Reti di Calcolatori

Crittografia &
Java Cryptographic Architecture (JCA)

La crittografia

La crittografia è un particolare processo grazie al quale, per mezzo di sofisticati algoritmi, è possibile trasformare una sequenza di byte con senso logico (messaggio) in un altra del tutto incomprensibile.

Scopo della crittografia è consentire la trasmissione di un messaggio in forma non intellegibile ad altri che non sia il destinatario inteso, che deve essere il solo a poterne capire il significato.

La trasformazione avviene grazie ad una chiave: solo chi possiede la chiave per aprire e chiudere il messaggio potrà criptare e decriptare il messaggio.

Gli elementi

Il metodo tramite il quale, dato un messaggio in chiaro, si produce il corrispondente messaggio cifrato viene chiamato algoritmo di cifratura.

L’algoritmo deve garantire una proprietà fondamentale: dalla versione cifrata del messaggio deve essere impossibile risalire al messaggio originale che l'ha generata.

Algoritmi a chiave simmetrica ed asimmetrica

Un algoritmo di cifratura si dice a chiave simmetrica quando la stessa chiave K viene usata sia per la cifratura, sia per la successiva decifratura.

Un algoritmo è a chiave asimmetrica quando cifratura e decifratura richiedono due chiavi diverse.

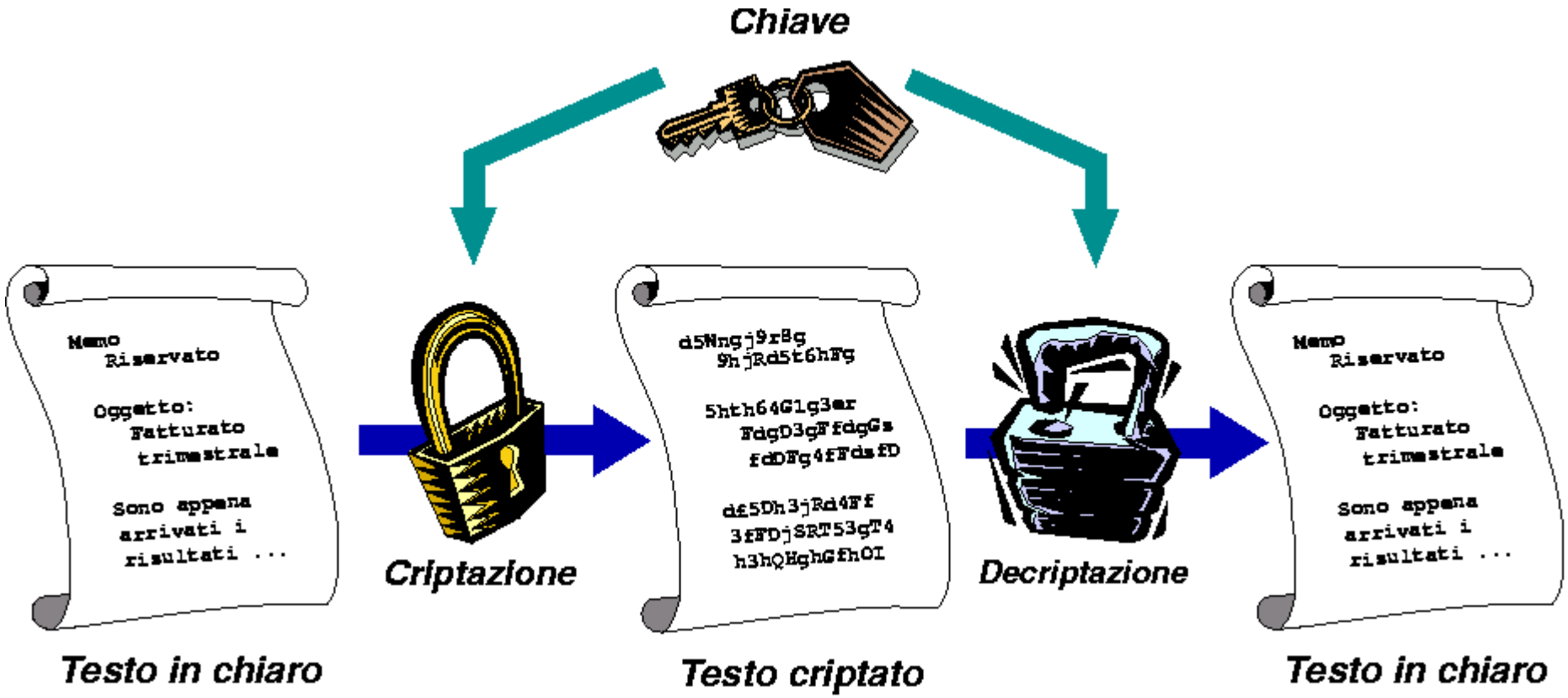
Algoritmi a chiave simmetrica

Gli algoritmi a chiave simmetrica sono semplici da realizzare, relativamente efficienti e quindi assai diffusi: fra i più noti vi sono il DES (Data Encryption Standard), triple-DES, IDEA, ecc.

Sfortunatamente, la chiave non può essere trasmessa come un normale messaggio, in quanto essa stessa potrebbe essere intercettata, e ciò renderebbe inutile la cifratura del messaggio.

Perché questo schema funzioni occorre dunque che la chiave possa essere inviata dal mittente al destinatario in qualche altro modo.

Crittazione simmetrica



Algoritmi a chiave asimmetrica

Sebbene con gli algoritmi a chiave asimmetrica la chiave di cifratura non debba essere trasmessa, occorre comunque che il destinatario "conosca" in qualche modo l'altra chiave, quella di decifratura.

Questo problema è stato risolto da una particolare categoria di algoritmi a chiave asimmetrica, noti come algoritmi basati su chiave pubblica e chiave privata.

Non esistono a priori una chiave per cifrare e una chiave per decifrare: semplicemente, se una delle due chiavi viene usata per cifrare, occorre l'altra per decifrare.

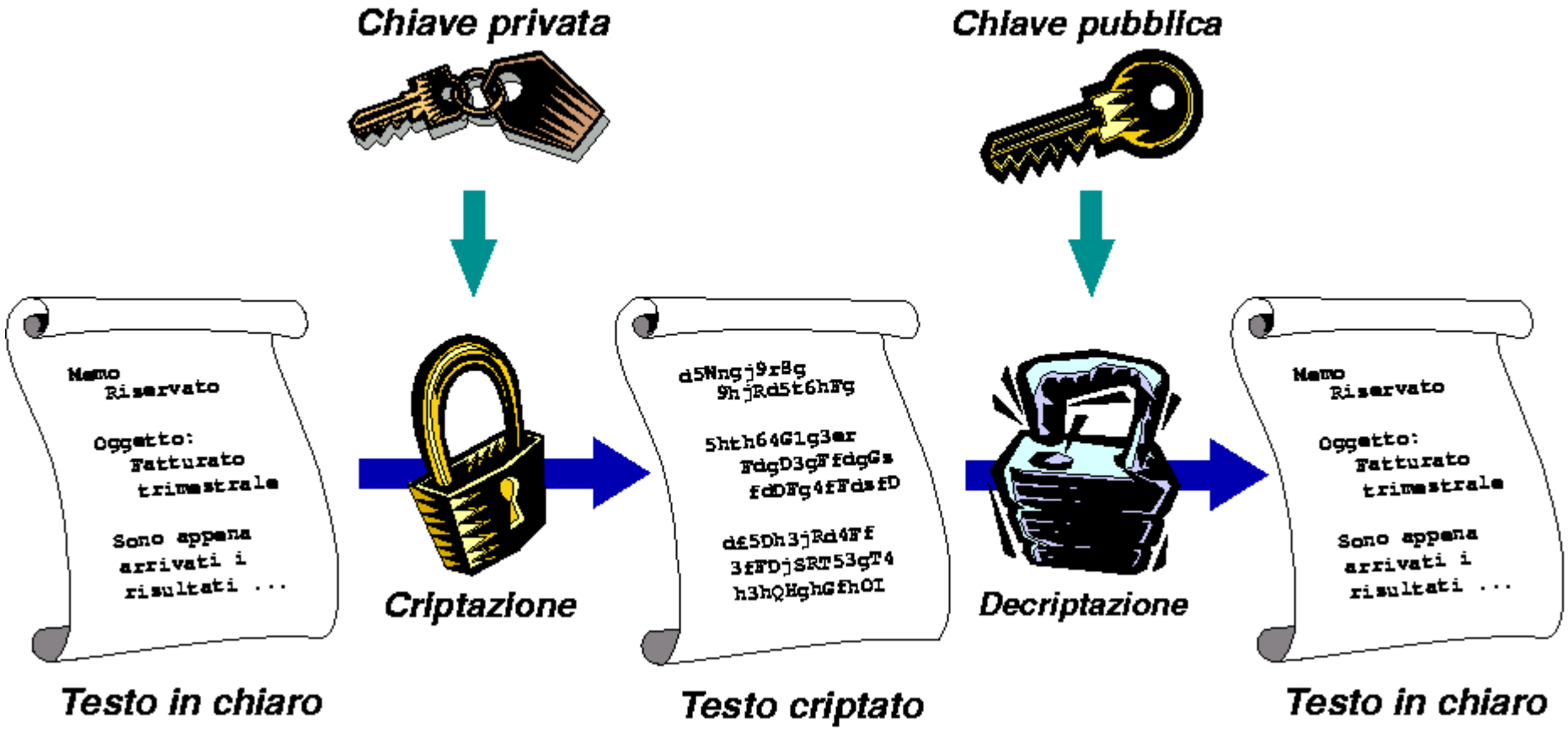
Chiave asimmetriche

Le due chiavi sono dunque perfettamente intercambiabili: tuttavia, la conoscenza della chiave usata per cifrare non è di alcuna utilità per decifrare, in quanto per farlo occorre l'altra chiave.

L'algoritmo è strutturato matematicamente in modo che da una chiave sia impossibile risalire all'altra; in questo modo si supera il problema della trasmissione della chiave che affliggeva gli algoritmi a chiave simmetrica.

Chiaramente, la potenza di questo schema sta nel non richiedere la trasmissione "sicura" della password - anzi, una delle due è talmente poco sicura da poter essere pubblicata.

Crittazione asimmetrica



Identificare il destinatario

Se A vuole inviare un messaggio a B con la certezza che solo B lo possa leggere, basta che lo cripti usando la chiave pubblica di B (che è disponibile proprio in quanto pubblica: si può pensare di recuperarla da appositi elenchi, non diversi dagli elenchi telefonici).

Dopo di ciò, solo B potrà leggere il messaggio cifrato, in quanto è il solo a disporre della chiave privata necessaria per la decifratura.

Chiunque intercetti il messaggio non potrà decifrare nulla, neppure procurandosi la chiave pubblica di B (che pure è disponibile), in quanto quest'ultima è incapace di decifrare un messaggio da lei stessa cifrato.

NB: B non può avere alcuna certezza che l'autore del messaggio sia proprio A: infatti, chiunque può prendere la chiave pubblica di B e inviargli un messaggio.

Indentificare il mittente

Se A vuole inviare un messaggio a B con la certezza che sia stato mandato da A, basta che lo cifri usando la sua chiave privata (di A).

In questo modo, chiunque potrà leggere il messaggio cifrato (perché la chiave pubblica di A necessaria per la decifratura è liberamente disponibile), ma nel farlo avrà anche la certezza che solo A può esserne l'autore, in quanto solo A poteva conoscere la chiave privata di A che **certamente è stata usata per la cifratura.**

Questa certezza deriva dal fatto che il messaggio si decifra con la chiave pubblica di A.

Comunicazioni private

Se A vuole inviare un messaggio a B con la certezza che solo B lo possa leggere **e inoltre comprovando che l'autore è** proprio A. Il messaggio M verrà cifrato una prima volta usando la chiave privata di A, e subito dopo una seconda volta usando la chiave pubblica di B (l'ordine di questi due passaggi è irrilevante).

Ovviamente, tale messaggio è incomprensibile per chiunque lo intercetti. Ma soprattutto, per ricostruire il messaggio in chiaro B dovrà decifrare due volte il messaggio ricevuto: una prima volta usando la sua chiave privata (e ciò fa sì che solo B possa leggere il messaggio), e poi una seconda usando la chiave pubblica di A (il che comprova che proprio A ne è l'autore).

Digest di messaggi

Un DIGEST (o hash) di un messaggio è una particolare versione cifrata del messaggio, caratterizzata dal fatto di avere dimensione fissa indipendente dalla dimensione del messaggio originale.

Gli algoritmi usati per calcolare il digest devono assicurare che sia "estremamente improbabile" che due messaggi diversi possano condurre alla medesima impronta.

Grazie a questa proprietà, i digest possono essere usati per identificare in modo certo e univoco i dati del messaggio, di cui costituiscono a tutti gli effetti una sorta di "impronta digitale".

Questa tecnica è spesso usata per "garantire l'autenticità" di oggetti, nel senso di assicurare chi li riceve che essi non sono stati alterati.

Signature di messaggi

Una SIGNATURE (o firma) di un messaggio è una stringa cifrata, spesso relativamente corta e di lunghezza fissa, ottenuta dal messaggio originale tramite un algoritmo e una chiave (privata).

Questa stringa viene usata per firmare i dati: a tale scopo è solitamente trasmessa insieme ai dati da cui è stata ricavata e di cui costituisce la firma digitale.

Tale firma può poi essere verificata (tipicamente dal destinatario) applicando ai dati e alla firma ricevuta l'algoritmo di verifica, unitamente alla chiave pubblica del mittente.

L'architettura crittografica di Java (JCA)

Formata da una **engine class** e da un **provider** .

Una **engine class** **è una classe che definisce le funzionalità di un dato** tipo di algoritmo crittografico, senza però fornire alcuna implementazione (classi astratte). Ad esempio, MessageDigest, Signature e KeyPairGenerator sono tre diverse engine class, e definiscono rispettivamente le funzionalità attese dagli algoritmi di tipo Message Digest, Signature, e Generatori di coppie di chiavi.

Un **provider** **("fornitore") è un package che fornisce l'implementazione** concreta di un certo insieme di funzionalità crittografiche. Più provider, di diversi produttori, possono coesistere e interoperare.

L'architettura crittografica di Java in versione base (JCA) definisce quindi il framework generale per la crittografia, lasciando alla **Java Cryptographic Extension (JCE)** il compito di fornire l'implementazione completa delle funzionalità di cifratura e decifratura.

Esempi:

1.MessageDigestExample.java , **Digest di messaggi**.

1.PrivateExample.java , **Algoritmi a chiave simmetrica** con chiave privata.

1.PublicExample.java , **Algoritmi a chiave asimmetrica** con chiave pubblica e privata.

1.DigitalSignatureExample.java , **Signature di messaggi.**