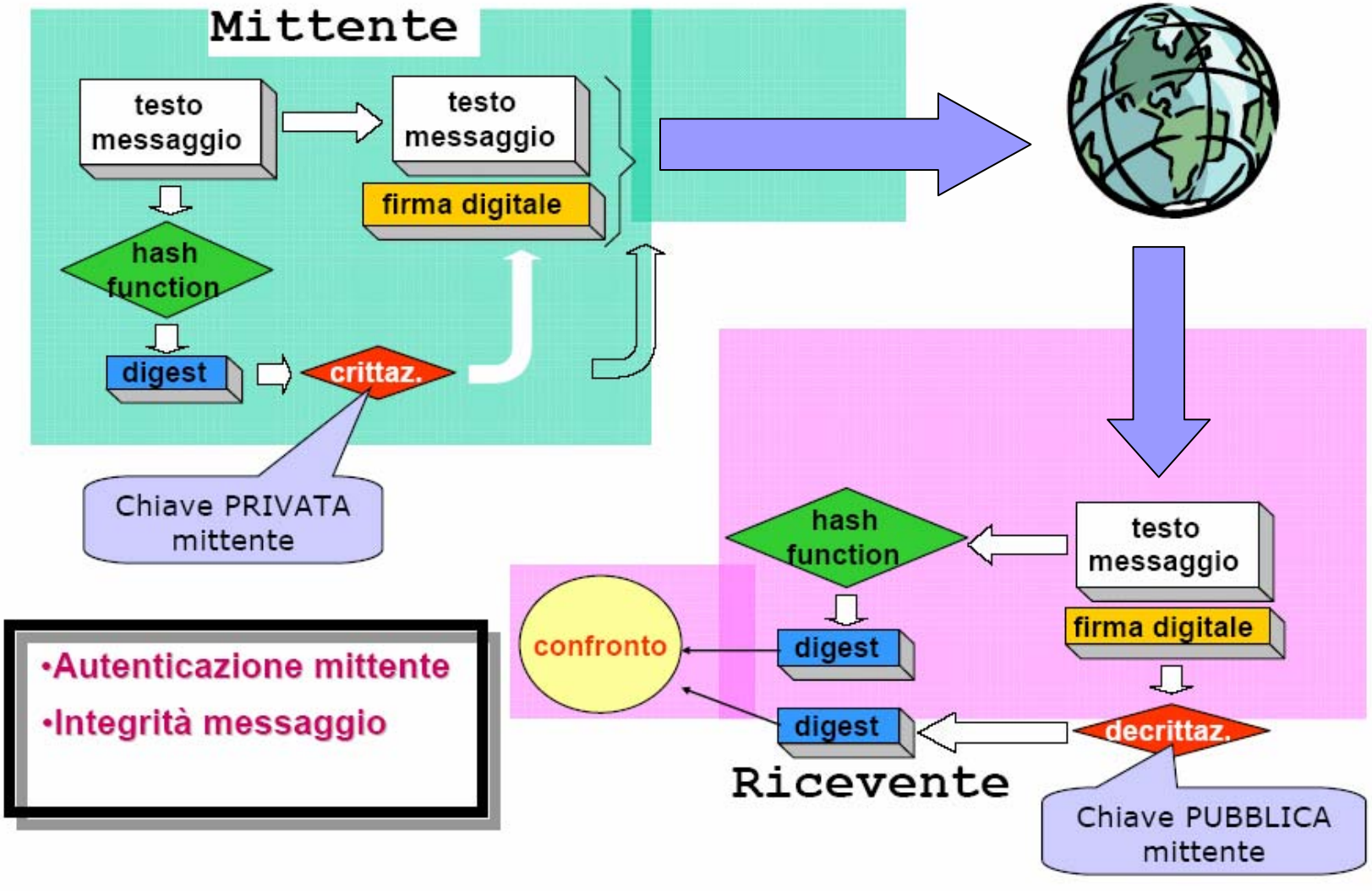
Problematiche operative dei tool per firma digitale

D'Amico Stefano
Security WorkShop
Università di Catania

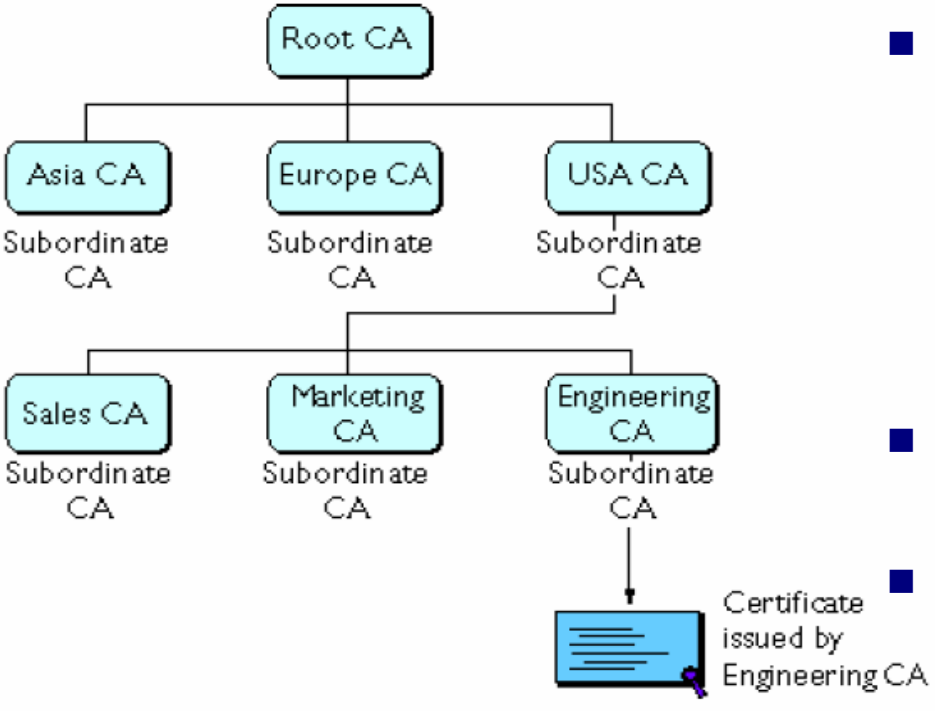
Firma Digitale con Crittografia Asimmetrica



Problema dell'identità

- L'utilizzo dell'algoritmo di firma digitale garantisce che un documento sia stato firmato con una determinata chiave privata
- Questo non dice nulla sull'AUTORE della firma,se non si è a conoscenza in modo certo della chiave pubblica di una persona
- Non c'è un modo sicuro di scambiarsi le chiavi pubbliche, se non out of band (di persona e mediante dischetto, per esempio)
- Lo scopo di una **PKI** (Public Key Infrastructure) è di garantire l'associazione chiave pubblica mittente su una scala più vasta

Gerarchia
Certification Authorities



- Per poter garantire l'associazione tra un soggetto e una determinata chiave pubblica è necessario utilizzare una terza parte fidata che autentichi la chiave pubblica
- Questa terza parte è la Certification Authority (CA)
- La CA rilascia un **certificato digitale** che contiene alcune informazioni sull'utente che l'ha richiesto, tra cui la chiave pubblica dell'utente
- Il certificato è firmato digitalmente dalla CA

Firma digitale a valor legale (fino ad oggi)

- Normata dal D.P.R. 513/97 e successive modificazioni;
- Basata sull'uso di certificati X.509 (come da DPCM 8/02/99) a bordo di strumenti di firma hardware modificabili solo all'origine (al momento smart card, ma non necessariamente)
- Rilasciata da certificatori abilitati iscritti nell'elenco AIPA (Agenzia per l'Informatica nella Pubblica Amministrazione) e dotati di particolari requisiti (a livello di struttura societaria e di procedure)
- **Ha lo stesso valore di una firma su documento cartaceo, e non può essere "disconosciuta"** se non "a querela di parte"

Utilità e inutilità della f.d.

- Cosa si può fare con la firma digitale:
  i. Firmare un contratto (“scrittura privata”)
  ii. Firmare documenti destinati alla Pubblica Amministrazione
  iii. Applicare una marca temporale ad un Documento
- Cosa non si può fare:
  i. Comprare una casa (l’atto notarile richiede la presenza): potete però usare una smart card presso il notaio.
  ii. Firmare un referendum (dovete essere identificati da un pubblico ufficiale) potreste però usare la smart card al “banchetto”
- Per cosa non è utile una firma a valore legale
  i. per il commercio online: avete mai firmato un contratto per comprare un oggetto al supermercato ?
  ii. Per identificarsi verso un server remoto SSL: non avrebbe comunque valore legale!

Alcuni punti fermi…

- Una firma “digitale” è persino più forte di una firma “reale”

i. Il documento firmato realmente può essere modificato, mentre una sequenza di bit firmata digitalmente, a meno di perdita della chiave privata o di vulnerabilità dell’algoritmo, non è modificabile

ii. La firma reale è sempre uguale e può essere riprodotta, la firma digitale varia a seconda dei bit di informazione firmati.

- Gli algoritmi di crittografia asimmetrica utilizzati per la firma digitale italiana sono fondamentalmente sicuri.

i. La crittografia asimmetrica si basa, lo ricordiamo, sulla “sostanziale impossibilità” di invertire un’operazione matematica. Per ora tale presupposto non è stato compromesso da nessuno

- Tuttavia, sono emersi alcuni problemi …

Bug su documenti Microsoft Word

- Bug scoperto il 9 Settembre 2002
- Il software di verifica di Firma Digitale di vari certificatori (originariamente rilevato su Dike di InfoCamere) consente di firmare documenti in formato Word con campi dinamici.
- In questi documenti il contenuto del file non cambia ma ciò che viene visualizzato all'utente si!!!
- Il software non si accorge di nessuna alterazione del file quindi non avvisa l'utente e firma il documento in modo del tutto normale.
- Il software firma un file, ma allo stesso tempo firma documenti diversi.

Risultati ?

provaDiKe.doc - Microsoft Word

File Edit View Insert Format Tools Table Window Help

Normal Arial 11

Questo è un documento di prova scritto con MS Word 2000.
Nelle righe seguenti sono inseriti campi variabili tra asterischi:

Data e ora *17/09/2002 08.48.53*
I dati inseriti sono 17/09/2002 08.48.53

Data e ora *17/09/2002 08.48.53* (Questa non dovrebbe modificarsi)
I dati inseriti sono 17/09/2002 08.48.53

Nome del documento *provaDiKe.doc*
Dato inserito provaDiKe.doc

Nome autore *Manlio Cammarata*
Dato inserito Manlio Cammarata

Nome utente *Manlio Cammarata*
Dato inserito Manlio Cammarata

Ora il file sarà firmato digitalmente con DiKe e dovremo controllare se la verifica va a buon fine anche dopo l'aggiornamento dei campi.

Page 1 Sec 1 1/1 At 2.5cm Ln 1 Col 1 REC TRK EXT OVR Italian (Italy

DiKe

File Modifica Strumenti Opzioni Guida

Nr. Firmatari: 1
Firmatario:1 CAMMARATA MANLIO;Cod. Fiscale:CMMMNL47P04L424H;Identificativo:2001111111812
Stato:IT;Organizzazione:Non Dichiarato;Unità Organizzativa:RA=INFOCAMERE S.C.p.A.
verifica completata correttamente

Documento: d:\test\provaDiKe18.doc.p7m Dimensioni: 21944 Data: 18/09/2002 14.20.44

Questo è un documento di prova scritto con MS Word 2000.
Nelle righe seguenti sono inseriti campi variabili tra asterischi:

Data e ora *18/09/2002 14.28.15*
I dati inseriti sono 17/09/2002 08.48.53

Data e ora *17/09/2002 08.48.53* (Questa non dovrebbe modificarsi)
I dati inseriti sono 17/09/2002 08.48.53

Nome del documento *provaDiKe.doc*
Dato inserito provaDiKe.doc

Nome autore *Manlio Cammarata*
Dato inserito Manlio Cammarata

Nome utente *Manlio Cammarata*
Dato inserito Manlio Cammarata

Ora il file sarà firmato digitalmente con DiKe e dovremo controllare se la verifica va a buon fine anche dopo l'aggiornamento dei campi.

Il Programma o le norme ?

- Il programma esegue correttamente la verifica crittografica: la stringa di bit non cambia.
- Tuttavia, il programma aderisce alle norme sulla Firma Digitale?
- Art. 1, sulla doc. amministrativa: il documento informatico è "*la rappresentazione informatica di atti, fatti o dati giuridicamente rilevanti*". Rappresentazione: ciò che si vede, non il mezzo.
- La Firma digitale va applicata non al **File**, ma al **documento.** Se il programma verifica l'integrità non del **documento** in senso giuridico, ma solo quella del **file** ("evidenza informatica" dicono le norme tecniche) sul quale è stato calcolato l'hash e quindi generata la firma stessa, esso verifica l'integrità dell'**evidenza informatica**, che non ha alcun valore legale!

Possibili soluzioni!

- L'applicazione di firma potrebbe mostrare una copia in pdf del documento al momento della firma, e firmare quest'ultima. Ciò creerebbe sicuramente problemi di compatibilità per i certificatori.
- Microsoft, il 30 gennaio 2003, ha completato una patch per Office per consentire la disabilitazione dei campi dall'esterno.
- Lo stesso DiKe nella nuova versione mostra una finestra di avvertimento…
- Adesso Microsoft Word modifica i campi dinamici solo su richiesta dell'utente e modifica anche l'**evidenza informatica.**

Bug di Firma&Cifra

- Firma&Cifra è l'applicativo di PostECom per la verifica di Firma Digitale
- Bug segnalato da anonimo il 20 Marzo 2003
- Risultato del bug: possibilità di creare qualsiasi certificato falso e di far verificare una firma apposta con tale certificato a Firma&Cifra.
- Anche in questo caso, il problema non sono gli algoritmi crittografici…

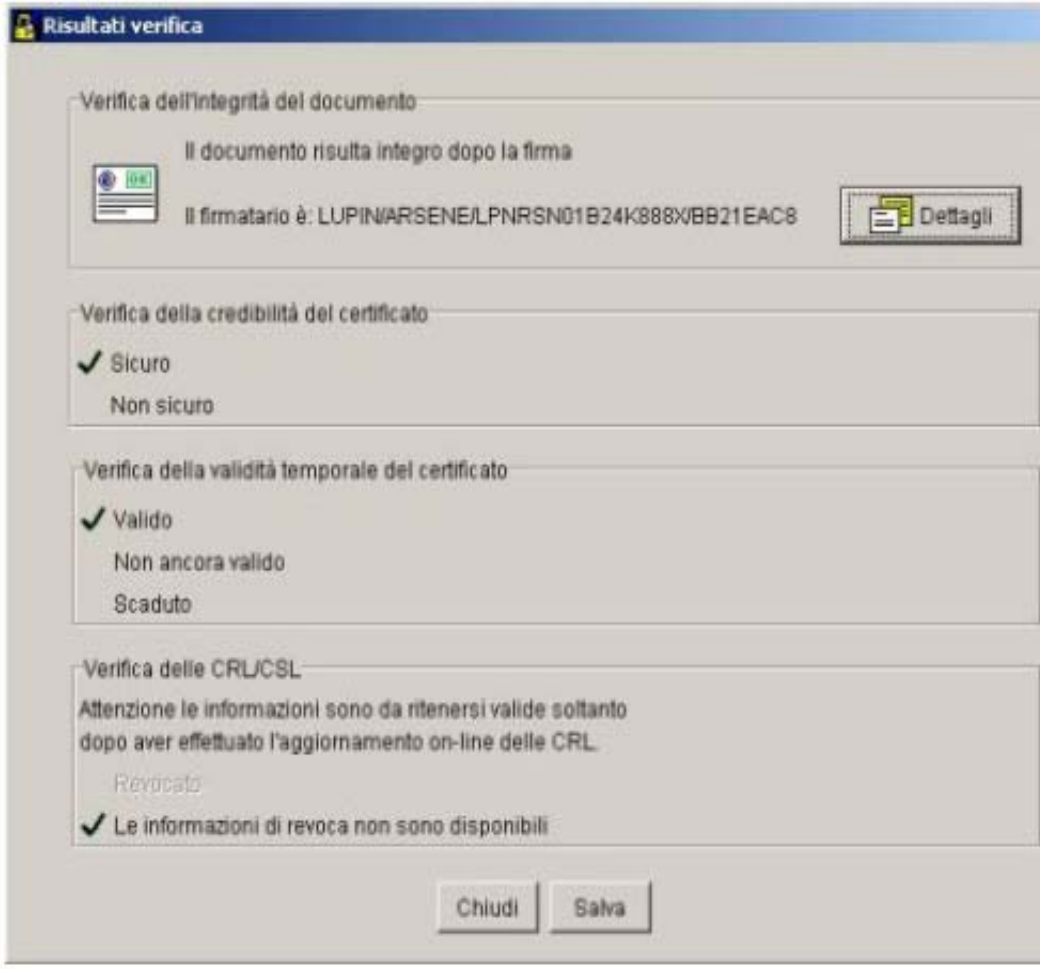
Meccanismo del Bug

- Per verificare una Firma bisogna possedere il certificato che l'ha generata:

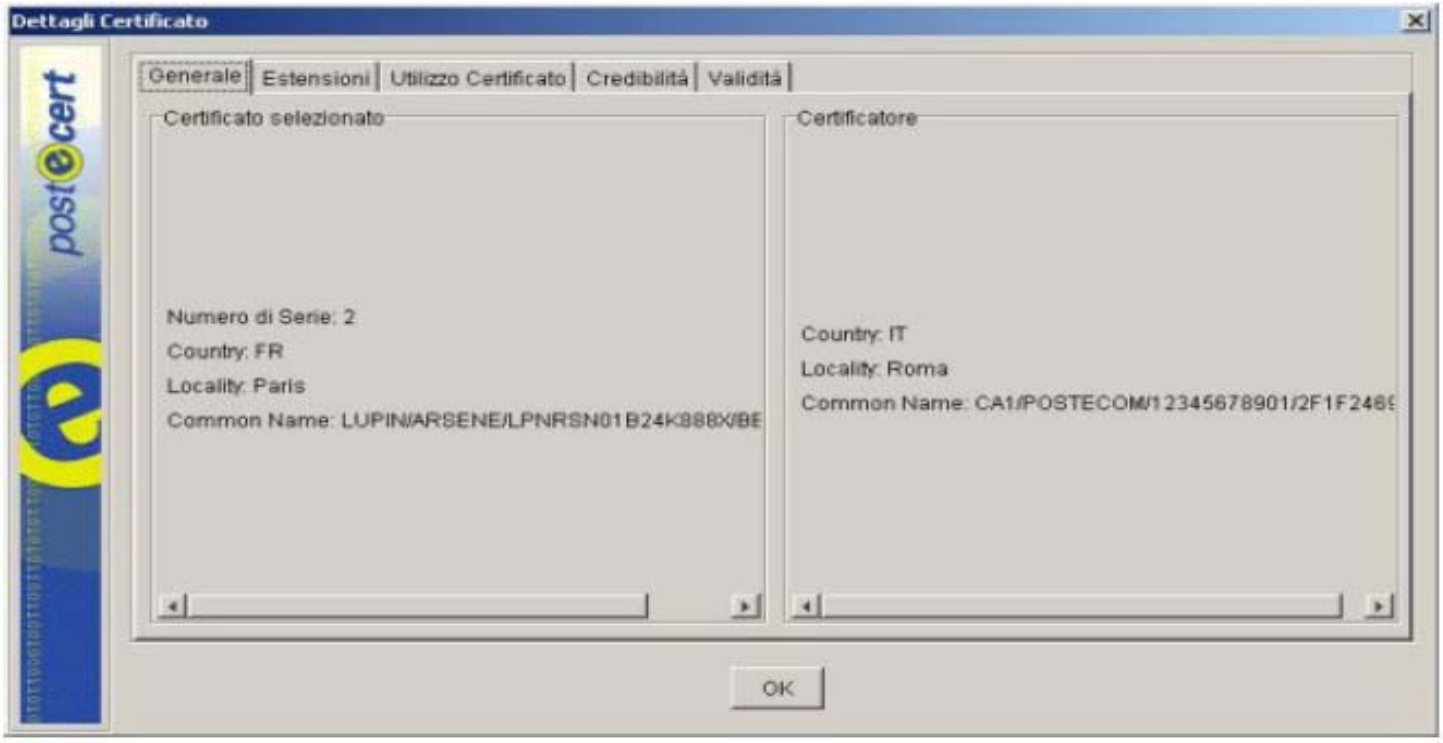
  nella firma in formato PKCS#7 della norma italiana viene allegato il certificato utente usato per firmare, che viene poi validato.
- Normalmente per verificare un certificato, si controlla la firma apposta dal certificatore, usando un **certificato di root** (disponibili sul sito dell'AIPA oppure preinstallati in modo sicuro nel software di verifica)
- Firma&Cifra commette un errore fatale: se nel PKCS#7 viene inserito oltre al certificato utente anche il certificato di root usato per firmare quello utente, il software non ricerca il certificato di root nel proprio database, ma si accontenta di quello fornito e decreta che la firma è autentica!

La Firma di Arsène Lupin

- Generiamo un falso certificato di root inserendo una denominazione uguale a quella di uno dei certificatori dell'elenco pubblico(nel nostro esempio PosteCom)
- Usiamo questo certificato per firmare un falso certificato utente intestato nel nostro caso ad Arsène Lupin.
- Usiamo il falso certificato utente per firmare il documento
- Aggiungiamo al documento firmato il certificato di PosteCom (root)



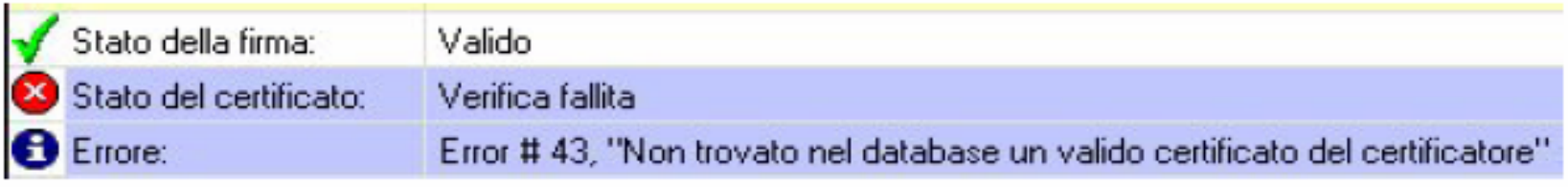
Continua: Arsène Lupin



Roberto Palombo (PosteCom), nella sua risposta su InterLex, sostiene che questo comportamento del software è “by design”, e che si limiteranno a rilasciare un aggiornamento che richieda “una più esplicita volontà dell’utente nell’importare un certificato root”.

Soluzioni!!!

Comportamento di un altro software!



Stato della firma:	Valido
Stato del certificato:	Verifica fallita
Errore:	Error # 43, "Non trovato nel database un valido certificato del certificatore"

- La “soluzione” è quanto mai semplice: implementare correttamente il software
- Spacciare il Bug di Firma&Cifra per una “lieve” svista è quantomeno riduttivo e fuorviante

Attacco tramite Trojan

- Il 13 Maggio 2003, varie testate, online annunciano : "*Il Dipartimento di Informatica della Statale di Milano ha violato un'applicazione di firma distribuita da un certificatore dell'AIPA*"
- Non si tratta di un Bug del software utilizzato. Il Team dell'Università di Milano ha utilizzato un **Trojan** per "firmare" documenti all'insaputa dell'utente.
- E' chiaro che su una macchina compromessa l'uso di qualsiasi software di firma non può essere garantito!
- Il Team che ha realizzato il Trojan ha dichiarato: "L'attacco è stato realizzato sfruttando alcune note debolezze dell'ambiente Java quando eseguito su sistemi operativi senza opportune protezioni, come accade per esempio con l'installazione di default di una Java Virtual Machine" (esempio Windows)

Possibili Soluzioni?

- Utilizzare un ottimo Antivirus !!!
- Purtroppo, come sappiamo, nessun Antivirus può proteggere da un Virus o Trojan che non sia ancora stato creato!!!
- Conclusioni: Nessuna Soluzione… ☺

security flaw nel formato OpenPGP (24 Marzo 2001)

- Bisogna sempre fare i conti con i Bug dei software crittografici.
- E' un attacco che sfrutta un bug sul formato aperto internazionale OPENPGP
- Scoperto da due crittologi della Repubblica Ceca, Vlastimil Klima e Tomas Ros nel 2001
- Alcune informazioni sulle chiavi pubbliche e private di un utente non sono protette adeguatamente nel file di configurazione del tool che si sta utilizzando
- Modificando queste informazioni con dei dati prestabiliti si possono ottenere dei valori numerici utilizzabili per il calcolo della chiave privata dell'utente.

Punto della situazione

- Gli attacchi discussi fino a questo momento, non sono stati causati dall'inefficienza degli algoritmi crittografici utilizzati, ma bensì dall'implementazione errata dei software di firma.
- Ma tutti i metodi crittografici utilizzati sono "*sicuri*"?

Attacco MD5

- E’ possibile creare collisioni di una funzione hash (md5)?
- Paul van Oorshcot scrisse un paper nel 1994.(“**Parallel Collision Search with Application to Hash Functions** “)
- Egli affermò che con l’utilizzo di un super computer da $10,000,000 era possibile ottenere collisioni con MD5 in **24 giorni**!!!
- Dato in input il messaggio m1 e l’Hash(**m1**), è possibile trovare un altro messaggio simile **m2** che si differisce per pochi byte da m1 tale che Hash(m1) = Hash(m2).
- Paul van Oorshcot pensò di aggiungere ad m1 dei caratteri che non fossero visibili in un Word Processor, oppure aggiungere dei caratteri che poi venivano nascosti da una proprietà del Word Processor.

Un altro Attacco MD5

- Il 17 Agosto 2004 Xiaoyun Wang, Dengguo Feng, Xuejia Lai e Hongbo Yu hanno pubblicato un articolo che annuncia la scoperta di un nuovo metodo di attacco per trovare collisioni, applicabile a molti degli algoritmi attualmente in uso (quelli basati sulla forma generale proposta da MD4).
- L'articolo è avvallato da collisioni complete per MD4, MD5, HAVAL-128, RIPEMD, ottenute in poche ore anziché gli svariati mesi che un moderno super-computer impiegherebbe.
- Non hanno rilasciato alcuna informazione sull'attacco.

Esempio concreto:

```
$ od -t x1 MD5.1
0000000 d1 31 dd 02 c5 e6 ee c4 69 3d 9a 06 98 af f9 5c
0000020 2f ca b5 87 12 46 7e ab 40 04 58 3e b8 fb 7f 89
0000040 55 ad 34 06 09 f4 b3 02 83 e4 88 83 25 71 41 5a
0000060 08 51 25 e8 f7 cd c9 9f d9 1d bd f2 80 37 3c 5b
0000100 96 0b 1d d1 dc 41 7b 9c e4 d8 97 f4 5a 65 55 d5
0000120 35 73 9a c7 f0 eb fd 0c 30 29 f1 66 d1 09 b1 8f
0000140 75 27 7f 79 30 d5 5c eb 22 e8 ad ba 79 cc 15 5c
0000160 ed 74 cb dd 5f c5 d3 6d b1 9b 0a d8 35 cc a7 e3
$ od -t x1 MD5.2 (ho evidenziato le differenze con <>)
0000000 d1 31 dd 02 c5 e6 ee c4 69 3d 9a 06 98 af f9 5c
0000020 2f ca b5<07>12 46 7e ab 40 04 58 3e b8 fb 7f 89
0000040 55 ad 34 06 09 f4 b3 02 83 e4 88 83 25<f1>41 5a
0000060 08 51 25 e8 f7 cd c9 9f d9 1d bd<72>80 37 3c 5b
0000100 96 0b 1d d1 dc 41 7b 9c e4 d8 97 f4 5a 65 55 d5
0000120 35 73 9a<47>f0 eb fd 0c 30 29 f1 66 d1 09 b1 8f
0000140 75 27 7f 79 30 d5 5c eb 22 e8 ad ba 79<4c>15 5c
0000160 ed 74 cb dd 5f c5 d3 6d b1 9b 0a<58>35 cc a7 e3
$ md5sum MD5.*
a4c0d35c95a63a805915367dcfe6b751 MD5.1
a4c0d35c95a63a805915367dcfe6b751 MD5.2
```

- Se ai due file aggiungessimo qualunque altro messaggio, H(md5.1)=H(md5.2)
- il noto "*length extension attack*" degli hash incrementali.

Tool conosciuti (Open Source)

- GnuPG (OPEN SOURCE)
- GPA interfaccia grafica per GnuPG
- WinPT interfaccia grafica GnuPG per Windows
- CTC sviluppato in Java.
- ...

Tool conosciuti (chiusi)

- PGP
- Dike di Info Camere
- SecurSIGN di BNL Multiservizi S.p.A.
- ...

Fine

- Grazie per l’attenzione