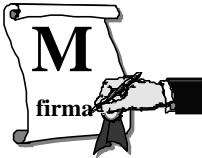

 **Firma Digitale**

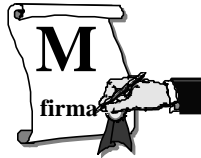
Equivalente alla firma
convenzionale



Firma Digitale 0


 **Firma Digitale**

Equivalente alla firma
convenzionale

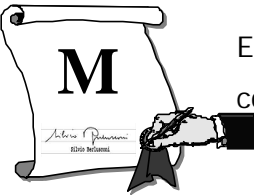


Soluzione naive:
incollare firma digitalizzata


Firma Digitale 1

 **Firma Digitale**

Equivalente alla firma
convenzionale



Soluzione naive:
incollare firma digitalizzata



Firma Digitale 2

 **Desiderata per la Firma Digitale**

La firma digitale deve poter essere
facilmente prodotta dal legittimo firmatario




Nessun utente deve poter
riprodurre la firma di altri



Chiunque può facilmente
verificare una firma



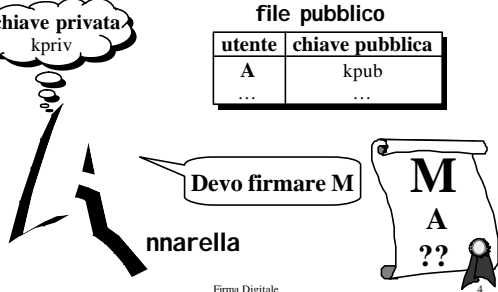
Firma Digitale 3

 **Firma digitale**

chiave privata
kpriv


| file pubblico | |
|---------------|-----------------|
| utente | chiave pubblica |
| A | kpub |
| ... | ... |

Devo firmare M



nnarella

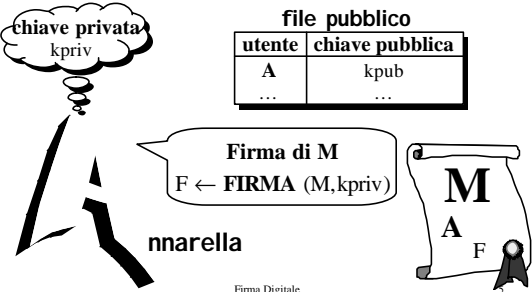
Firma Digitale 4

 **Firma digitale**

chiave privata
kpriv


| file pubblico | |
|---------------|-----------------|
| utente | chiave pubblica |
| A | kpub |
| ... | ... |

Firma di M
 $F \leftarrow \text{FIRMA}(M, kpriv)$




nnarella

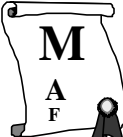
Firma Digitale 5



Verifica firma digitale



erificatore




file pubblico


| utente | chiave pubblica |
|--------|-----------------|
| A | kpub |
| ... | ... |

Devo verificare se F
è una firma di A per M


Firma Digitale 6



Verifica firma digitale



erificatore




file pubblico

| utente | chiave pubblica |
|--------|-----------------|
| A | kpub |
| ... | ... |

Verifica firma di M
vera se **VERIFICA** (F,M,kpub) = SI
falsa altrimenti

Firma Digitale 7



Firme digitali che vedremo

- RSA
- Digital Signature Standard (DSS)

Firma Digitale 8



RSA

Proposto nel 1978 da







Rivest Shamir Adleman

Sicurezza basata sulla difficoltà di fattorizzare


Firma Digitale 9



Chiavi RSA



nnarella



chiave privata
(n,d)


file pubblico

| utente | chiave pubblica |
|--------|-----------------|
| A | (n,e) |
| ... | ... |


$n = pq$
p,q primi

$ed = 1 \pmod{(p-1)(q-1)}$


Firma Digitale 10



Firma RSA



nnarella

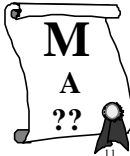


chiave privata
(n,d)

file pubblico

| utente | chiave pubblica |
|--------|-----------------|
| A | (n,e) |
| ... | ... |

Devo firmare M



Firma Digitale 11

Firma RSA

chiave privata
(n,d)

Firma di M
 $Firma_{(n,d)}(M) = M^d \text{ mod } n$

innarella

file pubblico

| utente | chiave pubblica |
|--------|-----------------|
| A | (n,e) |
| ... | ... |

M
A
 $M^d \text{ mod } n$

Firma Digitale

Verifica firma RSA

Verificatore

file pubblico

| utente | chiave pubblica |
|--------|-----------------|
| A | (n,e) |
| ... | ... |

M
A
F

Devo verificare se F è una firma di A per M

Firma Digitale

Verifica firma RSA

Verificatore

file pubblico

| utente | chiave pubblica |
|--------|-----------------|
| A | (n,e) |
| ... | ... |

M
A
F

Verifica firma di M
vera se $M = F^e \text{ mod } n$
falsa altrimenti

Firma Digitale

"Piccolo" esempio: Chiavi RSA

chiave privata
(n=3337, d=1019)

innarella

file pubblico

| utente | chiave pubblica |
|--------|--------------------|
| A | (n = 3337, e = 79) |
| ... | ... |

$3337 = 47 \cdot 71$
 $p = 47, q = 71$

$ed = 79 \cdot 1019 = 1 \text{ mod } 3220$
 $(p-1)(q-1) = 46 \cdot 70 = 3220$

Firma Digitale

"Piccolo" esempio: Firma RSA

chiave privata
(n=3337, d=1019)

innarella

file pubblico

| utente | chiave pubblica |
|--------|--------------------|
| A | (n = 3337, e = 79) |
| ... | ... |

Devo firmare M=1570

1570
A
??

Firma Digitale

"Piccolo" esempio: Firma RSA

chiave privata
(n=3337, d=1019)

innarella

file pubblico

| utente | chiave pubblica |
|--------|--------------------|
| A | (n = 3337, e = 79) |
| ... | ... |

Firma di 1570
 $= 1570^{1019} \text{ mod } 3337$
 $= 668$

1570
A
668

Firma Digitale

"Piccolo" esempio: Verifica firma

| file pubblico | |
|---------------|--------------------|
| utente | chiave pubblica |
| A | (n = 3337, e = 79) |
| ... | ... |

Verifica firma di 1570
 $1570 = 668^{79} \text{ mod } 3337$

Verificatore

Firma Digitale 18

Correttezza verifica firma RSA

$$F^e \text{ mod } n = (M^d)^e \text{ mod } n$$

$$= M^{ed} \text{ mod } n$$

$$= M \text{ mod } n$$

$$= M$$

$ed = 1 \text{ mod } (p-1)(q-1)$
 $x \in \mathbb{Z}_n^* \Rightarrow x^{(p-1)(q-1)} = 1 \text{ mod } n$

poichè $0 \leq M < n$

Prova per tutti gli x mediante il teorema del resto cinese

Firma Digitale 19

Esercizio

Svolgere "piccolo" esempio firma RSA

- Fissato $e = 3$
- Calcolo p, q
- Calcolo n
- Calcolo d
- Calcolo firma
- Verifica firma

Firma Digitale 20

Firma digitale di messaggi grandi

Se $M > n$, come si firma?

Prima soluzione

$M_1 < n$

$\text{Firma}(M) \leftarrow (\text{Firma}(M_1), \text{Firma}(M_2), \dots)$

Problemi { Efficienza
 Permutazione/composizione delle firme \rightarrow nuova firma

Firma Digitale 21

Funzioni Hash

lunghezza arbitraria/finita \rightarrow Funzione Hash \rightarrow b bit

- Il valore hash $h(M)$ è una rappresentazione non ambigua e non falsificabile del messaggio M
- Proprietà:
 - comprime
 - facile da computare
 - **Sicurezza forte**: computazionalmente difficile trovare 2 diversi messaggi con lo stesso valore hash
 - **One-way**: dato y è computazionalmente difficile trovare M tale che $y = h(M)$

Firma Digitale 22

Firma digitale di messaggi grandi

Soluzione

$\text{Firma}(M) \leftarrow (\text{Firma}(h(M)))$

Vantaggi { Efficienza
 Integrità

Firma Digitale 23

Sicurezza firma RSA

Voglio falsificare la firma di M da parte di A

| file pubblico | |
|---------------|-----------------|
| utente | chiave pubblica |
| A | (n,e) |
| ... | ... |

Devo calcolare $M^d \text{ mod } n$

Equivalente a "rompere" crittostema RSA

M
A
??

Firma Digitale 24

Sicurezza firma RSA

Voglio generare messaggi e firme da parte di A

| file pubblico | |
|---------------|-----------------|
| utente | chiave pubblica |
| A | (n,e) |
| ... | ... |

1. Scelgo F a caso
2. $M \leftarrow F^e \text{ mod } n$

M
A
F

Firma Digitale 25

Firma RSA con hash

chiave privata (n,d)

| file pubblico | |
|---------------|-----------------|
| utente | chiave pubblica |
| A | (n,e) |
| ... | ... |

Firma di M
 $Firma_{(n,d)}(M) = [h(M)]^d \text{ mod } n$

nnarella

M
A
 $[h(M)]^d \text{ mod } n$

Firma Digitale 26

Verifica firma RSA con hash

| file pubblico | |
|---------------|-----------------|
| utente | chiave pubblica |
| A | (n,e) |
| ... | ... |

M
A
F

Devo verificare se F è una firma di A per M

erificatore

Firma Digitale 27

Verifica firma RSA

| file pubblico | |
|---------------|-----------------|
| utente | chiave pubblica |
| A | (n,e) |
| ... | ... |

M
A
F

Verifica firma di M vera se $h(M) = F^e \text{ mod } n$ falsa altrimenti

erificatore

Firma Digitale 28

Sicurezza firma RSA

Voglio generare messaggi e firme da parte di A

| file pubblico | |
|---------------|-----------------|
| utente | chiave pubblica |
| A | (n,e) |
| ... | ... |

1. Scelgo F a caso
2. $z \leftarrow F^e \text{ mod } n$
3. $M \leftarrow h^{-1}(z)$

Come faccio ad invertire h?
 $M \leftarrow h^{-1}(z)$

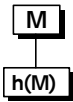
M
A
F

Firma Digitale 29

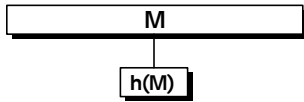


Firma digitale con hash

messaggi piccoli



messaggi grandi



$$\text{Firma}(M) \leftarrow (\text{Firma}(h(M)))$$

Vantaggi

- Efficienza
- Integrità
- Sicurezza



Digital Signature Standard (DSS)

- Proposto nell'agosto del 1991 dal National Institute of Standard and Technology (NI ST)
 - Digital Signature Algorithm (DSA)
 - Digital Signature Standard (DSS)
- Standard rivisto nel 1993, in risposta alle critiche
- Modifica ingegnosa dello schema di El Gamal
- Firme DSS sempre di 320 bit (buone per smart card)
- Sicurezza basata sulla difficoltà del logaritmo discreto



Chiavi DSA

chiave privata
(p,q,α,s)

file pubblico

| utente | chiave pubblica |
|--------|-----------------|
| A | (p,q,α,β) |
| ... | ... |

nnarella



Chiavi DSA

chiave privata
(p,q,α,s)

file pubblico

| utente | chiave pubblica |
|--------|-----------------|
| A | (p,q,α,β) |
| ... | ... |

p primo di 512, ..., 1024 bit

q primo di 160 bit, q|(p-1)

α in Z_p^* di ordine q

β = α^r mod p

s numero casuale, s < q

nnarella



Chiavi DSA ("piccolo" esempio)

chiave privata
(7879,101,170,75)

file pubblico

| utente | chiave pubblica |
|--------|---------------------|
| A | (7879,101,170,4567) |
| ... | ... |

p = 7879 primo

q = 101 primo, p = 78q + 1

α = 170 ∈ Z_{7879}^* di ordine 101

4567 = 170⁷⁵ mod 7879

s = 75 numero casuale

nnarella



Firma DSA

chiave privata
(p,q,α,s)

file pubblico

| utente | chiave pubblica |
|--------|-----------------|
| A | (p,q,α,β) |
| ... | ... |

Devo firmare M

nnarella



Firma DSA

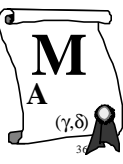
chiave privata
(p, q, α, s)

file pubblico

| utente | chiave pubblica |
|--------|-----------------|
| A | (p, q, α, β) |
| ... | ... |

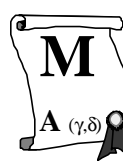
Firma di M

$r \leftarrow$ numero casuale in $[1, q-1]$
 $\gamma \leftarrow (\alpha^r \bmod p) \bmod q$
 $\delta \leftarrow (\text{SHA}(M) + s\gamma)r^{-1} \bmod q$
firma_(p,q,α,s)(M, r) = (γ, δ)



Emittente
Firma Digitale

Verifica firma DSA



file pubblico


| utente | chiave pubblica |
|--------|-----------------|
| A | (p, q, α, β) |
| ... | ... |

Devo verificare se (g, d) è una firma di A per M

Verificatore

Firma Digitale

Verifica firma DSA



file pubblico

| utente | chiave pubblica |
|--------|-----------------|
| A | (p, q, α, β) |
| ... | ... |

Verifica firma di M

$e' \leftarrow \text{SHA}(M)\delta^{-1} \bmod q$
 $e'' \leftarrow \gamma\delta^{-1} \bmod q$
 vera se $\gamma = (\alpha^{e'}\beta^{e''} \bmod p) \bmod q$
 falsa altrimenti

Verificatore

Firma Digitale

Efficienza firma DSA

Firma_DSA(M, p, q, α, s)

$r \leftarrow$ numero casuale in $[1, q-1]$
 $\gamma \leftarrow (\alpha^r \bmod p) \bmod q$
 $\delta \leftarrow (\text{SHA}(M) + s\gamma)r^{-1} \bmod q$
output firma_(p,q,α,s)(M, r) = (γ, δ)

- Lunghezza firma = 320 bit
- Computazioni off-line: r, sγ, r⁻¹ mod q
- Computazioni on-line: SHA(M), +, ·

Firma Digitale

Verifica firma DSA

Verifica_firma_DSA(M, γ, δ, p, q, α, β)

$e' \leftarrow \text{SHA}(M)\delta^{-1} \bmod q$
 $e'' \leftarrow \gamma\delta^{-1} \bmod q$

ver_(p,q,α,β)(M, γ, δ) = $\begin{cases} \text{vera se } \gamma = (\alpha^{e'}\beta^{e''} \bmod p) \bmod q \\ \text{falsa altrimenti} \end{cases}$

Output ver_(p,q,α,β)(M, γ, δ)

Firma Digitale

Correttezza verifica firma DSA

$$\begin{aligned}
 & (\alpha^{e'}\beta^{e''} \bmod p) \bmod q \\
 &= (\alpha^{\text{SHA}(M)\delta^{-1} \bmod q} \alpha^{s\gamma\delta^{-1} \bmod q} \bmod p) \bmod q \\
 &= (\alpha^{\text{SHA}(M)\delta^{-1} + s\gamma\delta^{-1}} \bmod p) \bmod q \\
 &= (\alpha^r \bmod p) \bmod q \\
 &= \gamma
 \end{aligned}$$

Firma Digitale



Efficienza delle computazioni

Come effettuare le computazioni?

- Generazione numeri primi p e q
- Generazione di α (elemento di ordine q)
- Elevazione a potenza modulare
- Calcolo inverso $r^{-1} \bmod q$



Generazione di p e q

Scegli p

Scegli q di 160 bit tale che $q|(p-1)$



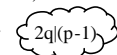
Generazione di p e q

Scegli p

Scegli q di 160 bit tale che $q|(p-1)$



Generazione di p e q

- Scegli un primo q di 160 bit
- Scegli un primo p di 512/1024 bit tale che $q|(p-1)$
 - o Scegli X di 512 bit (oppure ... 1024 bit)
 - o $p \leftarrow X - ((X \bmod 2q) - 1)$ 
 - o se p è primo e $p \geq 2^{511}$ esci altrimenti riprova



Scelta di un elemento di ordine q

- Ordine di $\alpha \in \mathbb{Z}_p^*$ = il più piccolo intero positivo r tale che $\alpha^r = 1 \bmod p$
- p, q primi tali che $q|(p-1)$

Scegli_ordineq (p, q)

1. $g \leftarrow$ elemento scelto a caso in \mathbb{Z}_p^*
2. $\alpha \leftarrow g^{(p-1)/q} \bmod p$
3. **if $\alpha \neq 1$ then return α else go to 1.**



Correttezza di Scegli_ordineq

Scegli_ordineq (p, q)

1. $g \leftarrow$ elemento scelto a caso in \mathbb{Z}_p^*
2. $\alpha \leftarrow g^{(p-1)/q} \bmod p$
3. **if $\alpha \neq 1$ then return α else go to 1.**

- $\alpha^q \equiv (g^{(p-1)/q})^q \equiv g^{p-1} \equiv 1 \bmod p$
- q è il più piccolo intero tale che $\alpha^q \equiv 1 \bmod p$
- α è di ordine q

dal Teorema di Lagrange
l'ordine di α divide q



Probabilità successo singola iterazione

- Se g è un generatore allora $g^{(p-1)/q} \neq 1 \pmod p$
- Probabilità successo \geq Probabilità che g è generatore $> 1/(6 \ln \ln(p-1))$
- Numero medio di iterazioni $< 6 \ln \ln(p-1)$

| | |
|----------|---|
| 512 bit | $6 \cdot \ln \ln(2^{512}) \approx 35,23$ |
| 1024 bit | $6 \cdot \ln \ln(2^{1024}) \approx 39,38$ |
| 2048 bit | $6 \cdot \ln \ln(2^{2048}) \approx 43,54$ |



Chiavi globali ed individuali

chiave privata
(p,q,α,s)

file pubblico

| utente | chiave pubblica |
|--------|-----------------|
| A | (p,q,α,β) |
| ... | ... |

- Sicurezza basata sul valore privato s
- I valori p,q,a possono essere gli stessi per un gruppo di utenti
- Un'autorità sceglie p,q,a
- Il singolo utente sceglie solo s,b



Chiavi globali ed individuali

chiave privata
(p,q,α,s)

file pubblico

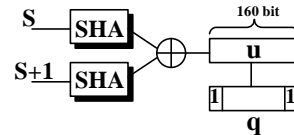
| utente | chiave pubblica |
|--------|-----------------|
| A | (p,q,α,β) |
| ... | ... |

- Sicurezza basata sul valore privato s
- I valori p,q,a possono essere gli stessi per un gruppo di utenti
- Un'autorità sceglie p,q,a
- Il singolo utente sceglie solo s,b



Generazione di q

- Scegli a caso S di ≥ 160 bit

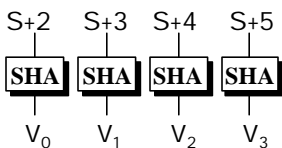


- Ripeti con un nuovo S finché q è primo

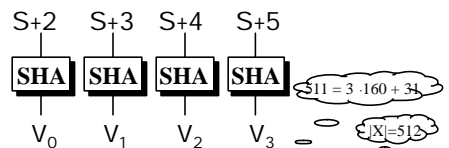
S è un testimone della validità di q



Generazione di p (512 bit)



Generazione di p (512 bit)



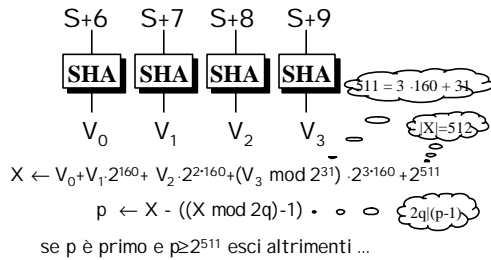
$$X \leftarrow V_0 + V_1 \cdot 2^{160} + V_2 \cdot 2^{2 \cdot 160} + (V_3 \bmod 2^{31}) \cdot 2^{3 \cdot 160} + 2^{511}$$

$$p \leftarrow X - ((X \bmod 2q) - 1) \cdot \dots \cdot 2q(p-1)$$

se p è primo e $p \geq 2^{511}$ esci altrimenti ...

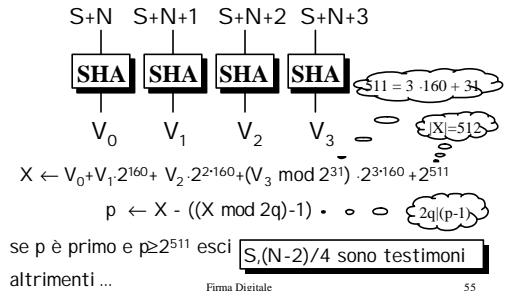


Generazione di p (512 bit)



Generazione di p (512 bit)

N ← 2,6,10,..., (per 4096 volte)



Generazione di p e q

```

Seleziona_pq(L)
(1) Computa interi n e b tali che L-1=160n+b
(2) repeat
(3)   repeat
(4)     S ← sequenza casuale di almeno 160 bit
(5)     g ← |S|
(6)     U ← SHA(S) ⊕ SHA((S+1) mod 2^n)
(7)     Forma q da U ponendo il MSB ed il LSB ad 1
(8)   until q primo
(9)   C ← 0
(10)  N ← 2
(11)  repeat
(12)    for k=0 to n do V_k ← SHA(S+N+k) mod 2^n
(13)    W ← V_0 + V_1 · 2^{160} + ... + V_{n-1} · 2^{160(n-1)} + (V_n mod 2^n) · 2^{160n}
(14)    X ← W + 2^{2^{n-1}}
(15)    p ← X - ((X mod 2q) - 1)
(16)  until (p primo) or (p < 2^{2^{n-1}})
(17)  if p < 2^{2^{n-1}}
(18)    then C ← C+1
(19)    N ← N+n+1
(20)    if C < 4096 then goto step (12)
(21)  else Help ← falso
(22) until Help
(23) return p,q,S,C

```



Confronto tempi firme RSA e DSA

| | DSS | RSA | DSS con p,q,a comuni |
|-----------------|-------------------|---------|----------------------|
| precomputazioni | 14 sec | | 4 sec |
| firma | 0.3 sec | 15 sec | 0.3 sec |
| verifica | 16 sec | 1.5 sec | 10 sec |
| | 1-5 sec Off Cards | | 1-3 sec Off Cards |

- Implementazioni su smart card [1993]
- Computazioni Off Cards su 80386 a 33MHz



Prestazioni algoritmi

Celeron 850MHz, Windows 2000, Crypto++
millisecondi/operazione

| | bit chiave | firma | Firma con precomputazione | verifica |
|-----|------------|-------|---------------------------|----------|
| RSA | 512 | 1,92 | | 0,13 |
| DSA | 512 | 1,77 | 1,19 | 2,02 |
| RSA | 1024 | 10,29 | | 0,30 |
| DSA | 1024 | 5,50 | 2,27 | 6,38 |



Prestazioni

Pentium II 400
OpenSSL

| | bit chiave | firme/s | verifiche/s |
|-----|------------|---------|-------------|
| RSA | 512 | 342 | 3287 |
| DSA | 512 | 331 | 273 |
| RSA | 1024 | 62 | 1078 |
| DSA | 1024 | 112 | 94 |
| RSA | 2048 | 10 | 320 |
| DSA | 2048 | 34 | 27 |