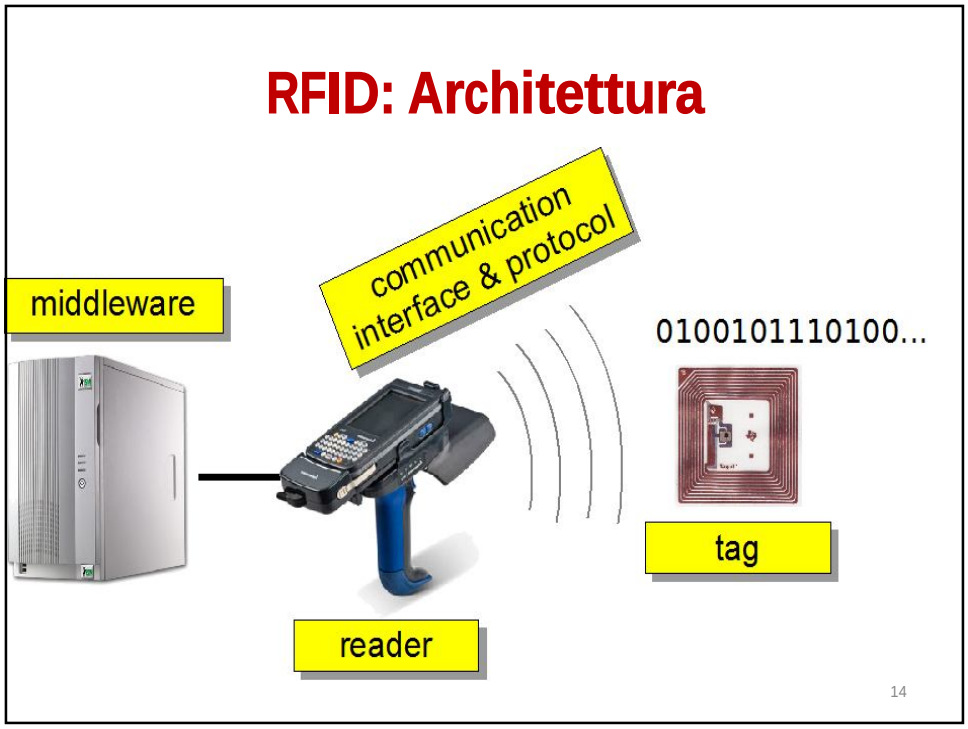
Procedura (2)

7. il terminale mobile invia all'autorità di verifica il messaggio
8. l'autorità di verifica invia al terminale mobile l'esito della verifica tramite messaggio di testo e opzionalmente delle informazioni di verifica associati al challenge ottenuto dalla scratchcard. L'esito è positivo se è la prima volta che si sottopone una certa CRP all'autorità di verifica e la response è conforme a quella che l'autorità conosce.
9. il terminale mostra al verificatore il messaggio con il codice segreto.
10. Il verificatore rimuove dalla scratch card la pellicola che ricopre i codici segreti rappresentante "verifica con successo" o "verifica fallita" e in base al confronto con il codice segreto ricevuto ottiene l'esito dell'autenticazione.

11

12

Attori	Interessi	Fidato
Autorità di Verifica		Tutti
Scratch Card	1. Combattere la contraffazione dei propri prodotti 2. Evitare errori in fase di distribuzione	Tutti
TAG PUF	1. Essere in grado di verificare l'integrità dei prodotti trasportati 2. Poter dimostrare l'avvenuta consegna	Se stesso
Console Verifica	1. Verificare l'autenticità dei prodotti 2. Evitare errori in fase di spedizione	Se stesso
Acquirente	1. Verificare l'autenticità dei prodotti	Se stesso



PUF: una primitiva leggera…

- PUF (Physically Unclonable Functions)
  - Facile da calcolare e difficile da caratterizzare
  - Alternativa a salvare chiavi su dispositivi hardware non sicuri

k ={ ritardi nei circuiti dovuti a variazioni nel processo produttivo}

$$\{c_1,c_2,c_3,\ldots,c_m\} \longrightarrow f(c_1,c_2,c_3,\ldots,c_m, k) \longrightarrow \{r\}$$

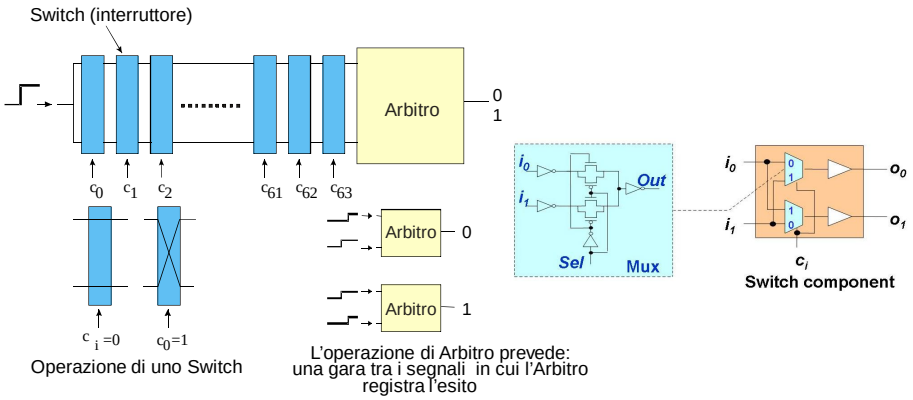
Dove:

$$\boldsymbol{c} = (c_1, c_2, c_3, \ldots, c_n) \in \{0,1\}$$
$$\boldsymbol{r} = (r_1, r_2, r_3, \ldots, r_m) \in \{0,1\}$$

c= challenge
r= response

15

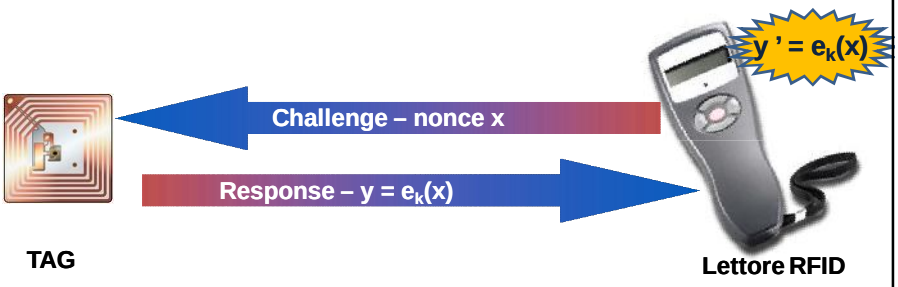
PUF: struttura



- Un PUF sfrutta variazioni nella produzione del circuito per generare un diverso bit response per ogni challenge presentato
- Lo stesso challenge produce generalmente response differenti su tag PUF diversi

16

Un semplice protocollo basato su sfida (challenge-response)

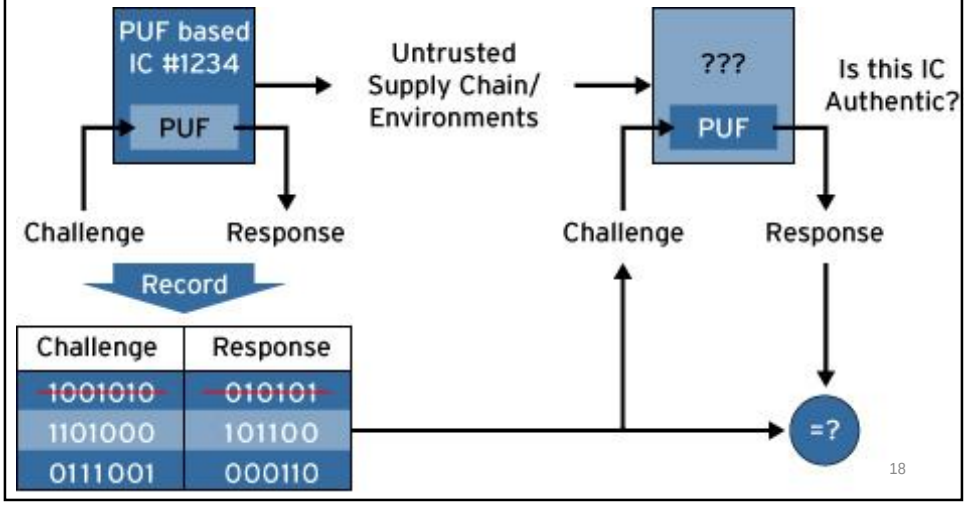


- Il lettore calcola $y' = e_k(x)$ e quindi verifica che $y' = y$
- **e** una generica funzione crittografica con algoritmo disponibile pubblicamente
- **k** una chiave segreta nota solo al lettore e al Tag

17

PUF

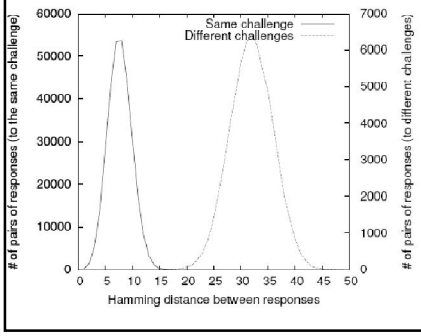
- ✓ Funzione dal comportamento non predicibile che permette di creare coppie challenge-response
- ✓ L'insieme di coppie challenge-response costituisce una sorta di DNA elettronico del tag RFId

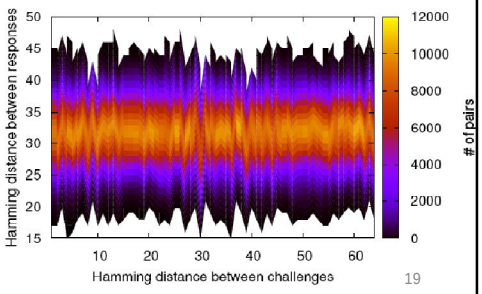


18

Test: Vera X512H

- Velocità di lettura 17 ms nel 96% dei casi, caso peggiore 28 ms
- Lo stesso tag sfidato con lo stesso challenge risponde con response ad una distanza di Hamming minore di 17 bit
- Lo stesso tag fidato con challenge diversi risponde con rsponse ad una distanza di Hamming maggiore di 17 bit
- Non esistono apparenti relazioni tra le distanze dei challenge e le distanze dei response

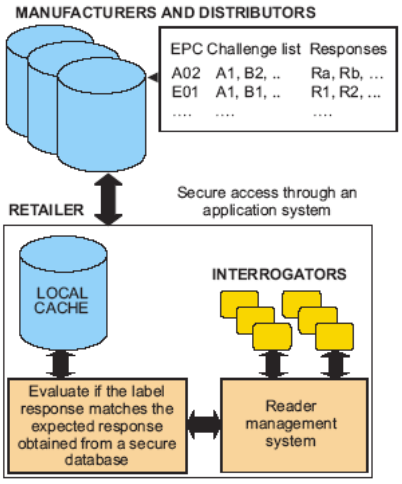




19

PUF: Infrastruttura di sicurezza

- Per garantire la sicurezza nei PUF è necessario:
  - Un database di backend per mantenere coppie challenge response (CRP)
  - Un metodo di distribuzione sicuro per i CRP
  - Costruire una tabella CRP per ogni TAG prima della distribuzione (successivamente dopo la verifica del TAG potrà essere estesa)

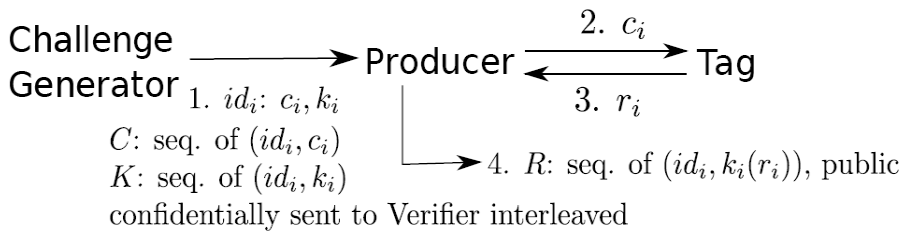


20

Entità	Ruolo	Attori
Producer	È l'entità che genera una serie di CRP interrogando ripetutamente Tag.	**Produttore**
Verifier	È l'entità che vuole verificare l'autenticità dei tag.	**Rivenditore** **Trasportatore** **Enti di controllo** **Cliente**
Tag	È un tag RFId con funzionalità PUF che, quando interrogato con un challenge c, risponde con un response r, calcolato tramite funzioni fisicamente non clonabile.	
Challenge Generator	È un'unità OTP trusted che genera in modo sicuro il challenge **c** con il quale il Verifier eseguirà la verifica di autenticità	**Produttore**

21

Fase di registrazione



Si ottiene

S_C: Trusted, che contiene C e K

S_R: Untrusted, che contiene R

22

Lamport One-Time Password

Autenticazione più forte dell'utilizzo di password statica (Vernam)

Utilizza catene di hash:

- h(s), h(h(s), h(h(h(s))), ..., $h^{1000}(s)$
- utilizza questi valori in ordine inverso come password

A è il dispositivo per autenticarsi presso B:

- A prende un valore w, una funzione hash H() e un intero t, calcola $w_0 = H^t(w)$ e invia w_0 a B
- B mantiene w_0

Il Protocollo per identificare B allo stato i con $1 \leq i \leq t$

- A invia a B: A, i, $w_i = H^{t-i}(w)$
- B controlla $i = i_A$, $H(w_i) = w_{i-1}$
- Entrambi aggiornano $i_A = i_A + 1$

23

Implementazione: sorgente S_C

1. Deve essere trusted
2. Deve contenere la sequenza di C
3. Deve contenere la sequenza di K
4. Si può realizzare come una struttura OTP

Nota: per non tenere traccia della sequenza K, si potrebbe utilizzare come K_i il valore di C_{i+1}

24

Implementazione: sorgente S_R

1. Può essere non trusted
2. Contiene i response cifrati
3. Deve contenere la sequenza di R
4. Autenticità e confidenzialità garantita dalla cifratura
5. Realizzata tramite una qualsiasi memoria di massa

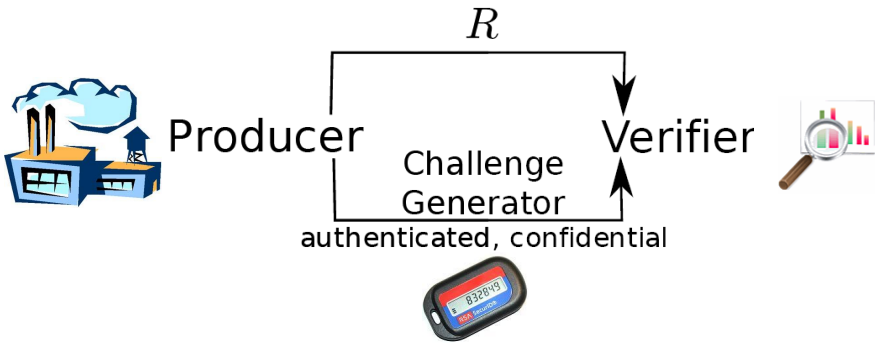


25

Fase di autenticazione

Per eseguire la fase di autenticazione:

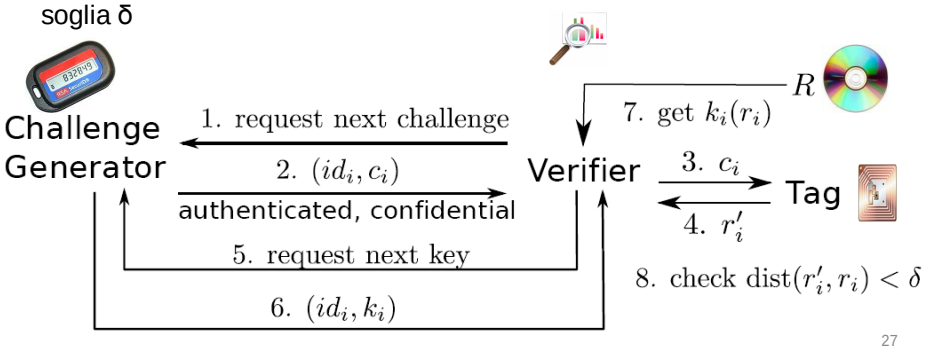
- ✓ Il Producer, dopo la fase di registrazione, crea uno o più Challenge Generator per i vari Verifier
- ✓ Ogni Challenge Generator visualizzerà challenge diversi



26

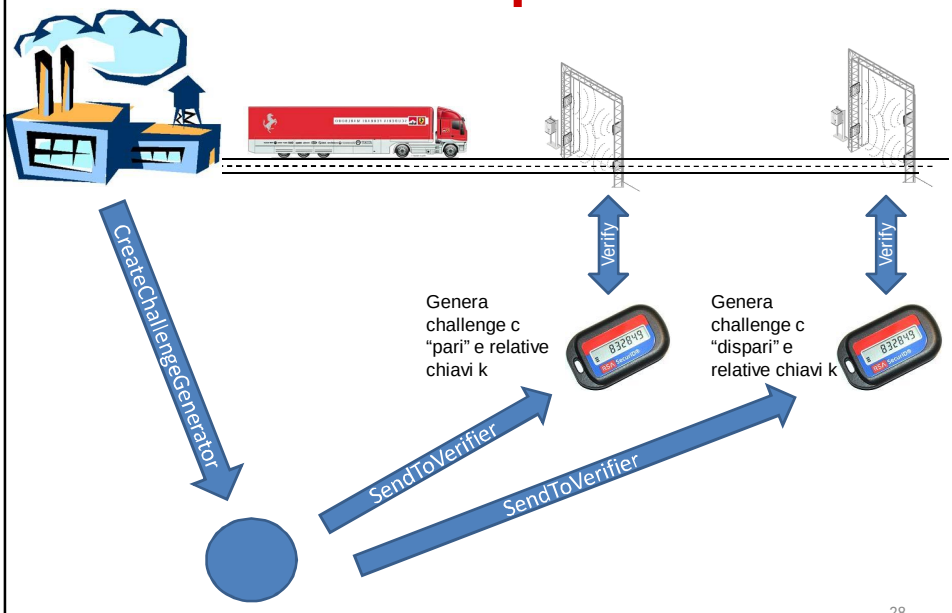
Fase di autenticazione

1. Il Verifier chiede al Challenge Generator il challenge da utilizzare
2. Il Challenge Generator risponde con la coppia (id_i, c_i)
3. Il Verifier sfida il Tag con c_i
4. Il Tag risponde con r'_i
5. Il Verifier chiede al Challenge Generator la chiave di decifratura da utilizzare
6. Il Challenge Generator risponde con la coppia (id_i, k_i)
7. Il Verifier tramite la sorgente S_R prende il valore $[r_i]k_i$ e lo decifra tramite k_i
8. Il verifier controlla che la distanza di Hamming tra r'_i e r_i sia minore di una soglia δ



27

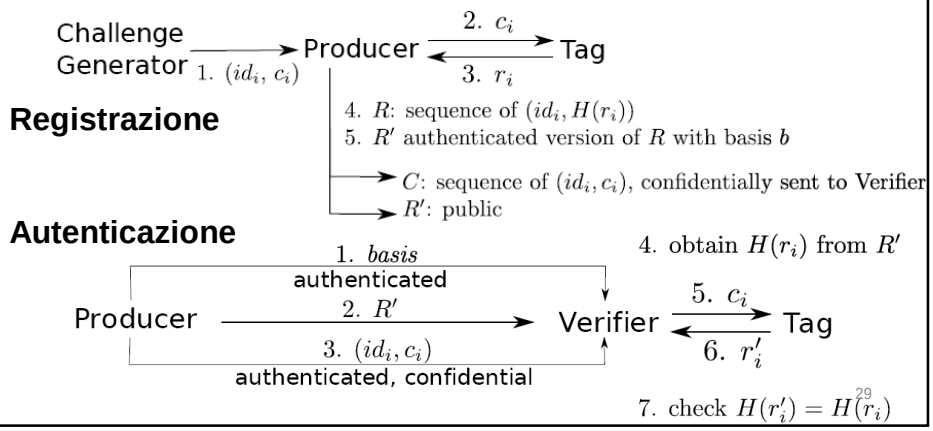
Esempio



28

Modello ideale

1. Utilizza una funzione hash crittografica per garantire confidenzialità dei response
2. Utilizza una ADS per garantire autenticità dai dati in R'
3. Minor costo computazionale
4. Realizzabile solo con tag PUF dal comportamento ideale



Algoritmi e Strutture Dati

ADS - authenticated data structures:

- Merkle Hash Tree
  (R. C. Merkle; A Certified Digital Signature; Crypto '89)
- Skip List Autenticate
  (M. Goodrich, R. Tamassia; Efficient Authenticated Dictionaries with Skip Lists and Commutative Hashing; T. R. John Hopkins Information 2000)

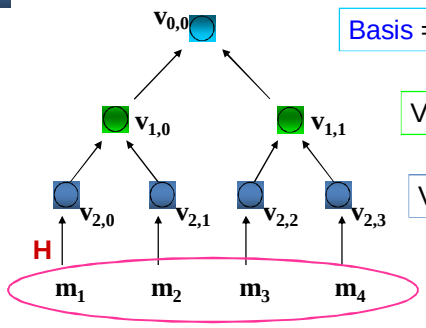
Tali tecniche assicurano lo stesso livello di sicurezza della funzione standard di hash crittografico utilizzata (es. sha-1, sha-256, ...)

Hash Tree (Merkle): costruzione

Basis: root dell'albero autenticato

Sovrastruttura di autenticazione

Hash dei dati



Basis = $V_{0,0}$ = H [($V_{1,0}$) || ($V_{1,1}$)]

$V_{1,1}$ = H [($V_{2,2}$) || ($V_{2,3}$)]

$V_{2,2}$ = H(m_3)

$V_{2,3}$ = H(m_4)

i dati devono essere ordinati

H è una funzione Hash

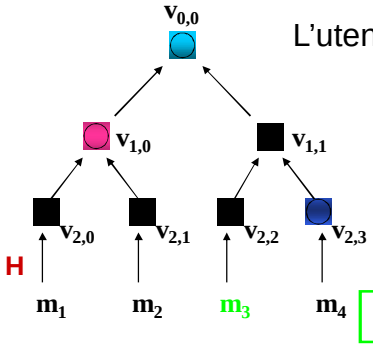
31

Hash Tree (Merkle): verifica

L'utente vuole verificare l'autenticità del dato m_3

La risposta autenticata è composta da: **m_3**, $V_{2,3}$, $V_{1,0}$

e dal Basis firmato dalla CA



L'utente verifica se m_3 è autentico:

$V_{2,2}$ = H m_3

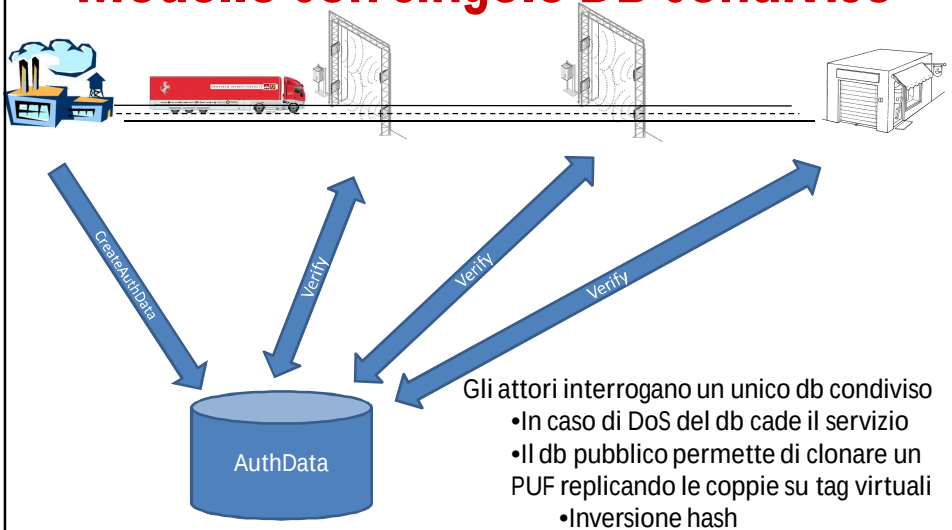
$V_{1,1}$ = H ($V_{2,2}$ || $V_{2,3}$)

$V_{0,0}$ = H ($V_{1,0}$ || $V_{1,1}$)

Se Basis = $V_{0,0}$ allora m_3 è autentico

32

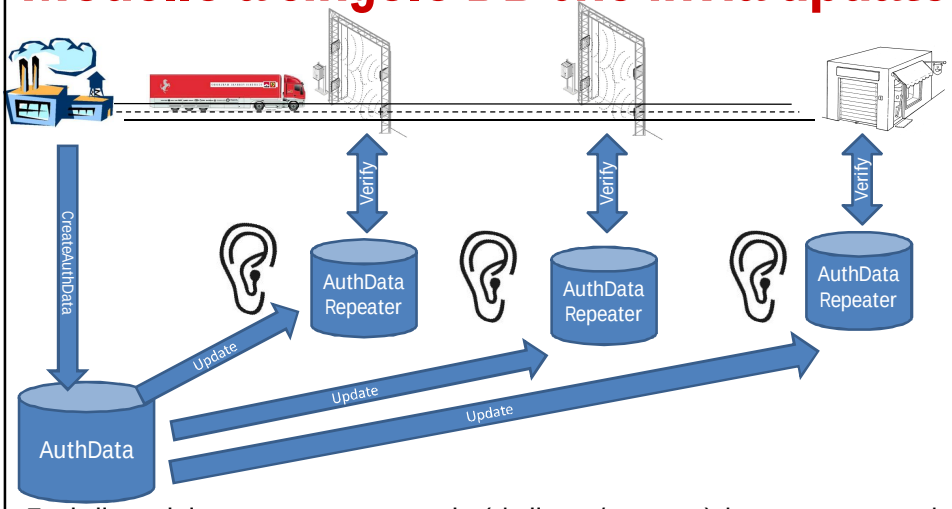
Modello con singolo DB condiviso



Gli attori interrogano un unico db condiviso
- In caso di DoS del db cade il servizio
- Il db pubblico permette di clonare un PUF replicando le coppie su tag virtuali
  - Inversione hash
  - Interrogando il tag originale con la lista dei challenge

33

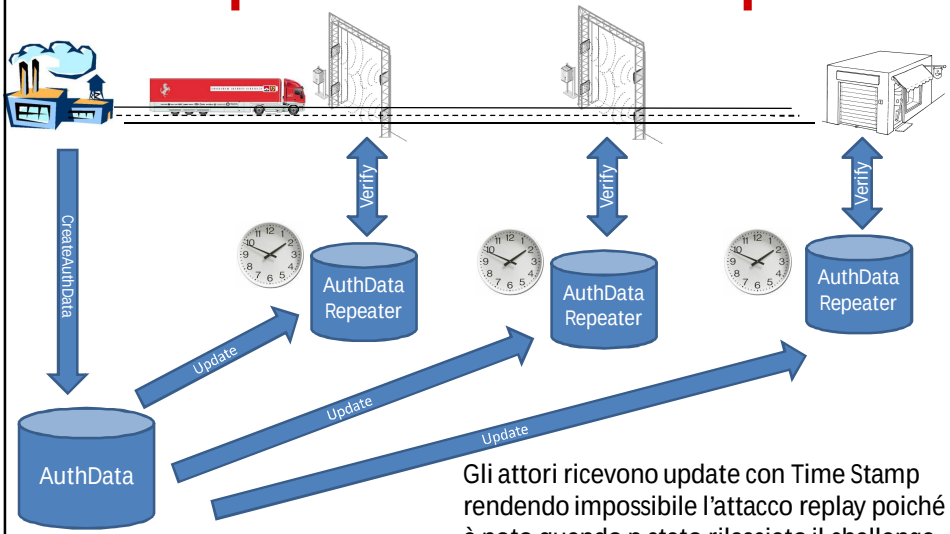
Modello a singolo DB che invia update



Tutti gli attori ricevono una nuova coppia (challenge/response) da un server trusted a un intervallo temporale prefissato:
- La clonazione diventa impossibile poiché non sono noti a priori tutti i challenge
- E' possibile inviare update contenenti vecchi challenge (attacco replay)

34

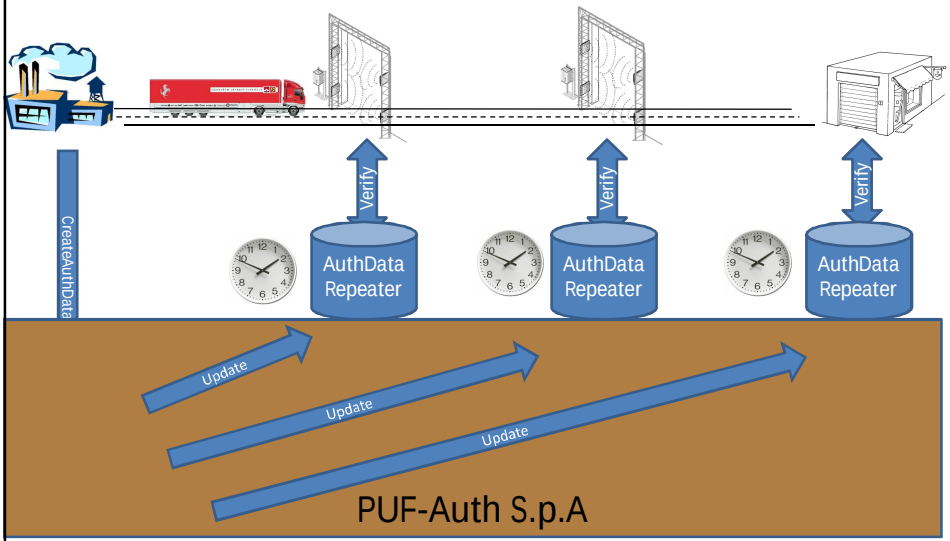
Modello con singolo DB che invia update con Time Stamp



Gli attori ricevono update con Time Stamp rendendo impossibile l'attacco replay poiché è noto quando p stato rilasciato il challenge

35

Modello ad autenticazione Outsourced



Il servizio è gestito da una società esterna che mantiene le strutture dati autenticate e che deve essere trusted per tutti gli attori.

36

Riferimenti

- Cortese, Gemmiti, Palazzi, Pizzonia, Rimondini. Efficient and Practical Authentication of PUF-based RFID Tags. TR 150, 2009; http://web.dia.uniroma3.it/ricerca/rapporti/rt/2009-150.pdf
- Di Battista, Palazzi. Authenticated Relational Tables and Authenticated Skip Lists. In Proc. (DBSEC'07), Springer-Verlag, pages 31-46, 2007
- Ferguson, Schneier. Crittografia pratica. Apogeo, 2005.

37